

ORIGINAL RESEARCH

Open access

Advancing AI-Augmented Business Process Management through Knowledge-Driven Explainability and Agile Knowledge Sharing

Maria Gonzales^{1*}, Jose Santos¹, Anna Cruz², Carlos Reyes³

Abstract

Merging Artificial Intelligence (AI) with Business Process Management Systems (BPMSs) has given rise to AI-Augmented Business Process Management Systems (ABPMSs). These modern setups deliver flexible adaptability, instantaneous process refinement, and superior knowledge management workflows. Yet, notable hurdles persist, especially regarding contextual clarity, user buy-in, and the incorporation of behavioral dynamics. This research introduces a novel conceptual model that unifies the Socialization, Externalization, Combination, and Internalization (SECI) knowledge paradigm with Agile workflows (specifically Scrum) and advanced AI capabilities like explainable AI (XAI), process mining, and Robotic Process Automation (RPA). This setup streamlines the formalization, validation, and dissemination of insights through structured, intuitive software and collaborative frameworks, particularly Communities of Practice (CoPs). The system highlights context-sensitive transparency, step-by-step deployment, and iterative refinement to foster productive human–AI partnerships. It delivers both conceptual and practical blueprints for linking AI strengths with corporate knowledge assets. The introduced model eases the upgrade path from legacy BPMSs to advanced ABPMSs by utilizing systematic practices and technologies. This strategy accelerates knowledge distribution and operational refinement, mapped out via exhaustive BPMN 2.0 diagrams.

Keywords Explainable artificial intelligence (xAI), AI-augmented business process management systems (ABPMSs), Knowledge management (KM), Scrum framework, Lessons learned, Patterns and best practices

*Correspondence:

Maria Gonzales

maria.gonzales@up.edu.ph

¹ Department of Business Process Management and AI Integration, Faculty of Business, University of the Philippines Diliman, Quezon City, Philippines

² Department of Knowledge-Driven Explainability, Faculty of Engineering, University of Santo Tomas, Manila, Philippines

³ Department of Agile Knowledge Sharing and AI, Faculty of Business, Ateneo de Manila University, Quezon City, Philippines

Introduction

Over time, intellectual assets have taken center stage in corporate strategy and are universally seen as a cornerstone of a firm's market edge. Knowledge management (KM) has emerged to counter the pressures of globalization, technological shifts, and the broader intellectual demands on modern firms. The core goal of KM is to sharpen a company's competitive edge and safeguard its internal intellectual frameworks.

Too often, corporate software offers only a superficial fix, focusing solely on archiving and indexing information—frequently requiring human oversight—without incorporating automated tools that could streamline management while actively mining and verifying that data. Meanwhile, a fresh paradigm is taking shape: the widespread availability of operational log data, combined with breakthroughs in AI, has cleared the way for a novel category of enterprise software

called ABPMSs. These platforms allow operational pathways to evolve on the fly, bypass the need for hardcoded software reconfigurations during disruptions, and independently spot, verify, and roll out optimization strategies instantly.

This shift is mirrored by active discussions in academic circles. A prime example is the foundational research by Dumas *et al.* [1], which explores how AI can be woven into BPMSs to serve as a cornerstone for future work. That paper aims to spark ongoing experimentation and engineering in AI-centric Business Process Management (BPM), mapping out a strategic roadmap for upcoming innovations while emphasizing how AI tools radically reshape corporate workflows.

Specifically, embedding AI components within a BPMS unlocks new avenues for automation, boosting operational flexibility and productivity while requiring minimal yet strategic oversight from human operators. In contrast to legacy BPMSs that operate strictly within pre-programmed rules and diagrams, an ABPMS evaluates live operational states across multiple workflows to pinpoint adjustments that maximize output.

Despite rapid strides in combining AI with BPMSs, the manifesto by Dumas identifies several roadblocks holding back broad corporate rollout and long-term utility. A major barrier is delivering context-sensitive explainability, meaning an ABPMS must be capable of generating clear, situational justifications for its automated choices, actions, and alignments with high-level corporate targets. These justifications must be practical and comprehensible for everyday staff, closing the gap between complex algorithms and human comprehension.

Another major hurdle involves accounting for human and organizational behavior. This requires software capable of tracking and deciphering task behaviors and group routines over prolonged observation windows, allowing a deeper grasp of intricate operational dynamics.

Furthermore, a vital trait for any functional ABPMS is its capacity to engage in multi-way dialogue with staff, offering justifications for its logic while proactively mapping out process adjustments or improvements.

Lastly, moving away from legacy BPMS layouts toward an ABPMS setup must happen progressively through a phased deployment strategy. This allows firms to phase in features over time, matching their immediate priorities while dodging major business disruptions.

Though the manifesto highlights these diverse pain points, it falls short of offering concrete workarounds or step-by-step methodologies to fix them, pointing to a distinct gap in practical application.

This study introduces an architecture built to bridge those exact gaps. Reflecting these realities, an uncomplicated interface—resembling a discussion board—was structured to emphasize the explicit input of every agile developer involved, maintaining an intuitive layout to stimulate active knowledge sharing. On top of that, this repository design weaves in automated validation phases, maximizing efficiency and making the entire pipeline more agile.

In particular, the methodology outlined by Yakyma [2] serves as a reference point, illustrating how an agile setup—the Scrum framework, specifically—can naturally map out and track the distinct phases of the SECI matrix.

However, the primary drawback of that specific model is its failure to pinpoint exactly when corporate lessons and best practices are actually codified and signed off within the broader KM pipeline. Additionally, it leaves out how standard Scrum personnel, like the Scrum Master or CoP networks, help validate and broadcast this intellectual capital across the wider firm. The architecture introduced here addresses these shortfalls by combining a dedicated software utility with content management guidelines to anchor the KM cycle. The software itself centers around the following capabilities:

- Operating as an intuitive, browser-based environment;
- Allowing teams to instantly record a lesson at any moment, broadcast it to relevant teams, and index it for rapid querying;

- Providing dedicated repositories for design patterns and historical lessons;
- Maintaining a structured queue of chat threads and insights to be debated and approved (or rejected) during team reviews and CoP meetups;
- Incorporating advanced query algorithms that pair keyword indices with cross-referenced hyperlinks.

The main drive of this work is to construct an intelligence-driven environment that patches the current flaws in ABPMS setups, injecting real-time data loops and organized information exchange to drive transparency. Merging operational behavioral data with Communities of Practice (CoPs) will further stimulate and simplify user adoption. Ultimately, a core priority is delivering a seamless upgrade path from older BPMS architectures via modular, phase-by-phase rollouts. Specifically, this inquiry centers on two pivotal questions: How can AI engines boost the flexibility and throughput of a BPMS without sacrificing situational clarity? What role do agile collaboration practices play in refining AI-driven business processes?

State of the Art

In recent years, BPMSs have advanced considerably through the integration of AI methodologies, leading to the creation of ABPMSs. These advanced frameworks aim to boost operational throughput and aid human personnel in executing intricate processes and making critical choices, while simultaneously mitigating the typical limitations and hurdles of legacy BPMSs. Within this domain, several literature studies serve as foundational cornerstones; for instance, the research by Dumas *et al.* [1] underscores how an ABPMS can implement a Hybrid Process Intelligence methodology that achieves an optimal balance between automated execution and human oversight. This strategy not only encourages continuous learning but also increases the responsiveness of business processes to evolving requirements.

Nonetheless, numerous practical and conceptual challenges remain unresolved. This section is dedicated to evaluating the current state of the art, pinpointing specific domains of interest that this investigation aims to address. Concurrently, a synthesis of contemporary difficulties and trends will be presented, evaluating the prospective trajectories of AI deployment within BPMS configurations.

The impact of AI on KM

AI has emerged as a foundational technology capable of enhancing every stage of the knowledge lifecycle, including its creation, storage, distribution, and utilization. As demonstrated in the research by Kovačić *et al.* [3], AI techniques have been successfully deployed in intricate domains like manufacturing, aiding in the refinement of corporate strategies and operational workflows.

When used to streamline manufacturing pipelines, AI enables the detection of recurring patterns across disparate datasets, facilitating the generation and uncover of novel insights that firms can leverage to boost competitive advantage and drive continuous improvement. Furthermore, literature analysis yields significant proof that AI can simultaneously generate new insights—such as through the live monitoring of operational telemetry—and assist personnel in augmenting their existing expertise. Additionally, as indicated by Psarommatis and Kiritsis [4], the insights generated by automated AI frameworks are frequently insufficient on their own to rectify anomalies that arise during manufacturing processes; consequently, the internal expertise and corporate best practices preserved within an enterprise knowledge repository remain vital for mitigating these operational issues during live execution.

AI demonstrates significant potential in fields like Data Management and Analysis, as observed by Enholm *et al.* [5], helping to govern the expanding volumes of information gathered via sophisticated instruments to isolate valuable intelligence for decision-making support. This is achieved by automating intricate operations like information aggregation, indexing, and evaluation, thereby substantially decreasing manual labor requirements.

Moreover, as emphasized by Dumas *et al.* [1], the deployment of state-of-the-art Natural Language Processing (NLP) methodologies advances information retrieval, granting swift and precise access to pertinent information. These technologies enable the establishment of structured knowledge repositories with minimal human intervention. Utilizing sophisticated NLP instruments not only optimizes information retrieval but also increases the efficiency of accessing vital data by automating knowledge collection from multiple channels and consolidating it in a structured and uniform manner. This automation allows for the construction of highly granular and adaptable knowledge repositories with minor manual involvement, thereby freeing resources for higher-value operational tasks. Additionally, AI streamlines the customized distribution of information, tailoring it to the distinct needs and tastes of users, which upgrades the overall KM process. AI-driven instruments, in turn, help solidify institutional synergy because they deliver perspectives drawn from a firm's aggregated intelligence, sparking data-driven discussions focused on problem resolution. AI is also recognized for bolstering organizational learning and synergy. In this regard, the investigation by Taherdoost and Madanchian [6] demonstrates how individuals leverage a deeper comprehension of their operational settings to utilize real-time analytical and predictive instruments. Jarrahi *et al.* [7] also underline the value of knowledge sharing, given that AI bridges connections and builds networks among personnel tackling comparable challenges, thereby encouraging cross-departmental synergy. Concurrently, AI, as noted by Alavi *et al.* [8], delivers custom recommendations and transparent tools that streamline asset access, nurturing a corporate ethos centered around continuous learning. Indeed, AI, particularly Generative AI (GenAI), simplifies knowledge transfer, aiding onboarding and cultivating a learning mindset; however, it simultaneously introduces anxieties regarding over-dependence on automated systems and the disclosure of proprietary records. It likewise boosts output and original thinking by automating operations and offering practical perspectives. Even so, hurdles like algorithmic bias and the isolation of junior staff must be tackled.

Conversely, from a tactical standpoint, AI proves vital for fostering innovation and engineering new offerings and services. As underlined by Thakuri *et al.* [9], the influence of AI on KM is substantial, sharpening decision-making pathways via data assessment and Machine Learning (ML). It is thus apparent that AI improves knowledge capture and distribution by automating data collection and dissemination, making it more available to staff, thereby driving innovation through market trend evaluation that results in the creation of fresh products and services.

This permits the refinement of internal workflows and the codification and distribution of best practices, such as in the Compliant Knowledge Transfer Model introduced by Linder [10], which can guide the movement and allocation of expertise within a firm so it can be leveraged for subsequent iterations of a related product or process. This model, however, relies on a sequential blueprint that is difficult to align with the needs and traits of enterprises operating in aggressive business arenas, such as modern markets defined by increasingly condensed delivery timelines. Furthermore, integrating AI into KM introduces distinct difficulties: the investigation by Chen [11] points out that automating steps like data tagging and classification can foster an unhealthy reliance on a system's technical capability, necessitating human supervision to avert errors. The paper analyzes the interplay between AI and KM, demonstrating how their integration can upgrade corporate output and innovation. KM has been shown to deliver the essential context and architecture for AI models, while AI can accelerate the uncovering of insights and optimize content curation. Multiple investigations indicate that productive collaboration between human specialists and AI models is vital to extract maximum value from KM architectures. The literature reveals that the most prominent consequence of AI on KM is an increase in efficiency. This consequence is crucial because it directly shapes how firms operate and govern their intellectual capital: by automating routine operations such as data labeling, indexing, and categorization, AI markedly minimizes the time and labor needed for KM workflows. Consequently, enterprises can direct their attention toward tactical goals rather than manual chores, gaining quicker and more precise access to intelligence, which is vital in modern, rapidly moving commercial environments.

Impact of AI on BPMSs

The fusion of AI with BPMSs is radically transforming how enterprises orchestrate and refine their operational workflows, a shift substantiated by a broad range of academic literature. Within this domain, AI functions multi-dimensionally to augment operational throughput, expand automation, and sharpen decision-making quality, fundamentally shifting

organizational structures toward heightened agility and continuous innovation. Casciani *et al.* [12] investigate the innovative territory of ABPMSs, which utilize AI methodologies to elevate the deployment and flexibility of corporate processes. They argue that a foundational characteristic of these frameworks is their interactive capability, which permits proactive engagement with human operators to render business architectures both more efficient and accessible. Nevertheless, deploying these tools necessitates careful consideration of ethical constraints and algorithmic transparency, ensuring that human oversight and verification remain central to the architecture.

In addition, Kokala [13] emphasizes the disruptive potential of AI-driven workflows and cognitive automation in reshaping the landscape of Business Process Management. By harnessing an array of cutting-edge technologies, enterprises can attain unprecedented breakthroughs in cost optimization, quality enhancement, and operational efficiency. Zebec [14] similarly highlights the capacity of AI to elevate analytical decision making within BPM frameworks. This technology exerts a direct influence on organizational responsiveness, flexibility, and overall performance, thereby ensuring swift alignment with fluctuating market dynamics.

Furthermore, AI serves as a cornerstone for optimizing Supply Chain Management (SCM), navigating massive datasets, and consequently establishing higher tiers of organization and knowledge management performance. As demonstrated by Helo and Hao [15], embedding AI architectures within SCM allows firms to extract actionable intelligence from highly complex datasets, thereby refining operational flows, demand modeling, and client service standards with an eye toward continuous enhancement. This evolution is driven by ongoing ML loops, which cultivate persistent adaptability to operational and commercial shifts. AI also introduces a powerful asset for managerial choice. As Aggarwal [16] notes, AI reinforces BPM via predictive analytics, data-grounded insights, and automated recommendation engines that augment and optimize diverse decision-making methods. Perhaps most vital is the capacity to evaluate systemic risks and future opportunities through active synergy with internal human expertise, highlighting how AI sharpens administrative judgment to make strategic selections more robust and dependable. Correspondingly, Dumas *et al.* [1] champion the paradigm of ABPMSs—platforms that merge AI capabilities with human expertise to grant adaptive autonomy to workflows without eliminating the path for human intervention, thereby validating an equilibrium between automation and control.

Sophisticated automation, fueled by systems like RPA, represents a pivotal dimension of AI's enterprise footprint, as detailed by Rosemann and Szelągowski [17, 18]: automating routine, administrative duties not only diminishes operational expenditure and clerical errors but also reallocates human assets toward initiatives of greater creative and strategic value, thereby boosting output and inspiring innovative thinking. Schaschek *et al.* [19] further clarify how AI, coupled with predictive and prescriptive analytics, facilitates the detection and mitigation of process bottlenecks to maximize total operational throughput.

In specific domains like PLM, AI paves the way for automated workflows, predictive diagnostics, and customized consumer engagement, as articulated by Wang [20]. This advances both systemic scalability and the user experience. Moreover, combining AI with other vanguard technologies, such as IoT and blockchain, establishes more secure and transparent operations while simultaneously addressing the ethical friction tied to such innovations. Advanced methodologies like Large Process Models (LPMs) and NLP unlock new horizons for process design refinement, as demonstrated by Kampik and De Nicola [21, 22]. The capacity of GenAI models to parse event histories permits the discovery of latent patterns and prospective improvements in functional workflows, encouraging more productive and cooperative synergy between human operators and machine agents. According to Fahland [23], XAI maintains an indispensable position regarding trust and transparency. Foremost, providing clear, human-intelligible justifications for automated paths enhances user confidence in AI-driven agents within specialized operational settings, thereby elevating user assurance and reinforcing the perceived safety of deploying advanced digital frameworks. This also drives the widespread adoption of pioneering technologies while fostering an informed and accountable use of AI. Live process orchestrations backed by process mining, predictive analytics, or fluid interactions via NLP and conversational interfaces have been established for a considerable duration, as analyzed by Olatunji [24], Chapela-Campa [25], and Dumas [1]. This enables corporations to react instantly to volatile situations and deliver bespoke customer experiences, maximizing perceived value and brand satisfaction. Within this digital transformation arena, Gabryelczyk *et al.* [26] position AI as a

primary catalyst for intelligent automation, real-time telemetry, and process refinement. Rooting out operational friction and providing targeted workarounds represent core pillars for fostering corporate flexibility and customer retention.

Lastly, the research by Salvadorinho [27] targets the significance of KM within digital manufacturing paradigms, noting that the swift, actionable conversion of tacit expertise from seasoned specialists to incoming personnel remains a primary obstacle. That study underscores the utility of the BPMN 2.0 protocol for codifying functional instructions to mitigate workforce turnover, thereby safeguarding the enterprise's intellectual capital and driving a more effective exploitation of AI utilities. This alignment between KM and advanced technical tools further solidifies the sweeping, transformative impact of AI on corporate processes.

Challenges in ABPMSs

Beyond the myriad transformative advantages promised by incorporating AI into BPMSs, such as heightened efficiency and more intelligent decision-making, several significant research challenges warrant rigorous investigation, including the following:

1. **Situation-aware Explainability:** ABPMSs must provide explanations that are not only contextually relevant but also readily comprehensible to human users. Core inquiries typically concern the rationale behind specific task executions, the logic behind automated decisions, and the implications for business process objectives. The primary challenge lies in developing mechanisms that translate complex system outputs into human-readable narratives, serving as a basis for informed user decisions and subsequent system actions.
2. **Autonomy Framed by Constraints:** While ABPMSs are engineered for independent operation, this autonomy must be governed by defined operational objectives, parameters, and environmental constraints. These systems should function independently while maintaining an active dialogue with human participants. This requires conceptualizing a bi-directional interaction model that preserves system efficacy without undermining human oversight and accountability.
3. **Continuous Improvement:** ABPMSs should actively present executed process optimizations. This necessitates new methodologies by which a system can assimilate applications according to its functional capacity and expected results. The challenge involves integrating advanced AI capabilities that enable the system to learn and adjust its operations through feedback and evolving environmental conditions.
4. **Hybrid Process Intelligence:** This concept posits that an ABPMS must interact with humans as a "learning apprentice." Rather than mandating that users adapt to software, AI should learn from human workflows. The challenge, therefore, is designing architectures that observe human experiences and modify their behavior to operate in synergy with users during process execution.
5. **Trust and Reliability:** To secure adoption and operational efficacy, an ABPMS must establish credibility and trustworthiness. Information regarding the transparency of AI decision-making, the ethical use of data, and the handling of unforeseen exceptions must be accessible to users. Beyond merely improving system performance, convincing users to trust and rely upon the system's actions and decisions remains a critical hurdle.

These obstacles demonstrate that the advancement of ABPMSs is a complex, ongoing research endeavor. Each challenge presents unique opportunities for innovation and refinement within the BPM discipline.

In this context, the framework proposed in this study offers a pertinent contribution to overcoming these limitations. Initially, the framework incorporates a dedicated section for lessons learned and patterns, directly addressing the challenge of situation-aware explainability. It provides a structured methodology for recording and elucidating decisions, actions, and outcomes. By including automated workflows, the framework ensures that these explanations are precise, relevant, and accessible to users, allowing agents to act upon this information with greater efficacy. The framework's capacity to consolidate discussions and lessons learned into a centralized backlog serves as a vital mechanism for aggregating behavioral data over time. This historical repository enables the identification of recurring patterns within task

performance, organizational routines, and the nature of work itself. Furthermore, this utility enhances the identification and evaluation of complex behaviors through sophisticated search functions and hyperlinked relationships between objects and events. Additionally, it connects related entities across distinct timeframes, deepening the comprehension of complex behavioral dynamics within the organization.

The forum-centric architecture is engineered to foster active conversational engagement among Agile teams, directly addressing the challenge of Conversational Engagement. This design establishes an interactive setting where shared knowledge acts as a catalyst for continuous improvement and process adaptation. Integrated automated tools optimize this engagement by dynamically suggesting contextually pertinent actions, aligning these recommendations with operational constraints while maintaining a focus on team productivity and synergy.

Furthermore, the proposed framework effectively addresses the challenge of Broader Contextual Reasoning: by emphasizing the unconstrained capture and distribution of lessons, the framework accounts for a wide spectrum of situational factors, such as time, location, and group associations. This capability ensures that critical contextual variables are integrated into the decision-making process. Advanced search functionalities permit the linking of discussions and lessons to specific scenarios where they are most applicable, thereby elevating the overall utility of the framework. Consequently, the system is underpinned by a more comprehensive and context-sensitive methodology for organizational problem-solving and learning.

The modular architecture of the framework, coupled with its intuitive web-based interface, facilitates a flexible and gradual transition from legacy systems to more sophisticated ABPMSs. Consequently, organizations can tailor their transformation trajectories to meet specific requirements, allowing for the progressive introduction of key features such as lessons learned, collaborative discussions, and CoP processes. Given the framework's inherent adaptability, organizations are empowered to extend its application effectively across diverse business contexts in alignment with their unique operational goals.

Emerging trends in ABPMSs

While substantial progress has been achieved, the BPM discipline is undergoing a radical paradigm shift driven by continuous breakthroughs in AI. This transformation signals the prospective trajectory of BPMS configurations, achieved by embedding core AI capabilities such as ML, NLP, and predictive modeling. These integrations yield architectures that promise not only heightened operational efficiency and superior decision-making quality but also more customized and ethically grounded solutions aligned with modern enterprise demands. To map out the boundaries of innovation, synergy, and ethical responsibility within the sphere of AI-augmented BPM, prominent emerging trends are examined in the following sections.

The expansion of Intelligent Process Automation warrants specific attention, as it converges AI methodologies such as ML, NLP, and RPA. These sophisticated frameworks possess the capacity to extract insights from data, evolve dynamically, and execute autonomous decisions. This progression signifies a major transition from rigid, legacy automation to cognitive systems [19].

A dominant trend is the heightened academic and practical focus on transparency and explainability. Future iterations of ABPMSs will exhibit an expanded capacity to deliver granular and human-intelligible justifications for their automated actions, which is essential for fostering user confidence and encouraging cooperation between human personnel and AI models. Furthermore, these frameworks are projected to become more interactive, user-focused, and data-driven, enabling them to initiate multi-way dialogues with operators, advise on contextually appropriate adjustments, or reshape workflows in real time through live data analytics [28].

The literature [1] likewise stresses the necessity of refining human–AI interaction models. Within these frameworks, ABPMSs should function as learning apprentices that track human workplace routines and leverage this behavioral data,

alongside direct user feedback, to modify their automated workflows. This dynamic considerably strengthens the synergy between the AI agent and the human operator. Concurrently, mechanisms of continuous learning and iterative improvement are expected to empower these platforms to recalibrate their operations based on historical data and user interactions, ensuring long-term relevance and efficacy.

The ultimate objective of AI-driven BPM is the automation of corporate workflows via Cognitive Process Automation (CPA), which replicates human cognitive functions to execute tasks requiring complex reasoning, logic, and learning. Such functionalities allow enterprises to navigate volatile operational settings with increased sophistication. Additionally, AI architectures are increasingly designed as lifelong learners, incorporating self-improvement loops that model fresh data streams and novel experiences. Consequently, BPM instruments can outgrow their initial constraints and mature over time to maintain alignment with shifting market expectations. However, embedding AI deeply within BPM frameworks highlights the critical requirement for robust governance structures and ethical benchmarks. Deploying AI for high-stakes, critical determinations necessitates strict adherence to principles of fairness, accountability, and transparency to secure stakeholder trust and maintain regulatory compliance [1, 19].

Within Customer Experience Management, AI also unlocks novel avenues for exploration. It streamlines service personalization by evaluating existing consumer databases to guide contextually precise decisions, thereby deepening and maximizing client satisfaction [19].

Another highly relevant domain of inquiry centers on Generative Models for Synthetic Data. Contemporary organizations encounter a primary obstacle when validating AI and ML models within BPM frameworks due to the scarcity of comprehensive, robust datasets. Consequently, further investigations are required to assess the capacity of Generative Adversarial Networks (GANs), Transformers, and Autoencoders to construct artificial datasets. Researching how these distinct architectures can be unified to generate high-fidelity synthetic data for training purposes would substantially advance this field of study [28].

Lastly, dedicated efforts toward interoperability and standardization will empower disparate systems and technologies to interface seamlessly. This compatibility will simplify the integration of complex ABPMSs into pre-existing organizational IT infrastructures, boosting their total utility and systemic effectiveness [1].

Materials and Methods

Framework description

The subsequent section details the architectural components of the proposed innovative framework. This conceptual model is founded upon the strategic convergence of the SECI [29] knowledge paradigm with Agile engineering methodologies—specifically the Scrum framework—and the deployment of advanced computational technologies including XAI, RPA, and process mining (PM).

The Scrum framework delivers a transparent organizational blueprint grounded in explicit roles, such as the Scrum Master and Team Members, alongside iterative workflows. This structural design accelerates the creation, distribution, and validation of insights, promoting agile and collaborative governance across operational units [30].

The SECI model, which maps out the knowledge transformation cycle, delineates how intellectual capital matures across four distinct phases: beginning with informal socialization, where insights are exchanged implicitly among participants, and progressing through formalization, combination, and systemic dissemination as explicit knowledge.

The introduced methodological and technological framework is built upon the reference architecture illustrated in **Figure 1**. This structure comprises several distinct architectural tiers containing both methodological components—defined as a

set of rules, approaches, best practices, and conceptual definitions—and technological components, which encompass specific computational tools and techniques.

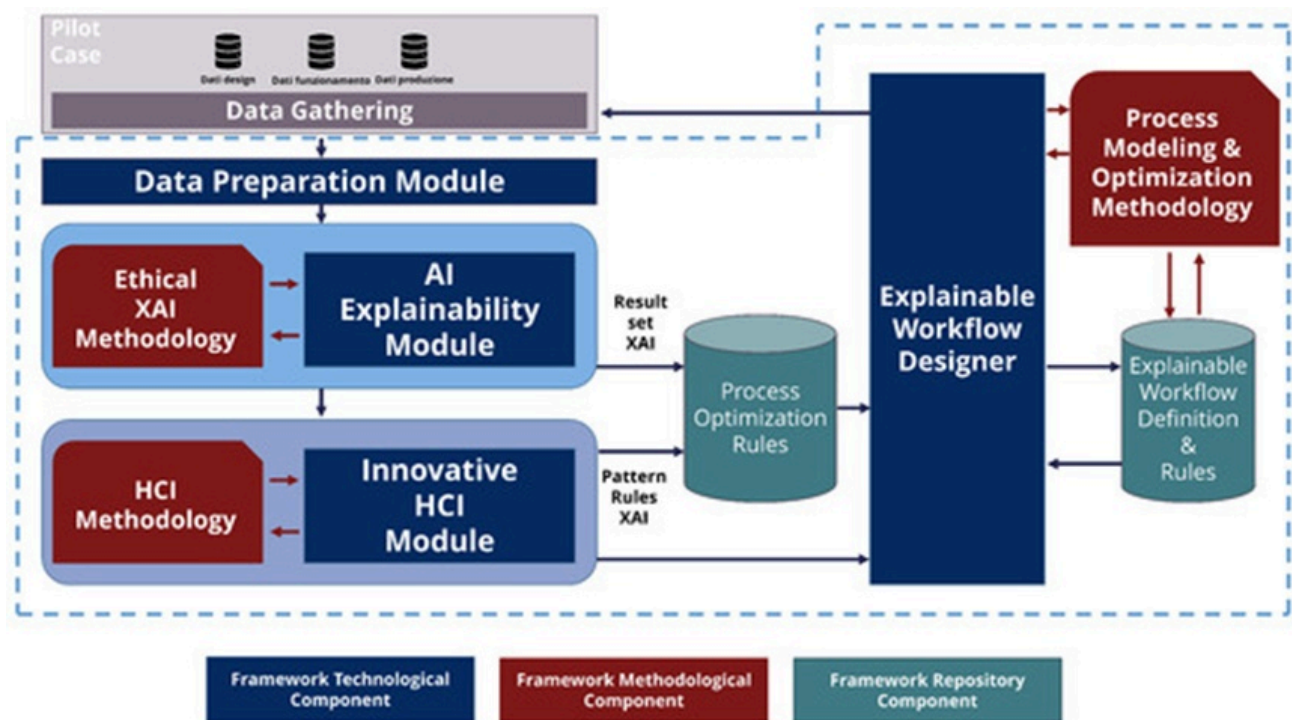


Figure 1. The proposed methodological and technological framework.

Data are gathered and systematically processed to ensure suitability for complex analytical evaluation within advanced computational systems. Once structured, the information follows the logical path mapped out in the reference architecture, progressing to the subsequent layer which houses the component dedicated to AI and XAI evaluation: the AI Explainability Module.

This specialized module incorporates an AI model tailored for processing domain-specific datasets alongside diverse methodologies and tools linked to XAI. This operation is driven by open-source libraries and utilities accessible via data science software (such as R3 or pipelines deployable via Python), reinforced by a business layer engineered to formalize technical outputs into formats suitable for the assessments executed within the module.

Concurrently, an appropriate framework, designated as the ethical XAI methodology, establishes a comprehensive set of directives and strategies useful for evaluating and enhancing the dependability and ethical alignment of the engineered systems. The Innovative Human–Computer Interaction (HCI) Module presents users with an array of graphical user interfaces designed to supply distinct operational roles with data and reports calibrated to the necessary level of granularity, thereby reinforcing diverse decision-making pipelines.

These user interfaces ensure that the underlying rationale governing the decision-making process is made explicit, thereby enabling cognitive assimilation by the human operator. These interfaces are structured around an HCI methodology that incorporates a series of established best practices and design standards native to the manufacturing sector to optimize interface engineering within this architectural layer. To guarantee usability and alignment with user needs, interface design was guided by foundational human–computer interaction (HCI) tenets [31], with rigorous focus directed toward User-Centered Design (UCD) [32] and Nielsen’s usability heuristics [33]. Primary architectural choices are governed by heuristics such as the visibility of system status, consistency and compliance with standards, error prevention, and favoring recognition over recall. These parameters were integrated to confirm that the Innovative HCI

Module facilitates efficient access to granular records while simultaneously supporting cognitive retention and user empowerment throughout the decision-making cycle. The outputs generated by the XAI methodologies, together with the insights surfaced by the Innovative HCI Module, are directed into a centralized repository termed Process Optimization Rules (constructed via structured and unstructured data archives). This repository functions as an enterprise knowledge base, serving to elicit and propagate all explicit knowledge generated through the functionalities detailed thus far.

Concurrently, this central repository supplies data to the Explainable Workflow Designer module. Utilizing standard rules and formalisms native to the BPM domain, this module enables the modeling, optimization, and execution of diagnosed industrial operations. It employs a notation that abstractly represents the syntax and semantics of traditional manufacturing sector formalisms across various product lifecycle stages—including Design, Manufacturing, Inspection, Delivery, and Maintenance and Service. This capability is realized through Business Process Engine platforms, such as Camunda or BonitaSoft, alongside dedicated Business Process Modeling solutions. To reinforce this module, the architecture incorporates an RPA-driven methodology designed to facilitate the Modeling and Optimization process by objectively and uniquely pinpointing the precise workflow operations suitable for automation. Furthermore, a secondary repository archives the specific rules essential for establishing an interpretable and explainable process workflow.

Ultimately, the workflows optimized and automated via this approach serve as the foundation for subsequent data gathering, which in turn drives iterative automation and optimization. This continuous loop establishes the cyclical character of the presented methodological and technological architecture.

More precisely, this study aligns with the perspective introduced by Yakyma [2], who illustrates how the distinct phases of the SECI methodology can be empirically identified within an Agile environment, specifically through the Scrum framework. Nevertheless, a major constraint of that framework is its failure to specify exactly when acquired lessons and best practices are formally documented and validated during the KM process. It also lacks clarity on how core Scrum roles—specifically the Scrum Master and Communities of Practice (CoPs)—participate in validating and circulating these knowledge assets across the firm. Additionally, while the SECI model is acknowledged as highly pertinent and constructive for this research, its practical implementation across diverse cultures has revealed distinct limitations. A conceptual investigation by Easa and Fincham [34] regarding the application of the model's socialization, externalization, combination, and internalization phases within Arab, Chinese, and Russian corporate settings identified both parallels and variances relative to the original Japanese framework. This indicates that the SECI model must be modified to incorporate a cultural variable to operate effectively within heterogeneous environments. Consequently, as illustrated in Figure 2, integrating a “Culturization” vector into the four core SECI processes is vital for multicultural deployment.

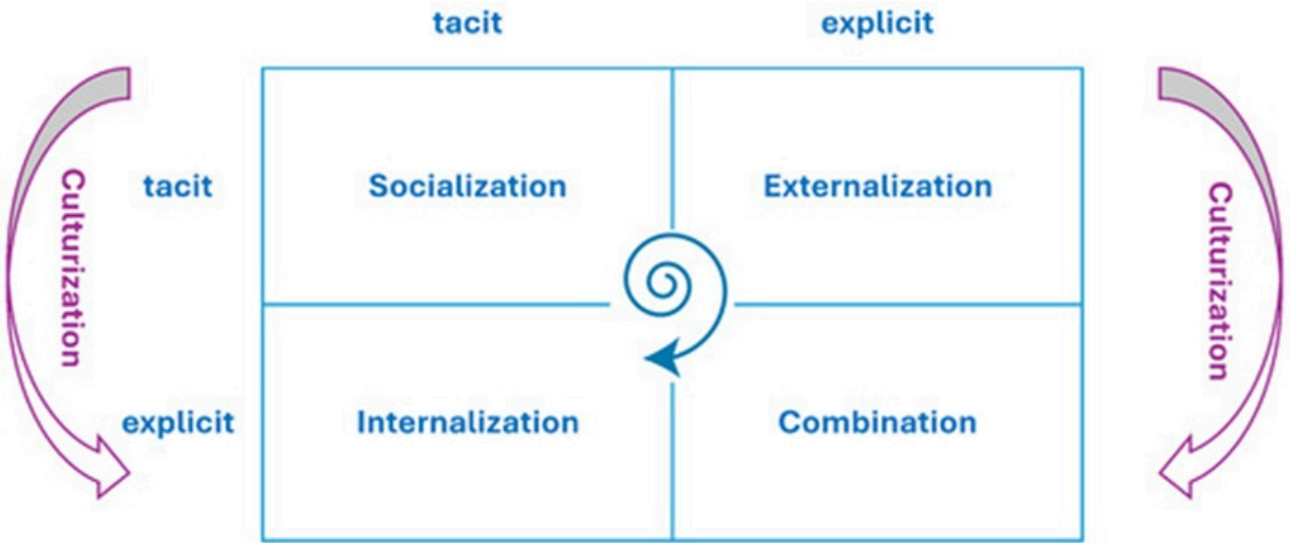


Figure 2. Suggested universal concept of Nonaka’s SECI model.

Lastly, the architecture incorporates a software application designed for high accessibility and ease of use via a web-based platform. This application enables users to swiftly record and catalog lessons learned within an organized structure, ensuring they are easily searchable and shareable among all organizational stakeholders. The platform also contains a dedicated section for patterns and lessons learned, equipped with advanced information retrieval utilities that leverage keyword queries and hyperlinks to promote cohesive, interconnected navigation. It further supports the generation of a discussion and lesson backlog to be formally reviewed and validated (or dismissed) during standard meetings and CoP sessions, alongside sophisticated search mechanisms utilizing keyword indexing and specialized hyperlink associations.

By integrating these distinct elements, the framework establishes an active, cooperative ecosystem that maximizes how knowledge is generated, disseminated, and applied within enterprises. Operating within the Scrum framework, the architecture assigns clear roles and duties to guarantee seamless execution. The Scrum Master functions as a KM facilitator, guiding the team toward objectives and advocating for the integration of best practices. The Scrum Team comprises all participants engaged in knowledge generation, collaborating to build content and assist in its verification. Meanwhile, CoP members bear the responsibility of validating and circulating these lessons learned throughout the firm, serving as a critical link to ensure knowledge is embedded into corporate operations and communicated across all organizational tiers.

A simplified overview of the used nomenclature is shown in **Table 1**. This nomenclature is useful when defining and managing roles in the requirements elicitation phase.

Table 1. Scrum framework nomenclature.

Scrum Terminology	Description
Scrum Team	A group comprising multiple roles, including the Scrum Master, Community of Practice (CoP) Member, and other related positions.
Scrum Master	Individual responsible for leading and facilitating the team.
CoP Member	Person serving in the role of unit leader.

The introduced architecture is structured around an integrated technological and methodological strategy, facilitating the explainable, AI-powered optimization of corporate workflows. It is constructed upon three primary technological modules—the AI Explainability Module, the Innovative HCI Module, and the Explainable Workflow Designer—which are sustained by three methodological cornerstones: a process modeling and optimization methodology, an HCI methodology, and an ethical XAI methodology. **Figure 3** provides a functional depiction that clarifies the underlying logic of the framework.

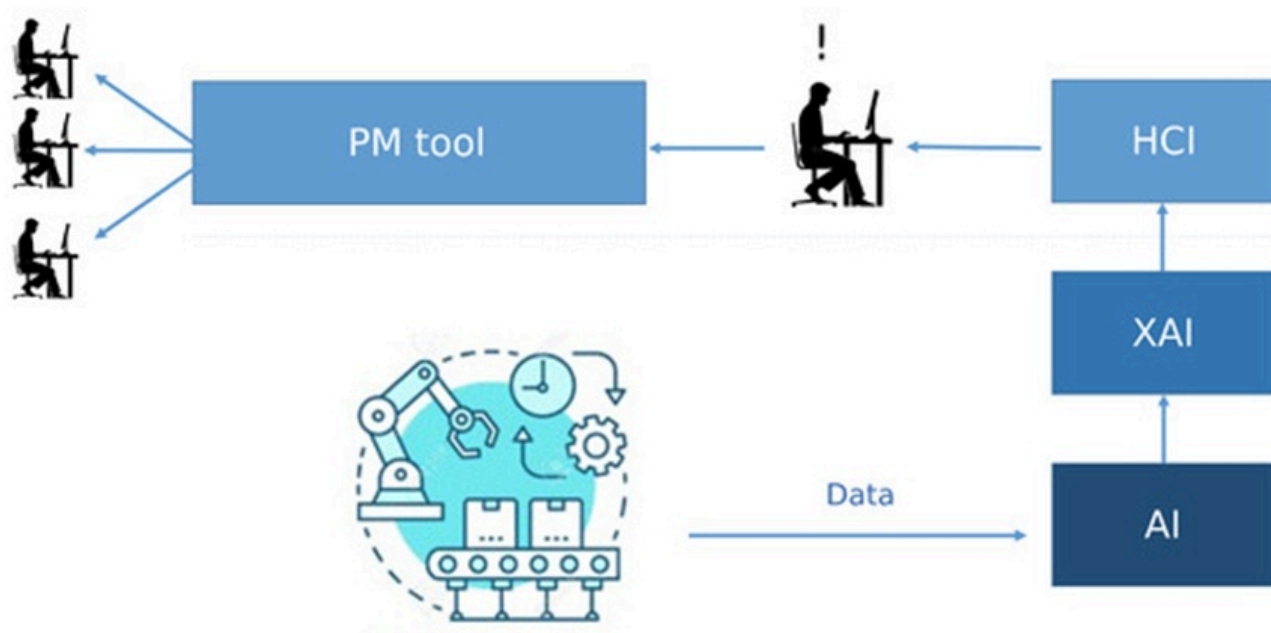


Figure 3. Framework functional representation.

The methodological process unfolds through the following key steps:

- Process Modeling and Formalization:** The initial stage centers on outlining the operational process flow via the Explainable Workflow Designer. This component permits domain specialists to structure business operations using a tailored, intuitive notation that bypasses the complexities inherent in BPMN 2.0. Utilizing a drag-and-drop interface alongside domain-specific semantics, non-technical stakeholders can formalize operational actors, information flows, and decision thresholds. This strategy not only accommodates preliminary process mapping but also streamlines subsequent system orchestration and automation.
- Data Analysis and Predictive Modeling:** Following process definition, the AI Explainability Module is deployed to execute predictive analytics via artificial intelligence models. The platform aggregates both structured and unstructured data streams (such as sensor outputs, maintenance logs, and production records) and utilizes machine learning methodologies to project critical operational metrics. For instance, it facilitates the calculation of the Remaining Useful Life (RUL) for mechanical parts.
- Explainability and Human-in-the-Loop Interaction:** To guarantee that AI outputs remain transparent and practically useful, interpretable XAI methodologies (such as SHAP and LIME) are integrated. These frameworks isolate and highlight the primary variables dictating specific predictions. Subsequently, the Innovative HCI Module conveys these data insights through interactive visual interfaces mapped to the cognitive frameworks of domain personnel. This Human-in-the-Loop (HITL) model allows operators to comprehend model determinations, verify automated guidance, and maintain ultimate decision-making command.
- Knowledge Codification and Execution:** Once verified, the resulting insights are structured into a knowledge repository via the Workflow Designer. Proven best practices and operational optimization criteria are maintained within a Process Optimization Rules database, encouraging ongoing knowledge sharing and reuse. Merging formalized procedural knowledge with active operational data enables real-time oversight and flexible decision-making.
- Ethical Evaluation:** The ethical XAI methodology directs organizations in benchmarking the ethical compliance of AI architectures against established EU core principles, such as transparency, technical robustness, data privacy, and societal welfare. Employing a structured matrix of indicators and visualization mechanisms (such as Kiviat diagrams), the framework delivers a quantifiable assessment of AI systems to guarantee responsible deployment.

The proposed framework implements an architectural strategy centered around functional modules designed to operate autonomously while maintaining real-time communication capabilities with all other system components. This specific design was selected because such modular architectures are standard practice for securing superior performance, system reliability, and scalability. Furthermore, the event-driven logical foundation of this architectural framework is distinguished by a distributed microservices paradigm. This design model facilitates the definition and creation of highly manageable, loosely coupled, and independently deployable technical components.

Significant emphasis has been directed toward the component tasked with executing business logic. Within this specific layer, the architecture establishes a Business Process Engine (BPE), operational support repositories, an array of components dedicated to AI-driven methodologies, and specialized modules for data gathering and transformation. Additionally, a Middleware Service Bus (MiSB) has been incorporated to guarantee the seamless operation of the execution engine, alongside domain-specific contextualization and interoperability.

Regarding the application layer, the logical–functional module governing user interaction is outfitted with methodologies and tools intended to simplify and streamline tasks throughout both the formulation and execution phases of a business process. This functionality is augmented by real-time messaging, which facilitates the transmission and distribution of corporate knowledge generated by optimization logic. More precisely, the technological utilities and instruments built within the project architecture utilize information models and content structured into aggregated, organized formats through advanced UI/UX principles. This approach guarantees optimal visualization and usability, reinforced by stringent authentication and authorization protocols.

The capabilities of this enabling platform allow users to govern a business process by cooperatively and interactively defining its traits, rules, and parameters, thereby fostering continuous knowledge enhancement across the process lifecycle. This setup streamlines process design, execution, orchestration, and monitoring, while aiding the identification of operational strengths and weaknesses for optimization. Another core characteristic of the framework is the creation of abstract logic components that permit the definition and subsequent deployment of user interface models via low-code/no-code strategies. These components aim to generate adaptive interfaces through an arrangement of graphical elements incorporating standard web components. Lastly, to achieve an all-inclusive and holistic design, a collection of dynamic connectors was formulated. These elements support interface configuration from an experience design perspective, permitting the capture, visualization, and management of technical data across real-time, near real-time, and historical frameworks. Correspondingly, strategies were implemented to construct technological modules and components tailored to sustain and promote more productive human–AI collaboration.

In tandem with the formulation of the described logical-functional architecture, an assessment was performed on prominent tools and libraries that support process modeling and execution, including Camunda, Power Automate, SAP Workflow Service, and Oracle BPM Suite. This evaluation indicated that Camunda facilitates event-based modeling utilizing the BPMN 2.0 standard and offers more straightforward integration with diverse IT environments. Another highly critical phase involved examining various frameworks, software libraries, and technology solutions frequently deployed in Machine Learning (ML) and Deep Learning applications. Their fundamental attributes, capabilities, advantages, and limitations were analyzed. Initial investigations focused on ML libraries and frameworks that operate independently of specialized hardware or infrastructure, aiming to simplify complex data analysis tasks and deliver integrated environments outside conventional programming languages. Subsequently, the evaluation transitioned to primary libraries designed to handle massive data volumes requiring high-performance computing capabilities.

The proposed knowledge management framework operates within a more extensive technological and methodological strategy comprising both theoretical and technical elements. Specifically, the ethical XAI methodology facilitates the evaluation of fairness and ethical compliance in AI techniques and algorithms. This assessment relies on distinct properties, metric scales, and KPIs extracted from an analysis of the relevant academic literature. More precisely, the ethical XAI methodology was formulated in alignment with legislative mandates and guidelines intended to encourage the deployment of ethical AI systems. In particular, the primary attributes utilized for this appraisal correspond to the core

requirements for ethical AI established by the European Commission [35, 36]: human oversight, robustness and safety, privacy and data governance, transparency, diversity and non-discrimination, social and environmental well-being, and accountability. For each attribute, it was possible to outline reference dimensions and measurement units necessary for objective, quantitative validation.

From the standpoint of algorithms, training protocols, and validation methodologies, predictive maintenance strategies within the manufacturing sector were adopted. This involved a data preprocessing stage featuring feature scaling and a partition into training, validation, and testing subsets using a 70%–20%–10% distribution ratio. The selected AI architectures (trained for the regression objective of 'predicting the numerical value of the remaining life of a machine') consist of the following:

- **An MLP deep neural network:** A straightforward neural network comprising approximately 100,000 parameters with fully connected ("dense") layers. This model was selected as a representative of neural network architectures (which currently constitute the most opaque AI algorithms), while maintaining a simplified structure to minimize training time via the Keras Python library.
- **A Random Forest:** This AI algorithm governs the training of 200 decision trees that collectively converge on a single prediction. In contrast to the neural network, Random Forest represents a medium-opacity algorithm and was selected to allow XAI methods to extract outcomes more straightforwardly, facilitating a direct comparison with the neural network's performance using Python's Scikit-learn library.

Two distinct XAI methods were implemented on these AI architectures (as detailed below). Both are model-agnostic (meaning they function independently of any specific AI architecture), enabling their application to both models while drawing from equivalent explanatory capabilities to ensure an equitable comparison of the generated outcomes:

- **Local Interpretable Model-Agnostic Explanations (LIME):** An XAI approach categorized under explanations by simplification, which leverages less complex surrogate models to approximate the intricate model and extract interpretable insights. Beyond its model-agnostic nature, LIME was selected due to its straightforward implementation and widespread adoption within scientific literature.
- **SHapley Additive exPlanations (SHAP):** An XAI approach belonging to the feature importance explanation category, where input samples are perturbed to calculate an index (known as a Shapley value) for each input, representing the measure of influence that a specific feature exerts on the final output (utilizing the KernelExplainer technique). SHAP was chosen because it shares thematic foundations with LIME (ensuring comparable explanatory potential) while offering an analytical contrast to LIME due to its unique methodological approach to generating explanations.

Furthermore, the deployment of process mining proved essential for extracting vital operational insights. Specifically, utilizing an event log—the historical archive of execution entries for process activities—process mining strategies were leveraged to map out workflows (Process Discovery), monitor execution pathways (Conformance Checking), and uncover systemic bottlenecks (Performance Analysis). The diagnostic insights achieved through these process mining methods yielded an exhaustive overview of the operational status quo, emphasizing the necessity of a subsequent phase to fulfill the overarching objectives of process optimization.

Hardware and software tools

To attain the documented experimental outcomes, the identification of appropriate supporting instruments was essential. The selection strategy focused carefully on optimizing system performance, long-term maintainability, and scalability. For process formalization, modeling, and execution, Camunda v.7.19 was chosen as the optimal solution. The business logic infrastructure was constructed using Java v.17, which provided a robust backend framework, streamlined system integration, and efficient processing of core application features. For the human–computer interaction (HCI) layer, AngularJS v.13 was selected to deliver an interactive, intuitive user interface.

Python v.3.11.1 was utilized to build the analytical modules, accommodating sophisticated data analysis tasks. Structured and unstructured data handling and version control were managed via PostgreSQL v.15 and MongoDB v.6, respectively, ensuring high operational reliability and persistent contextual storage. ActiveMQ v.5.17.0 was integrated to handle event-driven message queuing. Lastly, process definition and mapping adhered strictly to the BPMN 2.0 standard, utilizing the academic edition of the SAP Signavio Process Manager tool to ensure that all process diagrams complied with widely recognized industrial benchmarks.

Knowledge management system requirements

The KM framework encompasses 18 fundamental requirements, each designed to safeguard the efficacy, traceability, and enhancement of organizational knowledge assets. These technical specifications delineate the features required to sustain the entire knowledge lifecycle, spanning experiential data collection and dissemination, the formalization of corporate best practices, and the administration of organizational roles and cross-functional teams. The ultimate goal of the platform is to stimulate collaborative efforts among users, streamline knowledge dissemination, and guarantee the highly structured and systemic management of institutional information.

A comprehensive summary of these knowledge management specifications is outlined in [Table 2](#).

Table 2. Knowledge management system requirements.

ID	Requirement Name	Description
1	System Scope	Supports the collection, preservation, and presentation of information across its entire lifecycle. Includes capabilities for managing discussions, lessons learned, and patterns, while ensuring traceability of roles and groups and providing centralized pattern access.
2	User Account Administration	Provides account management features with role assignments determined by the user's position within the team or group.
3	User Authentication	Enables user verification through a standard authentication mechanism that complies with organizational security requirements.
4	Scrum Team Role Definition	Provides Scrum Team Members with capabilities to start discussions, document lessons, contribute to ongoing discussions, and suggest enhancements.
5	Scrum Master Capabilities	Extends Scrum Team Member functions by allowing approval of discussions, documentation of lessons, validation of lessons learned, and refinement of patterns.
6	Scrum Meeting Member Capabilities	Allows the Scrum Master to assume additional responsibilities for participation in SoS meetings.
7	CoP Member Capabilities	Supports involvement in validating lessons learned, patterns, and best practices. Includes all functionalities available to the Scrum Master.
8	Discussion Initiation and Documentation	Enables the Scrum Master to launch discussions and formally record new knowledge using a title, keywords, and descriptive information.
9	Lesson Documentation	Allows a user, or a user at the request of a team leader, to formalize lessons by entering a name, keywords, applicable areas, and a description. Related discussions are linked automatically.
10	Lesson Repository Management	Maintains a repository of documented lessons that can be viewed and modified by all users.

11	Lesson Learned Creation	Supports the generation of a lesson learned from an existing documented lesson or pattern, with relevant information inherited from the originating lesson.
12	Lesson Learned Validation	Facilitates validation by the SoS through the addition of KPIs and other necessary information required for approval.
13	Lessons Learned Repository Management	Provides a repository for lessons learned that is accessible to all users for viewing and continuous improvement.
14	Pattern Validation	Enables the CoP to validate best practices, requiring supplementary information and adherence to specific constraints.
15	Pattern Repository Management	Organizes content according to ID, category, and topic, with advanced metadata-based search capabilities.
16	Content Classification	Categorizes information by type (discussion, lesson, lesson learned, pattern), subject area, and unique ID, while supporting searches using tags, usage scope, and creation date.
17	Data Import and Export	Provides import and export capabilities for lessons, lessons learned, and patterns.

Each individual requirement is elaborated in depth throughout the subsequent technical subsections.

Requirement ID 1: System Scope

The KM platform must capture all relevant data linked to the KM workflow, store this information securely and permanently, and permit real-time visualization at any phase of its lifecycle and across every stage of the project management cycle. Specifically, the software architecture must support the following capabilities:

- **Initiate a discussion:** Enable users to log operational issues and submit recommendations regarding daily tasks, fostering collaborative problem-solving and peer discussion. The accumulated and categorized logs can populate an operational backlog administered directly by the Scrum Team (serving tacit KM objectives during the socialization phase).
- **Define a Lesson:** Allow practitioners to formalize an experiential insight and critique it collaboratively. This repository of documented lessons serves as a localized topic backlog accessible across all Scrum Teams during active operations (facilitating the transition from tacit to explicit KM during the externalization phase).
- **Define a lesson learned:** Provide a dedicated system function through which verified lessons learned are formalized and disseminated. The gathered records constitute a high-level backlog evaluated during Scrum of Scrums (SoS) sessions, which are attended by individual Scrum Masters or team coordinators.
- **Formalize a pattern:** Enable users to structure and distribute verified corporate best practices. This database of validated patterns creates an overarching topic backlog intended for review and debate within a Community of Practice (CoP).
- **Manage roles and groups:** Accommodate the structural reality that corporate teams comprise individuals with distinct operational profiles. Within the IT domain, for instance, a representative group configuration might include a developer, a quality assurance tester, a team lead, and a business unit supervisor.
- **Track historical changes:** Maintain an immutable audit trail for every logged discussion, lesson, lesson learned, and established best practice.
- **Ensure archive accessibility:** Guarantee that the repository of validated patterns remains searchable for all authorized personnel, encouraging the transmission of explicit knowledge back into tacit individual competency.

during the internalization phase.

Requirement ID 2—Account Management

The system must establish distinct user account categories configured according to the specific role and access level assigned to the individual within their respective organizational unit.

Requirement ID 3—System Authentication

Authorized personnel must log into the application using a standard credential-based verification mechanism. This portal must align with corporate cybersecurity standards, requiring the validation of an approved username and password pair.

Requirement ID 4—Definition of role in the scrum team

The system architecture must permit any user designated as an active member of the Scrum Team to execute the following core tasks:

- Open a new discussion thread dedicated to resolving a technical bottleneck or evaluating a collaborative process enhancement proposal;
- Document and formalize an individual lesson;
- Contribute actively to ongoing team discussions;
- Recommend specific refinements to an active discussion thread.

Requirement ID 5—scrum master functionality

The platform must provide specialized privileges to users holding team leadership responsibilities within a Scrum Team (i.e., the Scrum Master). In addition to inheriting every capability available to standard Scrum Team Members, these coordinators must be empowered to:

- Formally approve a pending discussion thread;
- Document a lesson and submit it to the platform's validation workflow;
- Authorize the transition of a basic lesson into the official lessons learned repository;
- Adapt and restructure an active pattern to generate a new instructional lesson.

Requirement ID 6—Scrum member of scrum meeting functionality

The platform accommodates users serving as team coordinators within individual Scrum Teams—namely, the Scrum Master—who additionally participate in higher-level Scrum of Scrums Meetings. Consequently, this user profile automatically inherits all baseline functionalities designated for both standard Scrum Team Members and specialized Scrum Masters.

Requirement ID 7—CoP member functionality

The system design mandates that business unit leaders may join an active CoP, granting them the capability to engage in dedicated CoP deliberations. These discussions focus on validating an approved lesson learned, elevating it to the status of an official pattern, and establishing it as an authoritative corporate best practice. Users with CoP membership inherit all fundamental privileges granted to a Scrum Master.

Requirement ID 8—Initiation and formalization of a discussion

The platform must enable users to input technical challenges or operational recommendations connected to the Scrum Team's day-to-day activities. Every individual entry must be tracked systematically to prevent data loss. The initiator of the thread (or the managing Scrum Master) must populate the following data fields:

- Title of the discussion thread;
- Official creation date of the discussion.

Upon resolution, the Scrum Master can choose to close and archive the thread. Alternatively, if the deliberation yields insights deemed critical for organizational knowledge creation, the user can formalize the entry by inputting the following parameters:

- An explicit title detailing the proposed knowledge asset;
- Relevant indexing keywords (metadata tags);
- A comprehensive technical description of the proposed knowledge.

Requirement ID 9—Lesson formalization

Any practitioner within the Scrum Team who has acquired shareable experience—regardless of whether the outcome was successful or problematic—can formalize a lesson. This workflow can also be triggered when a team lead requests the conversion of an active discussion thread. The individual generating the lesson entry is required to supply:

- The unique name of the lesson;
- Relevant indexing keywords (metadata tags);
- The specific operational domains where the lesson applies;
- A descriptive overview detailing the core lesson content.

If the entry is generated directly from an existing discussion thread, the system will automatically pre-populate the corresponding fields using data extracted from that source discussion.

Requirement ID 10—Lesson management

Following validation and storage within a designated knowledge repository, the system ensures the lesson becomes visible to the entire user base. This availability permits subsequent peer commentary and discussion by team members seeking to further refine the material.

Requirement ID 11—Creation of the lesson learned

A formal lesson learned entry can be initiated from a baseline lesson previously logged by a Scrum Master or an active Scrum Team Member. Alternatively, it can be generated by modifying an established pattern in a manner that requires a fresh round of system validation.

Requirement ID 12—Approval of lesson learned

The verification and validation of a lesson learned are conducted by the SoS panel. Because a lesson learned is derived from an antecedent lesson or pattern, it natively inherits its baseline metadata. Consequently, the user managing the validation process must verify and input specific parameters, a portion of which are auto-populated from the parent asset:

- Title designating the lesson learned;
- Relevant indexing keywords (metadata tags);
- Official validation date of the lesson learned (stamped automatically by the application);
- The intended operational application domains;
- A descriptive narrative of the lesson;
- The validation and mapping of corresponding Key Performance Indicators (KPIs).

Requirement ID 13—Managing Lesson learned

Following its verification and preservation in the designated data store, the platform ensures that the lesson learned remains visible to the entire user community. This accessibility allows individuals to engage in subsequent commentary and exchange views to further refine and enhance the recorded asset.

Requirement ID 14—Pattern approval

The validation and formalization of an organizational best practice are carried out under the purview of the CoP. Because a best practice originates directly from a verified lesson learned, the team coordinator must populate the system with the identical baseline data fields required for lessons learned, supplemented by specialized inputs detailing any operational constraints or boundaries that must be considered.

Requirement ID 15—Pattern management

Once an architectural Pattern achieves validated status, the system exposes it to all system users, offering them functionality to submit optimization feedback, request further analysis, or suggest structural modifications. Concurrently, the platform maintains a comprehensive traceability matrix that maps the underlying connections linking Best Practices to baseline Lessons, Lessons Learned, corresponding corrective actions, and related dependencies.

Requirement ID 16—Content categorization

All system assets must be rigorously classified according to both their structural category (comprising discussions, lessons, lessons learned, and patterns) and their specific thematic domain. Each distinct piece of content is designated

by an immutable, unique identifier. The platform equips all authorized users with search mechanisms to locate material using this unique ID or via combinations of metadata parameters, including keywords (tags), asset type, operational field of use, and date of creation. The interface subsequently renders a list of content matching the selected metadata criteria.

Requirement ID 17—Import/Export

The technical specifications dictate that the system architecture must facilitate the smooth ingestion and migration of generated lessons, lessons learned, and knowledge patterns. **Table 2** provides an aggregate architectural summary of the aforementioned requirements along with a concise functional breakdown of each.

Implementation

The following section delineates the formal construction of a process model tailored specifically for organizational knowledge creation. The configuration of this model is grounded in the methodological framework pioneered by Yakima. This specific approach establishes the mechanisms through which KM transitions across the distinct sequential phases of the SECI model within an active Agile operational workflow. The four core SECI phases are operationalized within the process design as follows:

- **Socialization (Tacit to Tacit):** In this foundational stage, implicit personal knowledge is exchanged informally among Scrum Team Members through direct interpersonal collaboration, co-working dynamics, and the hands-on sharing of practical experience. This occurs naturally within established Agile ceremonies—such as daily stand-up meetings, sprint reviews, and team retrospectives—where operational hurdles, technical breakthroughs, and nascent solutions are discussed transparently. Although formal validation mechanisms are absent during this initial contact, it serves as the essential incubator for uncovering raw insights that may mature into documented lessons.
- **Externalization (Tacit to Explicit):** Internalized tacit insights are translated into codified, explicit knowledge through structured verbalization and documentation. This phase manifests directly within Discussion Management workflows, where the Scrum Team—typically facilitated by the Scrum Master—records emerging knowledge in the form of written documents, technical presentations, or structured textual logs within shared digital repositories. While these submissions have not yet undergone formal validation, they are systematically indexed and tracked, laying the groundwork for subsequent evaluation. Furthermore, the Scrum Master actively guides the documentation process, ensuring that critical findings are systematically scrutinized during retrospective meetings and Scrum of Scrums sessions, thereby promoting cross-team validation and the spread of new perspectives.
- **Combination (Explicit to Explicit):** Throughout this phase, separate items of explicit knowledge are systematically aggregated, cross-referenced, and synthesized to produce highly structured corporate intelligence. This stage forms the core operational pipeline for the formalization, verification, and transformation of loose lessons into reusable, systemic patterns. Initial lesson insights are entered via a standardized submission form engineered to guarantee data consistency and completeness. The subsequent validation pathway diverges into two parallel streams based on the underlying evidence supporting the lesson. When an entry is backed by objective quantitative metrics, such as specific KPIs, it is routed through an automated Digital Knowledge Engine (DKE) for validation. Conversely, when a lesson is derived from qualitative observations, its organizational relevance and applicability are assessed during SoS assemblies, which provide a collaborative environment for expert appraisal and peer review. Once a lesson is deemed universally applicable across equivalent operational settings, it is elevated to a pattern. These patterns are then transferred to the appropriate CoP for deeper refinement and formal authorization. The CoP bears responsibility for verifying the scalability, structural integrity, and overall caliber of each candidate pattern, which includes establishing explicit input and output parameters, defining operational constraints, and estimating the projected impact on performance benchmarks (KPIs). Beyond validation, CoPs function as organizational hubs for

disseminating approved patterns and best practices, encouraging widespread reuse by organizing focused technical workshops, maintaining centralized repositories, and mentoring project teams on adapting these knowledge assets to their unique project environments. Taken as a whole, this phase acts as a cornerstone of institutional learning, transforming isolated personal or team experiences into standardized, actionable knowledge capital that guides future project design and optimizes corporate decision-making.

- **Internalization (Explicit to Tacit):** In the final phase, explicit organizational knowledge is absorbed by individual practitioners through direct, practical application within their daily assignments. Operational teams integrate the validated best practices or structural patterns into their day-to-day activities, contextualizing them as required by current demands. The practical feedback loops generated by this real-world execution frequently spark fresh individual insights, which, when relevant to the broader organization, restart the SECI continuum by looping back to the initial socialization stage.

Nevertheless, as highlighted previously, the baseline methodology fails to pinpoint the exact operational junctures within the Scrum framework where raw knowledge crystallizes into formal lessons learned and corporate best practices. It also lacks a precise role definition for the various participants operating along the organizational KM pipeline.

The process model introduced herein is explicitly engineered to address these structural ambiguities, thereby representing a novel evolutionary step in process engineering. The subsequent sections detail the macro-level workflow designed to govern the formulation of lessons learned and patterns recognized as authoritative corporate best practices.

To map this workflow, BPMN 2.0 notation was employed via the Signavio Process Manager platform, utilizing an academic licensing model.

Signavio Process Manager provides an intuitive Business Process Management (BPM) environment designed to let users map out, configure, and optimize their operational workflows. The software facilitates the automated generation of comprehensive user manuals derived from the modeled business processes, offering granular documentation for the execution of discrete activities and illuminating the logic behind organizational decision-making paths.

This system capability is of paramount importance given the core objective of this study: establishing a definitive process model for creating and distributing knowledge within highly complex operational ecosystems.

Overview of the process

Owing to the significant size and complexity of the resulting process architecture, we have elected to first present a macro-level perspective of the overarching framework, as illustrated in **Figure 4**, prior to delivering a granular breakdown of the separate constituent subprocesses and their respective tasks.

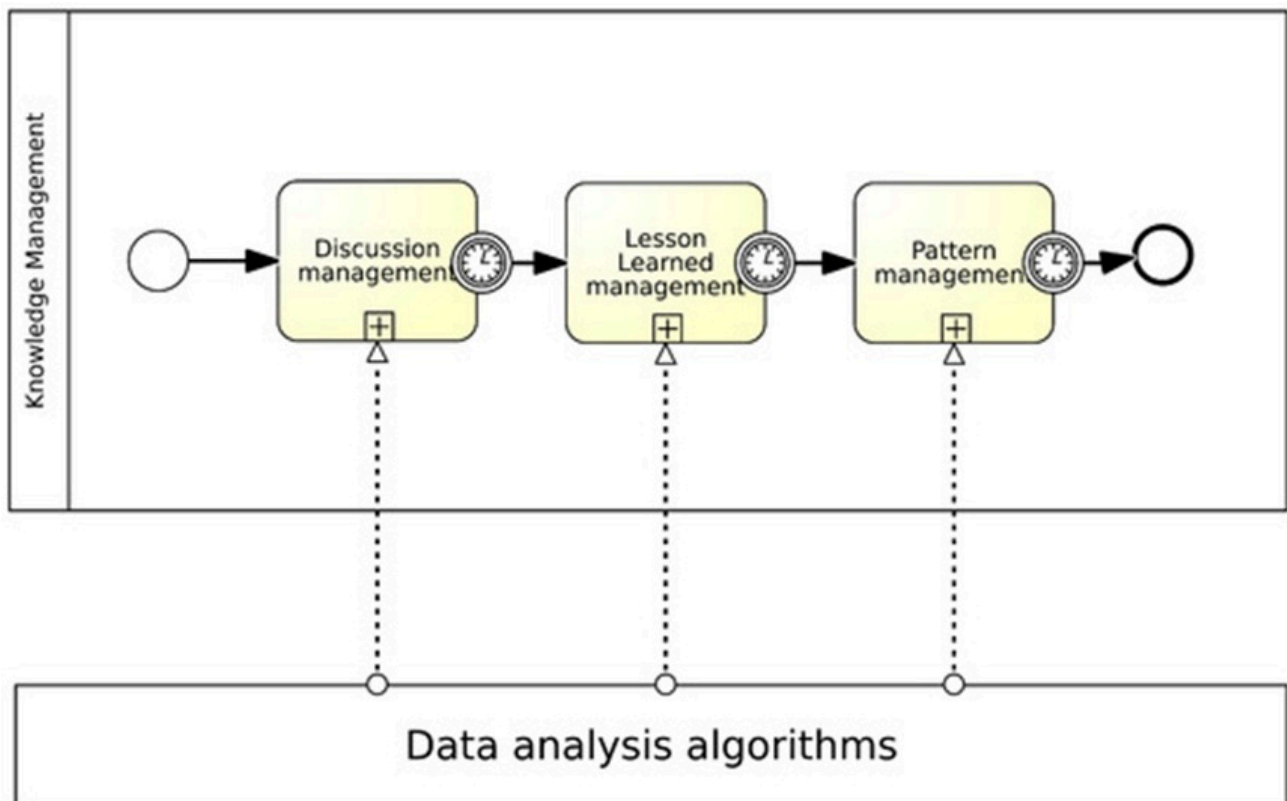


Figure 4. Overview of the KM process.

The logical progression that leads to the establishment of a verified pattern and its subsequent deployment as an approved best practice initiates within an individual Scrum Team's discussion forum. Should this collaborative exchange yield novel, actionable insights, the extracted material can be advanced through the validation pipeline to become a recognized lesson learned, and ultimately elevated to an authoritative pattern representing a corporate best practice. Operating in parallel with this workflow is a automated data analytics framework that uses dedicated machine learning algorithms to isolate and suggest potential new knowledge structures directly to the human actors charged with formalization and validation.

An essential characteristic of these combined subprocesses is their compressed operational duration, which underscores their close alignment with the foundational principles of Agile methodologies established in prior research.

Discussion management

The subsequent section directs attention to the comprehensive operational breakdown of the Discussion Management workflow, with its sequential structure visually modeled in [Figure 5](#).

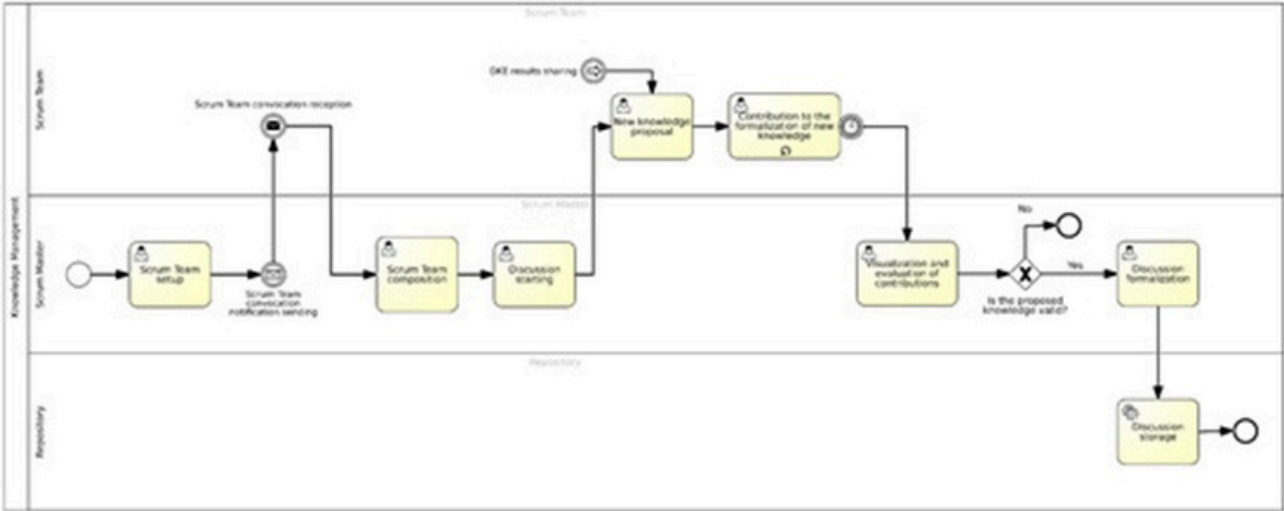


Figure 5. Discussion Management process.

To interpret this lifecycle accurately, an foundational premise must be established: the socialization sequence defined within the SECI matrix by Yakima [2] is not directly diagrammed via BPMN 2.0 notation, as this interactive phase occurs entirely upstream of the process modeled here. Furthermore, the Scrum Team referenced across this architecture is assembled dynamically; its formation and membership aggregation are executed responsively to satisfy specific functional challenges using the diverse skill sets of its participants. These baseline conditions allow for the presentation of the organizational knowledge taxonomy outlined in Table 3:

Table 3. Knowledge taxonomy.

Term	Description
Discussion	The process of articulating and sharing newly generated knowledge that arises during the execution of operational activities.
Lesson	A potentially useful experience that has not yet been fully implemented, validated, or adopted by others and remains under formal development and documentation.
Lesson Learned	A documented recommendation, guideline, or checklist that captures successful practices and identified shortcomings from a specific event or experience.
Best Practice	A systematically applied and formally documented approach that consistently contributes to the achievement of superior outcomes.
Pattern	A proven model supported by quantitative evidence demonstrating that its application reduces time, effort, and costs, enhances quality and customer satisfaction, or positively influences at least one of these performance factors.

Once the Scrum Master issues a formal summons to the designated personnel, they initiate an active discussion thread. During this session, select members of the Scrum Team submit raw knowledge proposals derived from notable experiences encountered throughout their day-to-day operations. These proposals can also emerge from sharing empirical outcomes observed by individual practitioners or organizational peers during earlier knowledge assessment cycles. The remaining participants joined in the session provide iterative contributions to enhance the initial presentation and clarify the boundaries of the proposed knowledge asset.

Following a sequence of collaborative refinements, the Scrum Master conducts an appraisal of the intelligence generated during the session. Upon confirming the viability of the proposal, the coordinator codifies the validated knowledge and its underlying arguments into a standard distributable format (such as a word processing document or a slide presentation). The resulting file is subsequently committed to the Best Practice Warehouse (BPWH) under the classification label “Discussion”.

The dedicated system interface requires the population of the following metadata fields:

- **Title of the discussion:** An explicit designation identifying the core focus of the thread.
- **Description:** A text field providing a summary of the conceptualized knowledge.
- **Attachment:** A digital file upload slot accepting doc, ppt, or pdf formats, which contains the granular technical breakdown of the newly externalized insight.

The following discrete tasks are vital for a comprehensive understanding of this operational sequence:

- **Configuring the Scrum Team:** In this phase, the Scrum Master identifies and coordinates the essential personnel profiles required to form the specialized working unit. Once these specific contributors are identified, the coordinator distributes a notification or email alert to convene the group.
- **Composing the Scrum Team:** Following the selection of candidate profiles, the Scrum Master formally organizes the functional Scrum Team, effectively grouping the personnel who will steer the creation, refinement, and administration of new knowledge assets.
- **Starting the Discussion:** With the team configuration finalized and mobilized, the Scrum Master launches the discussion forum. Here, team members introduce emergent insights discovered during the operational activities of a specific sprint cycle. The interface presents an initialization form requiring:
 - Title of the discussion;
 - Timestamp for the thread's initiation;
 - A concise abstract detailing the core problem or idea.
- **Proposing New Knowledge:** During this task, a team practitioner introduces and explains a new concept to their peers, such as a novel workflow, a troubleshooting methodology deployed to bypass a technical bottleneck, or a design choice discovered while developing a specific feature during recent sprints. Alternatively, a participant may present findings derived from an automated assessment of an asset they developed, which was previously passed through an evaluation matrix called the Data KPI Engine (DKE)—a subsystem engineered to analyze data arrays via selected KPIs to extract novel operational insights.
- **Contributing to the Formalization of New Knowledge:** Scrum Team Members collaboratively refine the documentation of the newly introduced concept. Rather than concluding within a single operation, this activity cycles iteratively until a unified, verified proposal is achieved. This task operates under a strict time window within which the documentation must be optimized and finalized. A dedicated chat interface allows team members to submit real-time comments, text revisions, and constructive critiques directly to the asset draft.
- **Viewing and Evaluating Contributions:** The Scrum Master reviews and critiques the compiled peer suggestions and the baseline knowledge presented during the session. This represents a critical gating phase where the knowledge asset is either validated—allowing it to advance to subsequent stages of the KM lifecycle—or permanently archived as discarded material.
- **Formalizing the Discussion:** The Scrum Master transforms the finalized conversation into a standardized format optimized for cataloging inside the BPWH, populating fields including:
 - *Title of the Discussion:* A descriptive title capturing the essence of the validated insight;
 - *Date:* The exact timestamp of formalization, providing a temporal anchor that simplifies future database queries;

- **Keywords (Tags):** Searchable index identifiers that optimize the indexing and retrieval of the proposed knowledge;
- **Description (Text Area):** A concise synthesis summarizing the primary technical properties of the validated asset.
- **Archiving the Discussion:** Following formalization by the Scrum Master, the file is transferred to a dedicated digital repository. This repository functions as a comprehensive backlog of historical discussions from which future formal knowledge proposals can be extracted and validated.

Lesson learned management

Once an item of knowledge is formalized and indexed as a discussion, the subsequent sequence involves its transformation into a formal lesson and its analytical evaluation to determine if it warrants promotion to a verified lesson learned. The mechanics governing these phases are detailed in the Lesson Learned Management workflow illustrated in **Figure 6**.

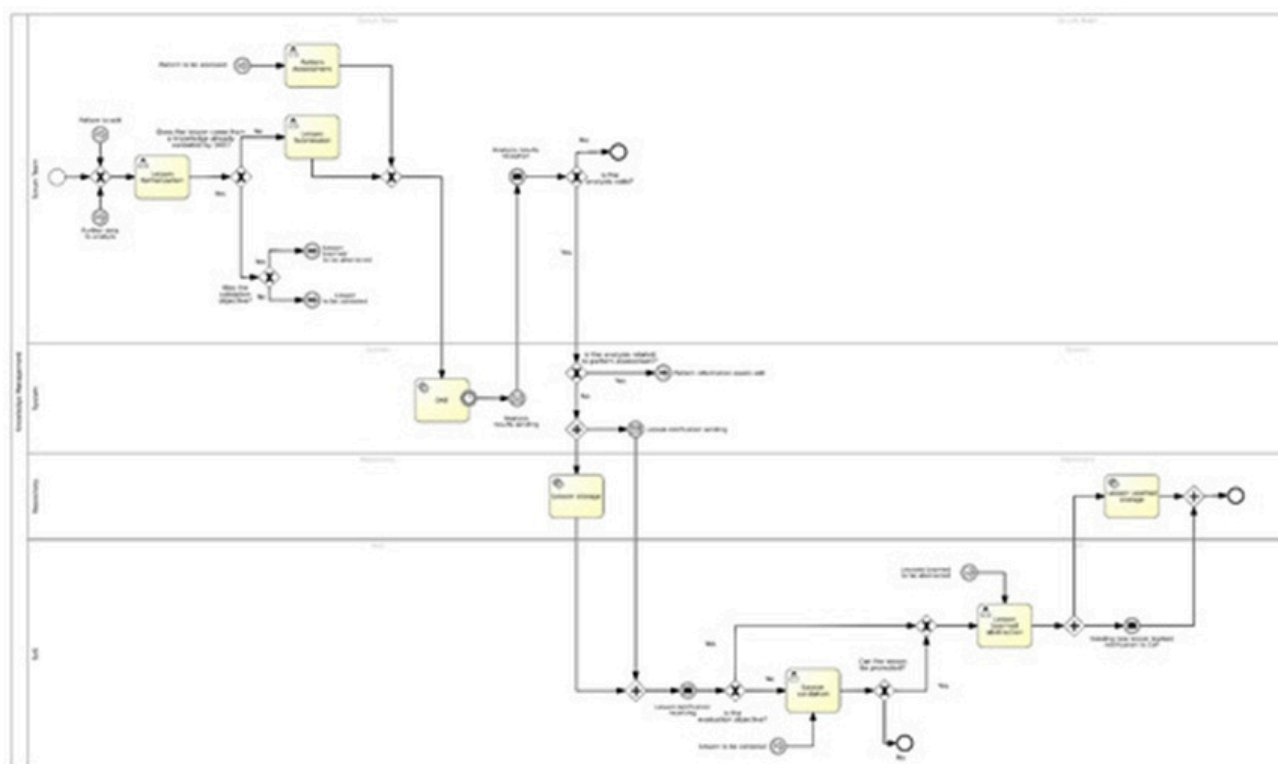


Figure 6. Lesson learned management process.

The Scrum Team (or an authorized user within the unit) initiates the transformation of the discussion into a lesson. This is achieved during the “Lesson Formalization” task by completing a dedicated interface form where structural information regarding the item’s technical description is submitted.

Upon saving the initial parameters, the designated team representative moves to the “Lesson Submission” task to configure additional analytical criteria, including:

- The priority hierarchy allocated to various established KPIs.
- The baseline target success rate regarding the projected outcome, which is visually represented via a RADAR chart; this benchmark establishes the minimum acceptable performance threshold achievable within a defined operational timeframe.
- The total chronological duration of the scheduled data analysis.

- The specific evaluation threshold applied to manage prioritized metrics during a subjective assessment run by the DKE.

Following the conclusion of this automated analysis, the platform transmits an alert detailing the results to all members of the Scrum Team. If the outcome satisfies the criteria, the resulting lesson is committed to the corresponding database tables within the BPWH. Conversely, if the generated outcome is flagged as “subjective”—indicating that only a subset of the preconfigured parameters were met—the asset is escalated to the Scrum of Scrums (SoS) assembly. This body functions as an cross-functional assembly uniting the organization's distinct Scrum Teams, with participation extended to external practitioners based on their specific technical mastery over the issues slated for evaluation.

Under this scenario, the SoS evaluates the borderline lesson. If the assembly confirms its validity, they oversee its abstraction into a formal lesson learned and authorize its storage in the BPWH. In contrast, if the automated analysis yields an “objective” validation, the SoS executes a formal, standardized abstraction process to finalize the lesson learned and place it inside the appropriate repository.

Once these administrative steps conclude, the SoS routes a notification to the organization's internal Communities of Practice (CoPs), directing them to conduct a deeper analysis of the newly minted knowledge asset and further abstract its properties into a reusable pattern.

As mapped within the lesson learned management process diagram, the system can accommodate an alternative operational pathway. This route is designated as a “Pattern Assessment” because it facilitates the re-evaluation of an established pattern within an operational context distinct from its original deployment environment. To execute this, the practitioner connects an appropriate testing dataset alongside the corresponding KPIs. Once these parameters are established, the analysis moves forward by replicating the identical processing tasks applied to standard lessons.

It is also critical to note that if the engine generates multiple valid, competing configurations, a system event triggers a background daemon. This process establishes a clear performance ranking among the variations and delivers an optimized configuration suggestion to the developer evaluating the pattern.

Additional alternative workflows permit direct modifications to the pattern framework following hands-on developer utilization, or the application of an entirely new dataset. Such adjustments may lead to structural alterations of the pattern's architecture guided by association rules generated by the underlying data analysis engine.

The following specific tasks are core to understanding this process layer:

- **Lesson Formalization:** The Scrum Master or a designated team practitioner pulls a previously archived discussion file and populates the specialized input form, filling out fields including:
 - Lesson title;
 - Tags;
 - Scope of use;
 - Lesson description;
 - Attachments (where applicable).

If the lesson is derived from a knowledge asset that has already undergone evaluation (and been deemed objectively or subjectively valid), the managing Scrum Master completes a series of pre-filled system fields. If the asset boasts prior objective validation, the workflow routes directly to the “Lesson Learned Abstraction” task. If it carries a prior subjective validation, the system steers the user to the “Lesson Validation” phase.

- **Lesson Submission:** Using a specialized control interface, the user defines the validation dataset alongside the target KPIs to be achieved within a designated time window. The required input fields include:

- Priority assigned to each KPI;
- Analysis duration;
- Threshold;
- Minimum success percentage.
- **DKE:** The Data KPI Engine executes the automated validation protocol on the submitted lesson. If the run yields a positive result, the file is archived in the appropriate data tier, and automated alerts are delivered simultaneously to the originating Scrum Team and the SoS assembly to initiate the secondary validation phase for promotion to a lesson learned.
- **Lesson Validation:** Members of the SoS assembly systematically inspect the analytical results produced by the DKE regarding the candidate lesson. This task can also be initiated automatically by a platform notification when a knowledge asset, previously marked as subjectively valid by a localized Scrum Team, is queued for full SoS review.
- **Lesson Learned Abstraction:** Upon confirming the empirical validity of the lesson's data outputs, the SoS panel authorizes the asset and elevates it to a formal lesson learned. The interface requires the population of the following fields:
 - *Lesson Learned Title:* A text box to input the definitive designation for the lesson learned;
 - *Tags:* A metadata field to supply keywords that index the asset, facilitating search operations across the enterprise;
 - *Scope of Use:* A field outlining the primary operational environments where the lesson learned applies;
 - *Lesson Learned Description:* A dedicated section explaining the core characteristics, mechanics, and features of the lesson learned;
 - *Achieved KPIs:* The primary performance metrics yielded by the validation of the lesson learned;
 - *Used Services:* A detailed listing of the core underlying technological services and software components leveraged;
 - *Attachments (where applicable):* Specialized uploads for PDF, PPT, or DOC files that provide deeper architectural information regarding the lesson learned's design.

The completion of this task automatically executes two systemic functions:

- It dispatches an automated notification to active CoP members so they can commence secondary evaluations to elevate the lesson learned into a best practice pattern;
- It securely stores the approved lesson learned inside the BPWH database.

Pattern management

The previously described lesson learned management sequence terminates with the distribution of an automated notification to the institutional CoP, informing them that fresh knowledge has been formalized. At this juncture, the pattern management workflow initiates. A granular depiction of this progression is illustrated in **Figure 7**.

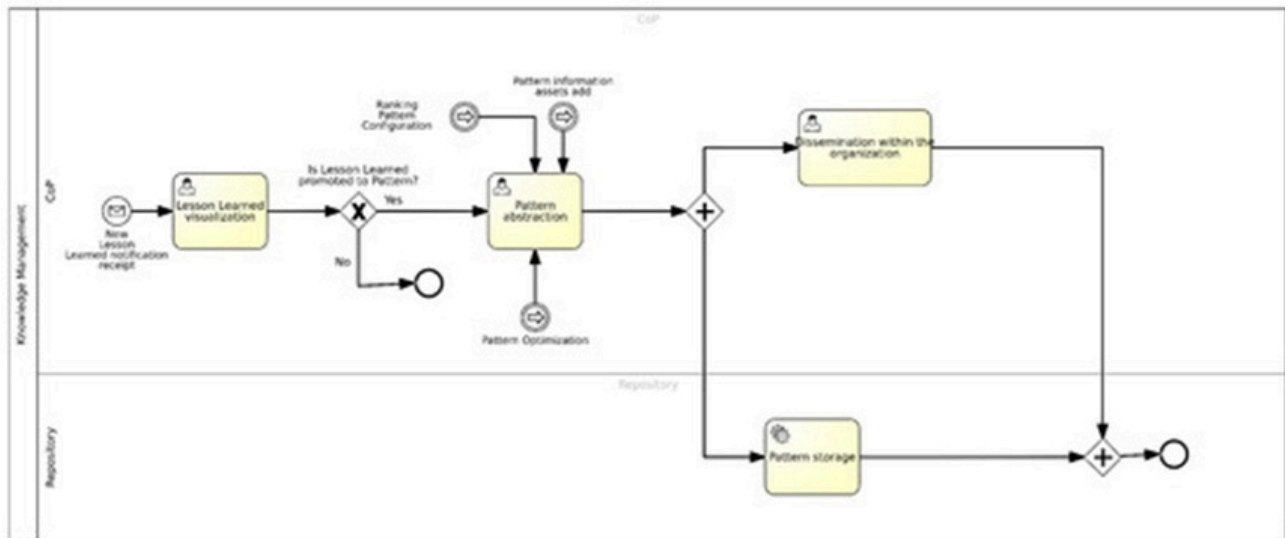


Figure 7. Pattern management process.

Upon reviewing the newly submitted lesson learned, the CoP determines whether the asset possesses the requisite attributes for elevation to an official pattern.

If this appraisal yields a favorable conclusion, the CoP abstracts and defines the pattern by completing a specialized interface form to register critical structural data. While a portion of this metadata may be inherited from the originating lesson learned, the majority is explicitly re-entered, and several new technical specifications must be constructed from scratch, including:

- Input parameters, explicitly defining their name, data type, and acceptable operational range;
- Output parameters, explicitly defining their name, data type, and acceptable operational range;
- Mathematical and logical functions designated to calculate output parameters based on the incoming input variables;
- System constraints to monitor, formulated around the input and output parameters and/or the data arrays supplied by active services;
- Application Programming Interfaces (APIs) exposed to external software systems.

Following the execution of this step, the CoP catalogs the newly minted pattern within the designated tables of the BPWH while simultaneously broadcasting it across the enterprise. This dual action ensures the asset is visible for internalization by various Scrum Teams, who can subsequently retrieve it to anchor further knowledge definition cycles. The fundamental tasks required to comprehend this process layer are outlined below:

1. **Viewing the Lesson Learned:** Members of the CoP review the validated lesson learned routed from the SoS, assessing whether the proposed solution exhibits the scalability required to serve as an enterprise-wide pattern.
2. **Abstracting the Pattern:** In this phase, the CoP determines that the analyzed lesson learned is reusable across diverse operational contexts and business domains, thereby establishing it as an official organizational pattern. Following this consensus, a specialized form is populated with the following technical entries:
 - *Pattern Title:* A text field where the defining name of the pattern is recorded;
 - *Tags:* A metadata field for entering index keywords that streamline asset discovery for other corporate members;
 - *Usage Domains:* A selection field identifying the primary operational areas where the pattern can be successfully deployed;

- *Pattern Description*: A dedicated section detailing the pattern's architectural properties, primary attributes, and core mechanics;
- *KPIs*: The primary performance metrics yielded by the pattern;
- *Services Used*: A specification of the underlying software services and technical infrastructure leveraged by the pattern;
- *Input Parameters*: A granular definition of acceptable inputs (specifying name, data type, and boundary limits);
- *Output Parameters*: A granular definition of acceptable outputs (specifying name, data type, and boundary limits);
- *Constraints*: Clear boundaries to be monitored, derived from input/output behaviors and/or data streams generated by dependent services;
- *Attachments*: Dedicated uploads for PDF, PPT, or DOC files that furnish supplementary data regarding the pattern's internal composition and technical traits.

Once the interface form is fully processed, the new knowledge asset is saved securely inside the BPWH and propagated throughout the corporate network. During this task, CoP members may evaluate supplementary informative updates, version releases, or structural optimizations recommended by the ML algorithms embedded within the data analytics subsystem.

3. **Dissemination within the organization**: The newly approved pattern, elevated to the status of an official best practice, is broadcast across the enterprise infrastructure to ensure Scrum Team practitioners can seamlessly integrate it into their ongoing software engineering and coding assignments.

Pattern usage and editing

This process encompasses the sequential activities executed by a Scrum Team practitioner to search for, isolate, and deploy an established pattern to fulfill an operational requirement. As illustrated in [Figure 8](#), the overall lifecycle unfolds through a standard sequence: the user scans the global registry of patterns stored inside the BPWH, isolates the most contextually appropriate asset, and executes a selection command.

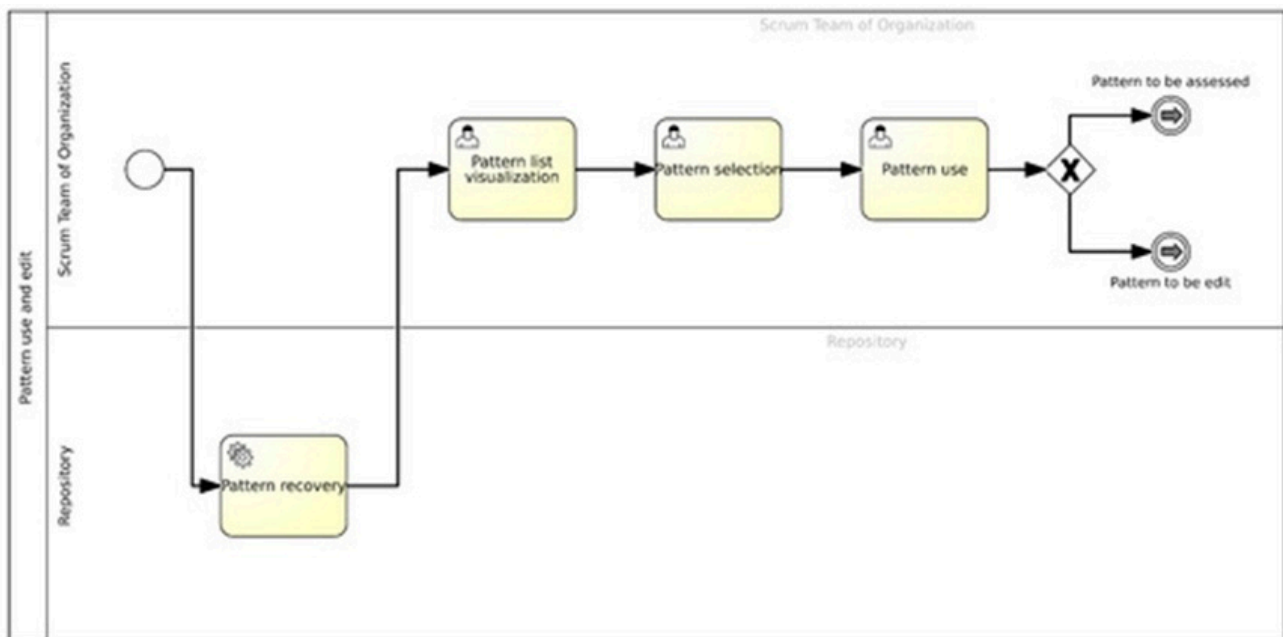


Figure 8. Pattern selection and usage process.

Following real-world deployment and the collection of resulting feedback, the practitioner can choose to alter the pattern, thereby triggering the continuous optimization cycle governing organizational knowledge assets. Initiating a pattern modification automatically activates the “Lesson Learned Management” workflow, looping the asset back to the initial phase of lesson formalization, validation, and subsequent abstraction into a lesson learned and subsequent pattern.

Concurrently, the practitioner can choose to subject a pattern to a formal assessment, measuring its performance metrics within an operational environment entirely distinct from the domain for which it was originally engineered. The empirical outputs derived from this test run serve to expand the global information matrix associated with the pattern under review.

The discrete individual tasks defining this process layer are detailed below:

- **Viewing the pattern list:** A Scrum Team practitioner queries the global directory of patterns housed in the BPWH, allowing them to inspect the distinct technical properties, configurations, and benchmarks of the available assets.
- **Selecting a pattern:** The practitioner isolates and selects the specific pattern deemed most compatible with current operational needs from the filtered results returned by the BPWH.
- **Using the pattern:** The Scrum Team member integrates and executes the selected pattern within their active environment, leveraging its pre-built capabilities.

Of significant architectural relevance are the workflows governing *patterns to be submitted for assessment* and *patterns to be modified*. These sequences connect directly back to the primary Discussion Management workflow.

Similarly critical is the pattern assessment loop, which serves as an intersection binding multiple distinct workflows. This systemic link automatically initializes a secondary branch of the lesson learned management process, routed through a dedicated task appropriately labeled “Assessment Pattern”.

Within this specific task, the pattern assessment is performed by processing a designated reference dataset to evaluate the pattern's efficiency inside a non-native operational context. If this trial run successfully satisfies all configured validation criteria, the empirical results are appended to the pattern's overarching informational profile. This directly influences the pattern formalization phase, enabling the user to input the novel analytical data generated by the cross-context experiment.

Knowledge assessment management

Within the Knowledge Assessment management workflow, dynamically mapped in **Figure 9**, an individual practitioner retains the capability to independently evaluate the viability of potential knowledge they have generated, without initiating a formal documentation or approval track.

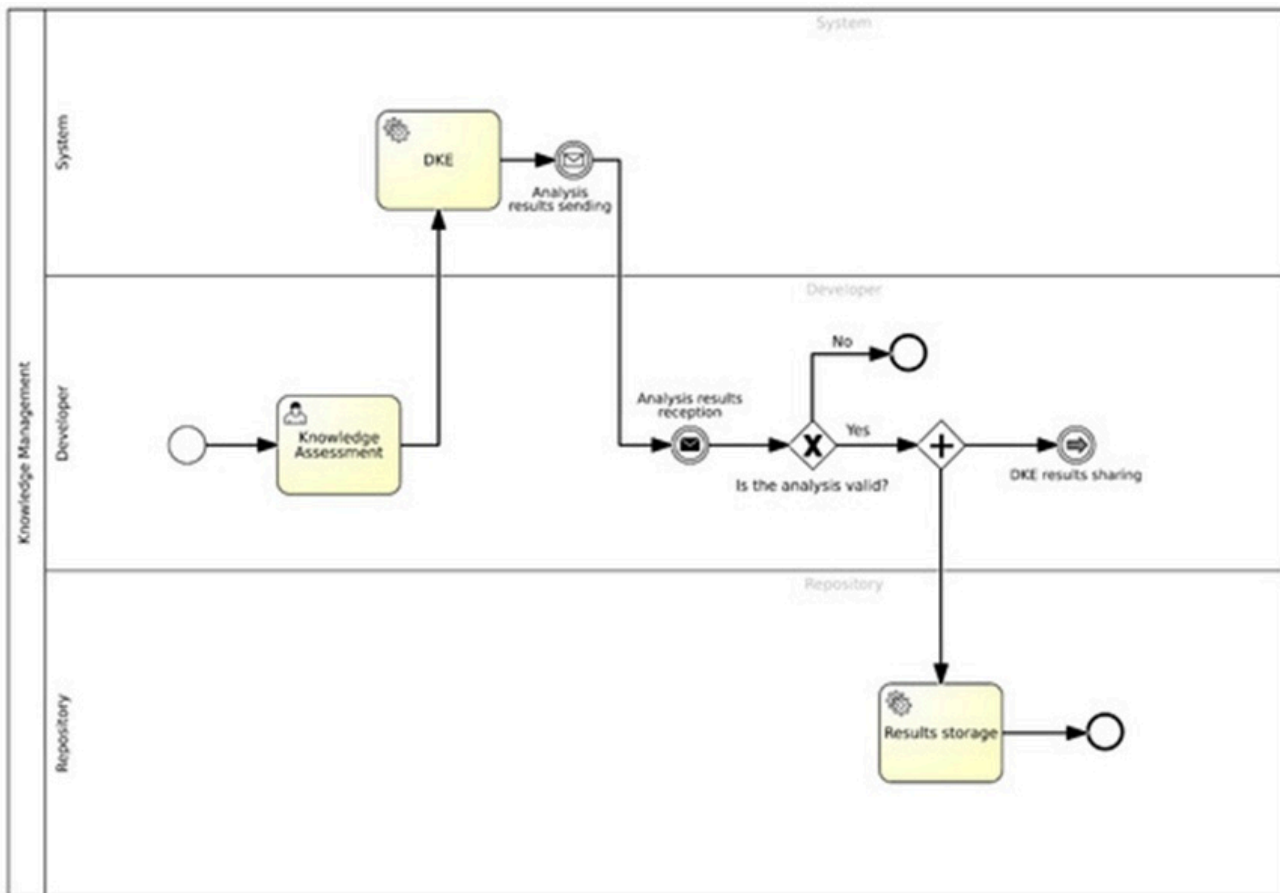


Figure 9. Knowledge Assessment management process.

This decentralized evaluation is performed by configuring specific operational parameters, including:

- Target KPIs, indicating the distinct performance metrics and quality benchmarks the analyzed knowledge is projected to achieve;
- The underlying data arrays slated for processing to extract the selected KPIs;
- The priority ranking assigned to each separate KPI;
- The baseline target success rate regarding the generated outcome, which is visually rendered via a corresponding RADAR chart; this benchmark establishes the acceptable efficiency threshold achievable within a specified time window;
- The total chronological duration of the scheduled analysis execution;
- The specific evaluation thresholds utilized to manage the prioritized metrics in the event that the DKE executes a subjective assessment run.

Once the analytical run concludes, the system fires a notification detailing the generated results. If the trial yields a positive outcome, the practitioner shares the complete data packet with their Scrum Master and immediate peers. Finally, the platform logs the analyzed knowledge asset within designated tables inside the BPWH environment.

The underlying process steps are defined below to clarify the workflow:

- **Knowledge Assessment:** Upon completing raw data capture and preprocessing, the user configures the parameters essential to launch the formal knowledge analysis. A dedicated submission interface records:

- Raw data paths;
 - Target KPIs;
 - Priority hierarchies assigned to each metric;
 - Chronological analysis duration;
 - Performance thresholds;
 - Minimum acceptable success percentage;
 - A concise technical description of the knowledge asset undergoing validation.
- **DKE:** The Data KPI Engine executes the automated validation sequence on the submitted data package. Upon a successful run, the knowledge asset is committed to the appropriate repository tier.

Data analysis engine

Within this specific architectural layer, the data analytics subsystem is driven by three specialized processing components:

- Best configuration routing;
- Association rule management;
- Machine Learning pipelines.

Best configuration

As previously noted in the lesson learned management workflow, during a pattern assessment cycle, the user must define multiple alternative configurations via a specialized form prior to routing the asset to the DKE for validation. If the engine encounters multiple valid, competing configurations, a dedicated optimization algorithm executes an automated query across the Data Lake to isolate historical records generated by prior assessment runs.

Once the historical data is retrieved, the algorithm evaluates the performance metrics of the current analysis against the past data. If the system succeeds in extracting a historical lineage containing alternative configurations of the pattern under review, it computes an optimized ranking matrix. This process is visually delineated in **Figure 10**.

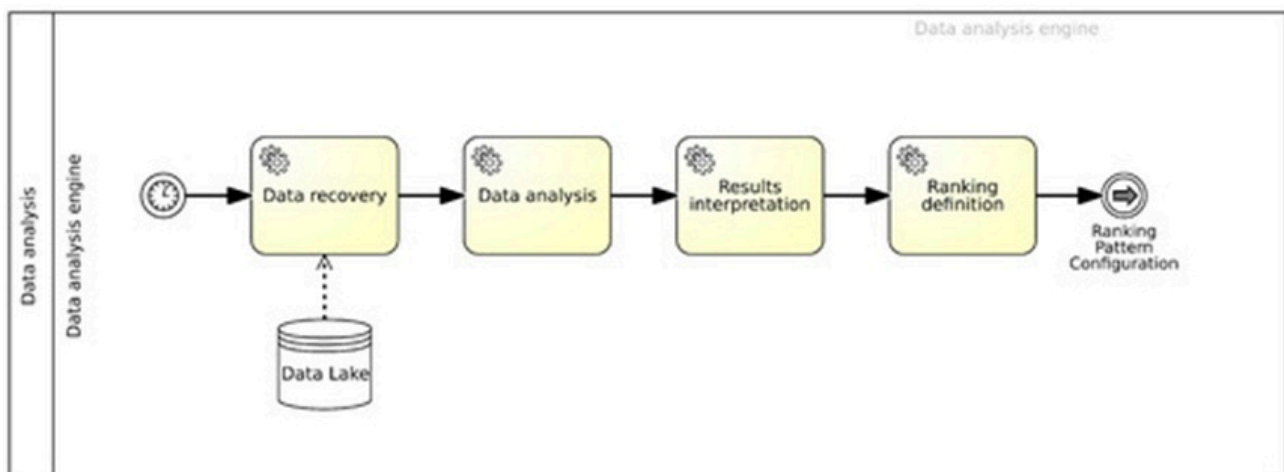


Figure 10. Data analysis process to validate the best pattern configuration.

Following the conclusion of this data analysis cycle, the platform transmits an automated notification detailing the generated results to the active members of the CoPs, who bear responsibility for pattern abstraction. The operational metrics necessary to establish an accurate pattern ranking are aggregated by the system iteratively after the underlying asset has been deployed across multiple production runs.

This ranking output—or more precisely, the structural configuration defined by the engine—constitutes a pattern in its own right, though it does not represent entirely new knowledge in the strictest sense of the phrase.

Instead, this subsystem serves to optimize and evolve the informational depth of an asset already housed within the active knowledge repository. Identifying the mathematically optimal configuration adds a layer of empirical metadata to the pattern profile, establishing a natural evolutionary path for the asset, which is managed as a specific version "release". Subsequent deployments of the same pattern may yield fresh data that refutes the current ranking, precipitating updated matrix calculations and the generation of newer version releases.

Naturally, the dynamics detailed here can reshape the core architectural properties that dictate pattern abstraction, such as altering the mandatory input and/or output parameter boundaries. This directly updates the primary pattern formalization interface, where the user is presented with a dynamic abstraction form adapted to incorporate the fresh insights yielded by the recently executed analysis.

Association rule management

In this configuration, the data analysis algorithm leverages association rules to map out dependencies between supplementary data arrays and the core parameters defined by the Scrum Team during the structural specification of a pattern, as illustrated in **Figure 11**. This operation takes place immediately after extracting and evaluating data blocks retrieved from a dedicated Data Lake. Grounded in these generated insights, the algorithm establishes the association rules through which novel, relevant variables can be correlated directly with the execution of a specific pattern.

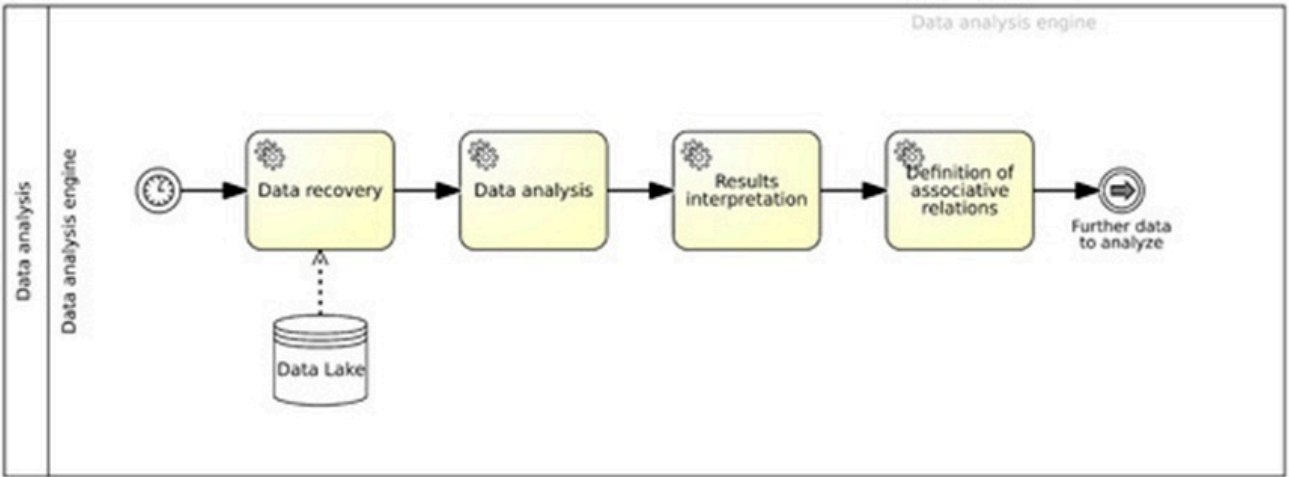


Figure 11. Data analysis process using association rules.

The systemic intelligence yielded by this segment of the analytical architecture manifests after a pattern has been deployed across multiple operational runs: the processing engine extracts statistically valid information by querying a large pool of observations gathered as different organizational users utilize the pattern in production.

The information produced by the data analysis engine under this scenario isolates additional data vectors that must be analyzed and incorporated into subsequent pattern assessment cycles. These architectural traits indicate that when the platform completes its data analysis run and broadcasts the findings, the designated recipients of these outputs are the

members of the Scrum Team. Guided by these automated insights, the practitioners initiate the formalization of a new lesson, thereby triggering the generation of fresh knowledge assets.

As highlighted previously, these system features inject considerable dynamism into organizational knowledge assets. The repository does not evolve solely through manual evaluation performed by software developers and cross-functional Scrum Teams; instead, it optimizes autonomously by harvesting the empirical data streams generated as these identical patterns are deployed across the enterprise. The broader workflow updated by the output of this analytical sub-layer is the lesson learned management process, and the specific operation most affected by this system alert is the lesson formalization task. Upon receiving the data broadcast, the members of the Scrum Team restructure and define a new lesson that accounts for the fresh data arrays and target KPIs, thereby cementing the core mechanism for autonomous knowledge creation.

Machine learning

The following section outlines the data analysis workflow driven by Machine Learning (ML) methodologies, with a comprehensive layout presented in **Figure 12**. While continuously monitoring the active execution of a specific pattern, the algorithm ingests empirical data blocks from the Data Lake, subsequently initiating advanced processing and analysis. Once the analytical cycle concludes and yields the metrics necessary to optimize the pattern's architectural framework, the algorithm targets the core structural elements that dictate the abstraction of that pattern for optimization.

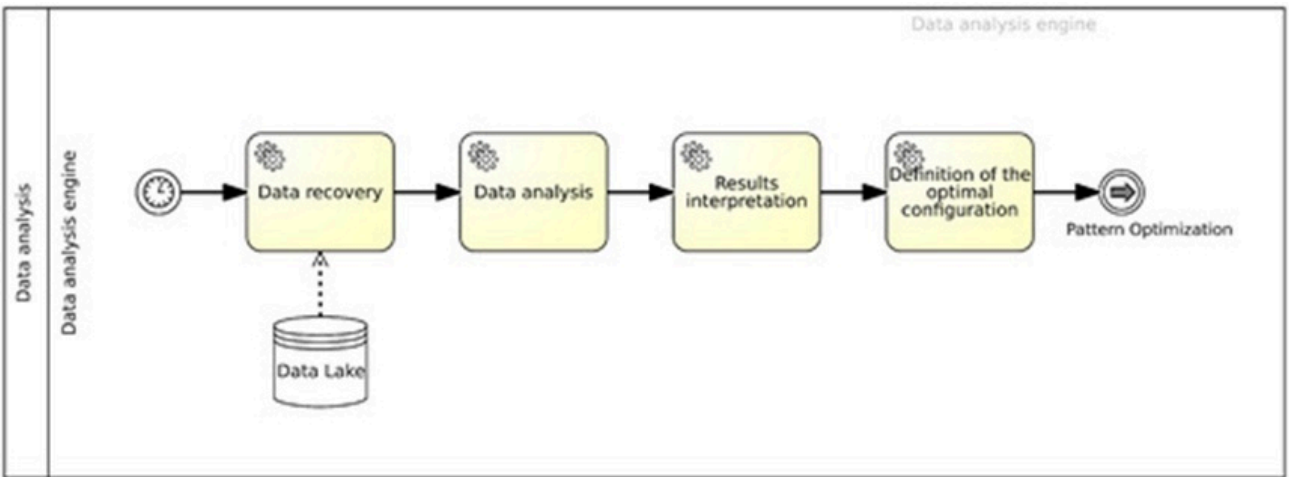


Figure 12. Data analysis process model using ML

The technical intelligence derived from this branch of the data analysis infrastructure further refines the information architecture of an active pattern, resulting in the publication of a new version release. Essentially, the output generated by this algorithm is powerful enough to directly modify the formal abstraction of a pattern and its underlying parameters, mimicking the operational impact observed in the first data analysis sub-layer. Here too, the parent workflow involved is pattern management, and the discrete activity impacted by this analytical output is the pattern abstraction task. Within this interface, the practitioner is presented with a dynamic abstraction form configured to ingest the supplementary data yielded by the recently concluded ML analysis.

Results and Discussion

To rigorously measure the efficacy and industrial validity of the proposed framework, validation across authentic manufacturing and operational environments was indispensable. A primary pilot study focused on the predictive maintenance architecture of a centrifugal hydraulic pump. Utilizing the Explainable Workflow Designer, the operational lifecycle was mapped out to capture critical tasks, sequences, and dependencies. This modeling phase successfully

brought implicit human expertise to the surface and codified organizational best practices that are historically difficult to capture or distribute via conventional methods.

Concurrently, through the AI Explainability Module, raw sensor data harvested from the machinery were processed via predictive algorithms. Advanced XAI methodologies, specifically SHAP and LIME, delivered highly interpretable, granular insights detailing the root causes of anomalies and identifying critical sensor performance thresholds. These interpretive outputs were rendered for factory operators through the Innovative HCI Module, ensuring an instantaneous, contextualized understanding of equipment deviations.

When specific sensor thresholds were breached, operators received immediate visual diagnostics and pre-modeled framework best practices, empowering them to intervene proactively and eliminate unscheduled downtime. This operational strategy enabled human operators to internalize complex domain expertise more efficiently, encouraged continuous workplace learning, and significantly minimized the facility's reliance on specialized technical experts to interpret process deviations.

In a secondary pilot application, the framework was deployed to optimize Maintenance, Repair, and Overhaul (MRO) workflows for commercial aircraft engines. The lifecycle of the engine servicing process was structured within the Workflow Designer to map tasks and information routing. Predictive analytics were leveraged to project component degradation and wear, while integrated XAI visualizations clarified which specific operational variables exerted the greatest influence on engine wear over time. This systemic blending of explainable AI facilitated more strategic, data-driven maintenance scheduling while strengthening overall process governance.

Furthermore, an array of rigorous evaluation tests was executed across various production environments reflective of Smart Manufacturing, Smart Health, and Smart Energy ecosystems. These verification runs were measured against a series of Key Performance Indicators (KPIs) rooted in Agile concepts, which directly mirror the methodology inspiring this knowledge management architecture. In particular, the following metrics were evaluated:

- **Lead Time:** The total chronological duration required to execute operational activities tied to the knowledge patterns generated and formalized via the platform.
- **Service Level Agreement (SLA):** Measured as the structural capacity of the framework to guarantee high service availability and operational standards by leveraging newly ingested knowledge assets.
- **Error Rate:** The mathematical frequency of operational mistakes or failures resulting from the real-world execution of newly codified knowledge.
- **Mean Time To Detection (MTTD) and Mean Time to Repair (MTTR):** Tracking the average time window required to identify operational errors during active processes, alongside the average duration consumed to resolve those disruptions using the acquired knowledge repository.

To evaluate the structural thresholds of the proposed platform, a series of comprehensive stress and load tests were performed to monitor how system response latencies evolved as the volume of concurrent API calls from external users increased. Specifically, performance values based on the standardized APDEX (Application Performance Index) benchmark were tracked across distinct operational testing scenarios by scaling the number of concurrent samples—representing active incoming requests—to obtain an objective map of system behavior and potential performance degradation.

Within each testing scenario, supplementary performance dimensions, including Response Time Over Time and Response Time Percentiles, were scrutinized due to the high diagnostic value these metrics yield. Tracking Response Time Over Time exposes how service performance fluctuates over extended operational runs, providing clear empirical evidence of processing bottlenecks or data queuing as user load and elapsed time scale up. Conversely, analyzing Response Time Percentiles maps out the precise SLA delivered across varying brackets of simultaneous user requests.

Regarding the actual performance data collected, the system successfully processed 100% of the simulated traffic during the 500-sample scenario, maintaining exceptionally low latency with only a marginal reduction in performance observed at the 99th percentile. This highly stable trend persisted uniformly when the load was scaled up to 2000 and 2256 concurrent requests. However, visible performance degradation began to manifest when the load reached the 5000-sample threshold, and distinctly suboptimal processing latency was documented under a heavy load of 15,000 simultaneous requests.

Additionally, a comprehensive comparative benchmarking analysis was conducted against contemporary ABPMS frameworks. The results indicate that the platform introduced here delivers distinctive, innovative architectural components, particularly regarding the native integration of knowledge management loops directly into the automated process adaptation cycle. **Table 4** outlines the compiled results of this benchmarking analysis.

Table 4. Benchmarking of AI-enhanced Business Process Management Systems (ABPMSs).

Evaluation Criterion	AI-Enhanced BPMS [1]	Self-Adaptive ERP [37]	Large Process Models [21]	Proposed Framework
Lifecycle Stages	Includes perception, reasoning, enactment, adaptation, explanation, and continuous improvement.	Supports ERP process adaptation using AI and natural language processing techniques.	Provides context-aware recommendations based on large language models (LLMs).	Encompasses perception, validation, formalization, adaptation, and dissemination phases.
Emphasis on Knowledge Management	Addresses knowledge management to a limited extent, without comprehensive mechanisms for validation or dissemination.	Concentrates primarily on technological aspects and lacks integration of socio-organizational knowledge processes.	Relies on data-centric approaches but does not provide structured knowledge-sharing mechanisms.	Places strong emphasis on knowledge management through structured roles and iterative learning cycles.
Organizational Structure	Organizational responsibilities are not explicitly specified.	Does not establish dedicated roles for knowledge integration activities.	Omits role definitions related to knowledge management processes.	Clearly specifies responsibilities through roles such as Scrum Masters and Communities of Practice (CoPs).
Socio-Technical Integration	Primarily technology-oriented, with organizational learning considered only implicitly.	Predominantly focused on technological adaptation and excludes community participation.	Centers on AI-enabled support while lacking a socio-organizational dimension.	Achieves a balance between technical and social aspects by fostering organizational learning and collaboration.
Validation Approach	Validation is implied but not associated with designated roles or procedures.	Does not provide a formal validation mechanism, depending instead on system-level adaptation.	Uses AI-based recommendations but lacks a structured process for validating lessons learned.	Establishes explicit validation procedures through CoPs and formally defined organizational roles.

Knowledge Dissemination	Knowledge-sharing processes are not clearly specified.	Does not include a structured mechanism for disseminating knowledge.	Knowledge transfer is mainly restricted to LLM-generated recommendations.	Knowledge dissemination is explicitly managed by Scrum Masters and CoPs.
Experience-Based Learning	Learning from experience is suggested but not formally articulated.	Gives limited attention to organizational learning processes.	Employs a data-driven approach without a structured organizational learning cycle.	Positions experiential learning at the core of the framework, using learning cycles to support process evolution.
Process Adaptation Mechanism	Supports adaptation but lacks a socio-technical perspective.	Adaptation is technology-driven and does not incorporate social learning elements.	Offers contextual adaptation but provides no structured feedback-loop mechanism.	Utilizes an iterative adaptation cycle based on validated feedback and clearly defined responsibilities.
Application Context	Designed for general Business Process Management (BPM) environments.	Intended for Enterprise Resource Planning (ERP) systems.	Developed for software-assisted BPM environments.	Tailored to complex and rapidly changing organizational settings.

The AI-Augmented Business Process Management System (ABPMS) manifesto introduced by Dumas *et al.* [1] delineates a modernized lifecycle consisting of six distinct phases: perception, reason, enact, adapt, explain, and improve, all driven by dependable AI technologies. Although this model provides a contemporary perspective on process development, it fails to specify operational protocols for validating and sharing knowledge through designated organizational roles. Crucially, it lacks a structured system that bridges experiential learning with proactive process modification.

Similarly, the Self-Adaptive ERP architecture designed by Maged and Kassem [37] serves as a prominent example of intelligent automation, targeting the technical reconfiguration of ERP platforms via AI and natural language processing (NLP). Nonetheless, its scope remains heavily technical, overlooking the socio-organizational elements of knowledge acquisition and discounting the vital contribution of human professionals and collaborative communities in maintaining sustainable innovation.

Furthermore, the Large Process Model methodology—identified by Kampik *et al.* [21] as a core conceptual structure for software-driven BPM in the era of generative AI—relies on merging large language models (LLMs) with knowledge-based systems to deliver context-aware recommendations. Here too, the priority is placed on data calculation and artificial intelligence, leaving an absence of defined roles or pathways for authenticating, formalizing, and systematically disseminating newly generated knowledge.

The table illustrates a comparative overview of these various ABPMS frameworks centered on their approach to knowledge management.

Within this landscape, the framework introduced in this study distinguishes itself through a socio-technical methodology that balances computational tools with cooperative organizational networks. A central feature is the deployment of explicitly designated responsibilities, such as Scrum Masters and Communities of Practice (CoPs), which are tasked with gathering, authenticating, codifying, and spreading experiential knowledge and optimal methodologies. This strategy

converts day-to-day operational experiences into collective knowledge, driving a continuous loop of education and advancement.

Additionally, the system executes a methodical iterative cycle that leverages verified feedback to guide process alterations as fresh requirements and shifting environments dictate. By blending automation with collaborative knowledge curation, this framework proves highly effective for intricate and fluid corporate settings where operational flexibility depends not only on technology but also on the capability to systematically capture, distribute, and recycle institutional knowledge over time. These capabilities address existing theoretical gaps, offering a conceptual advancement for ABPMS focused on boosting collective intelligence and nurturing continuous, durable, and flexible innovation.

The outcomes of this research demonstrate that the introduced framework successfully navigates multiple hurdles inherent in ABPMS engineering. By cultivating highly novel capabilities, the framework establishes a foundation for resolving these prominent obstacles, progressing beyond conventional methodologies. A primary difficulty involves guaranteeing situational explainability—namely, delivering context-specific and readily digestible justifications.

The framework tackles this issue by incorporating a dedicated repository for historical patterns and insights, where choices, operations, and outcomes are cataloged in a standardized arrangement. Offering constantly accessible and pertinent explanations enables operators to better comprehend system choices. Moreover, through sophisticated querying mechanisms and direct correlations to internal events and objects, the architecture permits the evaluation of complex operations alongside a comprehensive grasp of organizational behavior.

Balancing system independence with human oversight represents another significant challenge. To address this, the framework integrates an interactive, conversational architecture built around a forum-style environment. This layout encourages active participation from team members, allowing the system to operate autonomously while remaining fully accountable. Automated mechanisms offer context-driven recommendations and actions, complying with the constraints of dynamic collaboration while safeguarding human governance.

A foundational pillar of this framework is continuous optimization; by centralizing and archiving recorded insights and recurring workflow trends, business workflows are allowed to progress via an iterative methodology. Evaluating historical records uncovers core trends, aligning the system with shifting corporate demands—a strategy that refines current tasks while preparing the architecture to handle future disruptions efficiently.

From a Hybrid Process Intelligence perspective, the framework employs a cooperative paradigm that treats AI units as "learning apprentices" that conform to user workflows rather than enforcing inflexible procedures. This design, reinforced by the forum-centric platform, motivates team members to collectively contribute to designing workflows that are increasingly efficient and aligned with practical demands.

Ultimately, the architecture instills trust and dependability by maintaining transparency regarding exception management and unexpected disruptions. Centralized communication channels and categorized case studies boost user certainty, backed by advanced analytical tools that deliver precise, verifiable data.

Accuracy represents one of the most prominent metrics within the domain of data quality. It describes how faithfully the data points mirror real-world parameters or the actual values of the attributes they are intended to capture. Consequently, accurate records are those that precisely depict their target phenomena within a given operational scenario [38]. As specified by the ISO/IEC 25024 standard, accuracy is defined as:

"the degree to which data has attributes that correctly represent the true value of the intended attribute of a concept or event in a specific context of use".

Under this definition, any concepts tied to data rectitude, flawlessness, and the faithful replication of objective reality are governed by the accuracy metric [39].

A distinct element tied to this metric is the Data Accuracy Range, an analytical metric used to verify if dataset parameters fall within established boundaries. This form of evaluation tests whether values reside within acceptable limits; for instance, a data point is verified as accurate only when it falls between a specified minimum and maximum threshold.

The calculation of the Data Accuracy Range is derived using a basic equation: the volume of data records maintaining values inside the designated boundaries is divided by the overall volume of data records for which this threshold has been established. This calculation yields a metric between 0 and 1, where 1 demonstrates that all records comply with the anticipated boundaries, whereas lesser values point to the presence of anomalous or erroneous entries.

This metric enables an impartial determination of how closely the information aligns with operational expectations and whether it truthfully mirrors reality. A diminished output value highlights vulnerabilities within the dataset that could undermine subsequent data evaluations or strategic planning reliant on that information.

Furthermore, the framework designed in this study builds upon the foundations laid by Dumas *et al.* by merging multiple advancements. These include contextual and historical data analysis for a deeper view of corporate activities, a conversational layout that fosters teamwork, a modular architecture that supports step-by-step implementation, and utilities for advanced contextual interpretation. These outcomes demonstrate the capacity of the architecture to convert ABPMS platforms into more flexible, open, and cooperative ecosystems. This framework represents a major milestone in ABPMS development, offering solutions to well-documented literature limitations and opening new avenues for highly creative and efficient systems.

Limitations

While the introduced framework offers a substantial leap forward for ABPMS development, recognizing its structural limitations is critical. One constraint of the current iteration involves the evaluation and profiling of human behavioral variables. Even though the platform tracks cooperative dialogues, insights, and lessons using standardized and contextual metadata, it does not yet contain a dedicated analytical module designed to interpret individual or group behavior. Consequently, the diagnosis of behavioral patterns is only partially covered at this stage.

Additionally, although the framework is built for flexibility and scalability, implementing it within Agile settings characterized by accelerated sprints and intense development cycles warrants deeper investigation. Agile principles value rapid iterations, sparse documentation, and continuous shipping—factors that can clash with highly structured and formalized knowledge management workflows.

Another vulnerability is that the current execution pipeline lacks features for ongoing automated training or continuous learning. The underlying artificial intelligence instruments, including process mining and explainable AI (XAI), depend on static models trained on set datasets rather than refreshing automatically as new data flows in. Furthermore, the absence of an integrated MLOps framework prevents real-time performance tracking, model lifecycle governance, or automated training updates, restricting the platform's capacity to adjust to fluctuating contexts or changing input arrays.

The adaptability of the system across diverse corporate cultures represents another constraint; the heavy alignment with Agile and Scrum principles may reduce its viability within conventional or bureaucratic enterprises that operate under rigid hierarchies and strict protocols. Finally, regarding the codification of operational insights and best practices, the framework currently lacks an automated, systematic validation mechanism to officially verify incoming knowledge.

Future Research

The identified constraints offer valuable opportunities that delineate fresh avenues for investigation. Regarding the implementation of this framework within high-velocity Agile environments, the following trajectories for subsequent

advancement are recommended:

- Automation of knowledge capture: Embedded AI agents within the infrastructure could autonomously gather and categorize knowledge artifacts, thereby reducing manual labor and fostering frictionless knowledge transfer without interrupting sprint operations.
- Incremental and contextual integration: The structural design could be augmented to accommodate iterative knowledge enhancements that correspond with the cyclical nature of Agile methodologies, allowing for the ongoing documentation of architectural decisions and insights gained throughout each sprint.
- Toolchain compatibility: The systemic architecture could be engineered to integrate smoothly with pre-existing Agile tools utilized by development teams—such as Jira, Trello, or Confluence—thereby lowering the friction for knowledge reuse within daily operations.

Subsequent investigations will center on engineering lightweight components and add-ons optimized for core Agile ceremonies (e.g., sprint reviews and retrospectives). The objective is to facilitate real-time, low-friction knowledge management (KM) inputs while maintaining team velocity and delivery momentum.

To address the framework's scalability across varied industrial sectors and international operations, future studies could investigate the deployment of cloud-based distributed architectures to achieve flexible scalability and elasticity. Another viable research vector involves evaluating blockchain technology for decentralized consensus, which would ensure secure, verifiable, and transparent workflow synchronization in large-scale environments.

Furthermore, constraints arising from the primary focus on Agile and Scrum frameworks necessitate further inquiry into hybrid models that synthesize Agile concepts with traditional project management paradigms. The demand for automated verification procedures could also inspire the formulation of KPI-driven automated validation and peer-review systems.

Additionally, the framework could be bolstered by instruments and methodologies designed to analyze behavioral dimensions more thoroughly, as these areas remain underdeveloped. Specifically, techniques like process mining and social network analysis could be incorporated to evaluate group dynamics, shifting organizational roles, and knowledge-sharing behaviors within the Scrum Team. These integrations would yield deeper insights into cooperative processes and their subsequent influence on knowledge management.

Prospective development paths also include incorporating an MLOps framework to facilitate ongoing model updates and automated lifecycle orchestration. This approach would support the deployment of continuous re-training pipelines, automated verification, and monitored deployment strategies, ensuring sustained resilience and adaptability. Concurrently, more fluid approaches to process mining and Explainable AI (XAI) will be investigated to preserve high interpretability as the underlying models evolve. The enduring objective is to construct a durable, self-adaptive AI ecosystem that maintains transparency and adheres strictly to responsible governance standards.

Conclusion

This study addresses current concerns regarding the obstacles and opportunities of embedding AI within Business Process Management Systems (BPMSs). It introduces a structured framework that merges KM principles with AI-driven innovations, offering significant future advancements to the field of business process optimization. The framework enables efficient, dynamic knowledge dissemination via an accessible, web-based instrument engineered for recording and validating best practices and historical lessons. Moreover, it optimizes search methodologies to make information retrieval more efficient, addressing deficiencies in traditional KM approaches regarding the under-researched responsibilities of Scrum Masters and Communities of Practice (CoP) within Agile frameworks.

This inquiry also illustrates how AI can transform legacy BPM systems into real-time optimization platforms, spanning from automated knowledge validation to seamless human-AI collaboration. The modification of the SECI model for Agile environments demonstrates how empirical workflows allow organizations to achieve ongoing learning and iterative optimization. Emphasis is placed on modular deployment to ensure the new infrastructure can be custom-fitted to organizational requirements with minimal operational disruption.

A critical factor in establishing user trust within any AI architecture is the engineering of robust explainability mechanisms, ensuring that system recommendations are both transparent and actionable for diverse stakeholders. Furthermore, the ethical dimensions of AI deployment warrant deeper investigation, particularly regarding bias mitigation and data privacy preservation. A more comprehensive grasp of behavioral dynamics and institutional routines would facilitate the creation of more sophisticated and resilient systems.

Another promising domain for subsequent exploration involves enriching the framework's interactivity to support sophisticated dialogue between users and the system. Implementing proactive recommendations and real-time contextual feedback would be highly advantageous in this regard. Ultimately, empirical validation via real-world case studies will be imperative to verify system efficacy and refine the framework's architecture.

While the proposed model represents a significant standalone advancement, it concurrently serves as a foundation for extensive prospective research. Substantial exploration is still required concerning its scalability across different commercial sectors and diverse organizational structures.

Abbreviations

The following abbreviations are used in this manuscript:

ABPMS	AI-Augmented Business Process Management System
AI	Artificial Intelligence
BPMS	Business Process Management System
BPM	Business Process Management
BPWH	Best Practice Warehouse
CPA	Cognitive Process Automation
CoP	Community of Practice
DKE	Data KPI Engine
GANs	Generative Adversarial Networks
GenAI	Generative Artificial Intelligence
HCI	Human–Computer Interaction
KM	Knowledge Management
RUL	Remaining Useful Life
UCD	User-Centered Design
HITL	Human-in-the-Loop
BPE	Business Process Engine
MiSB	Middleware Service Bus

KPIs	Key Performance Indicators
LIME	Local Interpretable Model-Agnostic Explanations
SHAP	SHapley Additive exPlanations
LL	Lesson Learned
LPMs	Large Process Models
ML	Machine Learning
NLP	Natural Language Processing
PM	Process Mining
RPA	Robotic Process Automation
SCM	Supply Chain Management
SECI	Socialization, Externalization, Combination, Internalization
SoS	Scrum of Scrums
XAI	Explainable Artificial Intelligence
MRO	Maintenance, Repair, and Overhaul
SLA	Service-Level Agreement
MTTD	Mean Time To Detection
MTTR	Mean Time to Repair
APDEX	Application Performance Index

Acknowledgements

None

Conflict of interest

None

Financial support

This research was funded by **EXPLAIN “EXplainable Process and WorkfLow Design Automation with Artificial INtelligence”**—Programma operativo Puglia FESR 2014–2020 Regolamento Regionale PUGLIA per gli aiuti in esenzione N. 17 del 30-09-2014 e ss.mm.ii. Titolo II—Capo 2 AIUTI AI PROGRAMMI INTEGRATI PROMOSSI DA PICCOLE IMPRESE (ART. 27 Reg. n. 17 DEL 30/09/2014 E S.M.I.)—Asse III—Obiettivo specifico 1a (Innovazione)—Azione 1.3 Il futuro alla portata di tutti—Codice Progetto: IFNIMP2.

Ethics statement

None

Received: 07 Nov 2022

Revised: 15 Dec 2022

Accepted: 04 Feb 2023

Published online: 15 March 2023

Rights and permissions

Open Access The author(s) retain copyright. This article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. It may be shared and adapted for non-commercial purposes with appropriate attribution, an indication of changes, and distribution of adaptations under the same license. Third-party material may be subject to separate terms identified in its credit line. View the license at <https://creativecommons.org/licenses/by-nc-sa/4.0/>.

References

Dumas M, Fournier F, Limonad L, Marrella A, Montali M, Rehse JR, et al. AI-augmented business process management systems: a research manifesto. *ACM Trans Manag Inf Syst*. 2023;14(1):11:1-19.
<https://doi.org/10.1145/3576047>.

Yakyma A. Knowledge flow in scaled agile delivery model [Internet]. 2011 [cited 2025 Feb 20]. Available from: <http://www.yakyma.com/2011/07/knowledge-flow-in-scaled-agile-delivery.html>

Kovačić M, Mutavdžija M, Buntak K, Pus I. Using artificial intelligence for creating and managing organizational knowledge. *Teh Vjesn*. 2022;29(4):1413-8.
<https://doi.org/10.17559/TV-20211222120653>.

Psarommatis F, Kiritsis D. A hybrid decision support system for automating decision making in the event of defects in the era of zero defect manufacturing. *J Ind Inf Integr*. 2022;26:100263.
<https://doi.org/10.1016/j.jii.2021.100263>.

Enholm IM, Papagiannidis E, Mikalef P, Krogstie J. Artificial intelligence and business value: a literature review. *Inf Syst Front*. 2022;24:1709-34.
<https://doi.org/10.1007/s10796-021-10186-w>.

Taherdoost H, Madanchian M. Artificial intelligence and knowledge management: impacts, benefits, and implementation. *Computers*. 2023;12(4):72.
<https://doi.org/10.3390/computers12040072>.

Jarrahi MH, Askay D, Eshraghi A, Smith P. Artificial intelligence and knowledge management: a partnership between human and AI. *Bus Horiz*. 2023;66(1):87-99.
<https://doi.org/10.1016/j.bushor.2022.03.002>.

Alavi M, Leidner DE, Mousavi R. A knowledge management perspective of generative artificial intelligence. *J Assoc Inf Syst*. 2024;25(1):1-12.
<https://doi.org/10.17705/1jais.00859>.

Thakuri S, Bon M, Cavus N, Sancar N. Artificial intelligence on knowledge management systems for businesses: a systematic literature review. *TEM J*. 2024;13:2146-55.

Linder A, Anand L, Falk B, Schmitt R. Technical complaint feedback to ramp-up. *Procedia CIRP*. 2016;51:99-104.
<https://doi.org/10.1016/j.procir.2016.06.082>.

Chen E. Empowering artificial intelligence for knowledge management augmentation. *Issues Inf Syst*. 2024;25:409-16.

Casciani A, Bernardi ML, Cimitile M, Marrella A. Conversational systems for AI-augmented business process management. In: Guizzardi G, et al., editors. *Research Challenges in Information Science*. Cham: Springer; 2024. p.183-200.
https://doi.org/10.1007/978-3-031-59465-6_12.

Kokala A. Business process management: the synergy of intelligent automation and AI-driven workflows. *Int Res J Mod Eng Technol Sci*. 2024;6:12.

Zebec A, Indihar Štemberger M. Creating AI business value through BPM capabilities. *Bus Process Manag J*. 2024;30(8):1-26.
<https://doi.org/10.1108/BPMJ-07-2023-0566>.

Helo P, Hao Y. Artificial intelligence in operations management and supply chain management: an exploratory case study. *Prod Plan Control*. 2022;33:1573-90.
<https://doi.org/10.1080/09537287.2021.1882690>.

Aggarwal S. Guidelines for the use of AI in BPM systems: a guide to follow to use AI in BPM systems [master's thesis]. Lisbon: Universidade NOVA de Lisboa; 2021.

Rosemann M, vom Brocke J, Van Looy A, Santoro F. Business process management in the age of AI-three essential drifts. *Inf Syst E-Bus Manag*. 2024;22:415-29.
<https://doi.org/10.1007/s10257-024-00689-9>.

Szelągowski M, Lupeikiene A, Berniak-Woźny J. Drivers and evolution paths of BPMS: state-of-the-art and future research directions. *Informatica*. 2022;33:399-420.

Schaschek M, Gwinner F, Neis N, Tomitza C, Zeiß C, Winkelmann A. Managing next generation BP-x initiatives. *Inf Syst E-Bus Manag*. 2024;22:457-500.
<https://doi.org/10.1007/s10257-024-00681-3>.

Wang L, Liu Z, Liu A, Tao F. Artificial intelligence in product lifecycle management. *Int J Adv Manuf Technol*. 2021;114:771-96.

Kampik T, Warmuth C, Rebmann A, Agam R, Egger LNP, Gerber A, et al. Large process models: a vision for business process management in the age of generative AI. *KI-Künstl Intell*. 2025;39:81-95.
<https://doi.org/10.1007/s13218-024-00863-8>.

De Nicola A, Formica A, Mele I, Missikoff M, Taglino F. A comparative study of LLMs and NLP approaches for supporting business process analysis. *Enterp Inf Syst*. 2024;18:2415578.
<https://doi.org/10.1080/17517575.2024.2415578>.

Fahland D, Fournier F, Limonad L, Skarbovsky I, Swevels AJE. How well can a large language model explain business processes as perceived by users? *Data Knowl Eng*. 2025;157:102416.
<https://doi.org/10.1016/j.datak.2025.102416>.

Olatunji AO. Machine learning in business process optimization: a framework for efficiency and decision-making. *J Basic Appl Res Int*. 2025;31:18-28.

Chapela-Campa D, Dumas M. From process mining to augmented process execution. *Softw Syst Model*. 2023;22:1977-86.
<https://doi.org/10.1007/s10270-023-01132-2>.

Gabryelczyk R, Sipior JC, Biernikowicz A. Motivations to adopt BPM in view of digital transformation. *Inf Syst Manag*. 2024;41(4):340-56.
<https://doi.org/10.1080/10580530.2022.2163324>.

Salvadorinho J, Teixeira L. Organizational knowledge in the I4.0 using BPMN: a case study. *Procedia Comput Sci*. 2021;181:981-8.
<https://doi.org/10.1016/j.procs.2021.01.266>.

Abbasi M, Nishat RI, Bond C, Graham-Knight JB, Lasserre P, Lucet Y, et al. A review of AI and machine learning contribution in predictive business process management (process enhancement and process improvement approaches). *arXiv [Preprint]*. 2024;arXiv:2407.11043.

Nonaka I, Takeuchi H. *The knowledge-creating company: how Japanese companies create the dynamics of innovation*. New York: Oxford University Press; 1995.

Beck K, Beedle M, van Bennekum A, Cockburn A, Cunningham W, Fowler M, et al. *Manifesto for agile software development* [Internet]. 2001 [cited 2025 Apr 10]. Available from: <https://agilemanifesto.org/>

Carroll JM. Human-computer interaction (HCI). In: *Encyclopedia of Human-Computer Interaction*. Aarhus: The Interaction Design Foundation; 2009.

da Costa Brito L, Quaresma M. User-centered design in agile methodologies. *Ergodesign HCI*. 2009;7:126-37.
<https://doi.org/10.22570/ergodesignhci.v7iEspecial.1285>.

Nielsen J. 10 usability heuristics for user interface design [Internet]. Fremont (CA): Nielsen Norman Group; 1994 [updated 2024 May 30; cited 2025 May 6]. Available from: <https://www.nngroup.com/articles/ten-usability-heuristics/>

Easa NF, Fincham R. The application of the socialisation, externalisation, combination and internalisation model in cross-cultural contexts: theoretical analysis. *Knowl Process Manag*. 2012;19:103-9.

European Commission. *White paper on artificial intelligence: a European approach to excellence and trust* [Internet]. Brussels: European Commission; 2020 [cited 2025 Mar 13]. Available from: https://commission.europa.eu/publications/white-paper-artificial-intelligence-european-approach-excellence-and-trust_en

High-Level Expert Group on Artificial Intelligence. *Ethics guidelines for trustworthy AI* [Internet]. Brussels: European Commission; 2019 [cited 2025 May 5]. Available from: <https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai>.
<https://doi.org/10.2759/346720>.

Maged A, Kassem G. Self-adaptive ERP: embedding NLP into Petri-net creation and model matching. In: *Proceedings of the 2024 International Conference on Computer and Applications (ICCA)*; 2024; Cairo, Egypt. Piscataway (NJ): IEEE; 2024. p.1-6.

Miller R, Whelan H, Chrubasik M, Whittaker D, Duncan P, Gregório J. A framework for current and new data quality dimensions: an overview. *Data*. 2024;9:151.

Gualo F, Rodríguez M, Verdugo J, Caballero I, Piattini M. Data quality certification using ISO/IEC 25012: industrial experiences. *J Syst Softw*. 2021;176:110938.
<https://doi.org/10.1016/j.jss.2021.110938>.