KeAi
CHINESE ROOTS
GLOBAL IMPACT

International Journal of Innovation Studies

INNOVATIVE APPROACHES TO ADDRESS SCALABILITY CHALLENGES IN WORKFLOW AUTOMATION FOR LARGE-SCALE ENTERPRISE BUSINESS PROCESS MANAGEMENT SYSTEMS

Kanneganti Ravi Kiran[1*]
Department, FS SBU, Capgemini America Inc
kanneganti.ravi@gmail.com
ORCID ID: 0009-0003-5217-4344

Abstract

The paper describes a workflow orchestration framework that supports the automation of the incident management process in enterprise IT using scalable and AI-powered workflow orchestration. With the help of a real-world ITSM event log set of data made available by Kaggle and containing more than 240,000 entries, we will introduce a predictive mechanism to forecast the number of tickets coming and proactively make simulated autoscaling choices. Due to intensive data preparation, event trace generation, and discovering patterns, we can identify temporal patterns in workflow activities and focus incident processing on volume versus priority. A specific model, Facebook Prophet, is applied to predict daily ticket inflow, and autoscaling logic is prompted based on the prediction of predicted thresholds reaching 400 events/day. The simulation justifies using the model to facilitate the redistribution of resources before the busy season. Important performance tools, such as average resolution time, the number of escalations, and satisfaction ratings averaged by the impact level, are scrutinised extensively. The proposed framework is more effective than the usual static orchestration systems because it presents the idea of forecasting-based provisioning, which provides both the enforcement of the SLA and optimisation of operational costs. Another research gap that is closed in our study is that we integrate process mining and predictive autoscaling, in contrast with reactive triggers. The contribution of this work to the literature is a practical, modular and enterprise-capable solution of intelligent BP M orchestration. This model can also include LSTM and Transformer-based architectures in real-time streaming systems.

Keywords: *Business Process Management (BPM), Workflow Automation, Autoscaling, Incident Management, Time Series Forecasting, ITSM, Facebook Prophet, Process Mining*

1. Introduction

In this digital age, many businesses rely heavily on business process management (BPM) systems to promote operational effectiveness, process flexibility, and customer contentment (Ahmad & Van Looy, 2020). BPM combines business modelling, execution, monitoring and optimisation of business processes, commonly integrating disparate applications, working groups and services into one platform. Workflow automation, the systematic rules-based orchestration of parallel and sequential processes, intends to decrease customised intervention, make the most of throughput and satisfy regulatory necessities and is to be found at the heart of this ecosystem (Viriyasitavat et al., 2020).

There has been tremendous development in workflow automation, and a more innovative, dynamic environment, instead of the prior rule-based implementation, is being sought as an execution engine. DevOps pipelines' emergence and growing maturity, cloud-native platforms, containerisation, and the complexity of task routing, escalations and resource allocation have only enhanced this evolution (Liao et al., 2024). Despite all these developments, enterprises have not ceased to experience problems related to scalability when working on large volumes and other forms of dynamic loading. With organisations increasing the number of workflows that they digitalise and the incorporation of third-party APIs, IoT triggers, and real-time event streams, scalable and intelligent workflow orchestration is giving rise to two priorities: how to do it and how to do it effectively (Ramos & Arumugam, 2023).

In addition, the advent of AI, machine learning and predictive analytics opens up the possibility of streamlining how work is done and administered within enterprise ecosystems (Jauhiainen, 2024). Applying AI to BPM may allow for the implementation of smart schedules, adaptive resource allocation, fault identification, and self-healing of processes. Nevertheless, such innovations remain immature when it comes to their enterprise adoption, as most of the organisations around have to stick with the old school, predetermined rules, and non-elastic, monolithic process engines that cannot possibly elastically scale with spikes in workloads (Gadde, 2023).

Although BPM systems have advanced in functionality and coverage, scalability developments are still in their infancy, with enterprise-scale deployments limited by severe scalability constraints (Pourmirza et al., 2017). The main problem is that the system cannot change its workflow execution model due to high volume and concurrent process instances. Such an issue is especially sharp in IT service management (ITSM), financial operations, and customer service centres, where thousands of workflows can be launched simultaneously (Narne, 2023)One of the resulting impacts of poor scalability is latency, failure to meet the SLA criteria, show stoppages of the subprocesses, and eventual process failure, which negatively affects the user experience and business performance.

Conventional BPM models are more prone to using hard-coded orchestration logic that involves fixed resource pools, a straightforward task queue, and manual routing capabilities. Such architectures also rely on steady workloads and deterministic execution direction, and they do not support real-time variation in process volume and complexity (Bartlett et al., 2023). Moreover, rule-based engines cannot preemptively predict and respond to surges in processes and are context-blind, which causes inefficient resource utilisation and multiple system overloads at peak load times. The reimagining of the BPM architectures in terms of them being inherently scalable, intelligent and adaptable, is critical with the move by modern enterprises towards distributed, hybrid and multi-cloud environments (Gonzalez-Lopez & Bustos, 2019).

The advanced implementations of BPM systems are supposed to scale to thousands of workflow sources and nesting in the hierarchy of subprocesses, inter-functional approvals, and asynchrony in the case of handoff between processes, in practice, even to such an extreme that they were not expressly stated (Castro et al., 2020). Such systems will likely suffer service degradation without dynamic autoscaling, intelligent task distribution, and predictive orchestration processes. It highlights the importance of researching more modern methods to incorporate the best software engineering practices with AI-enhanced automation to solve these scalability issues (Szelągowski & Berniak-Woźny, 2024).

The proposed research aims to present and empirically test new strategies to leverage large-scale enterprise BPM systems in scaling and automating their workflows. It considers exercising real-world incident management data, which presents high-volume ITSM experience and simulates actual scalability problems that implicate enterprise workflow engines. On the one hand, it considers the effectiveness of a new framework combining AI-based forecasting, microservice architecture, and autoscaling rules.

The main goal is to create and verify a scheme that can dynamically predict workload peaks, estimate the required proportionate number of resources, and keep the performance stable over time without human involvement. To do so, we consider applying a time-series forecasting model (Prophet) to forecast the future workflow volume and use its results to conduct proactive scaling simulations. The research focuses on how the execution of distributed processes and flexible load capabilities can overcome the known bottlenecks in BPM by adjusting them to these predictive capabilities, combined with orchestration through microservices.

The study aims to reduce the understanding of the gap between theoretical process modelling and deployable solutions. The empirical analysis is provided in detail by implementing the proposed approach to answer the question of its applicability and benchmarking with the proposed approach about process metrics, such as the latency, throughput, level of satisfaction, and the time of the process accompanying the proposed approach.

This research has threefold contributions. Initially, the study proposes a scalable architecture to automate enterprise workflows, which should embrace AI-driven orchestration, distributed microservice execution, and auto-scaling. This framework is cloud-native and complies with contemporary trends in BPM deployment.

Additionally, the following study visually and statistically compares real process data with a real-life incident management event log that contains more than 240,000 events representing a large-scale incident management. This data represents a typical workload to compare the process density, execution pattern, and correlation of satisfaction and variations in resource demand.

Moreover, the proposed study shows the prediction-driven orchestration mechanism, which uses Prophet to predict workflow volume over time. Based on such forecasts, it simulates an autoscaling approach that has the potential to scale resource pools before the possibility of workload bursts. This innovation is an improvement over reactive scaling logic and thus can allow BPM systems to become more predictive, resilient, and cost-efficient.

The paper will add to the existing debate about the relationship between innovation, enterprise process management, and automation using artificial intelligence. It directly reflects the International Journal of Innovation Studies' priority in interdisciplinary research, value-creating, future-focused research, and extending enterprises' adaptability, resilience, and performance in technological innovation.

2. Literature Review

2.1 BPM and Workflow Automation Systems

Business Process Management (BPM) has become a fundamental area of expertise in managing and enhancing the end-to-end business processes in public and privately owned organisations. Based on initial workflow and enterprise resource planning (ERP) applications of the 1990s, BPM has developed into a comprehensive platform that supports design, automation, tracking and continuous enhancement of processes (Singasani, 2019). In the last 20 years, BPM systems

have evolved to be less detailed, less constrained, and much more dynamic, interoperable systems that can handle dynamic workflows that span functional boundaries. The BPM of today no longer concentrates solely on the efficiencies of operations, but also on customer-driven innovation and change responsiveness (Szelągowski & Lupeikiene, 2020).

The core of BPM is workflow automation, which allows orchestrating tasks, actors, rules, and events in a digital environment. Unrestricted BPM applications, such as IBM BPM and Oracle BPM Suite, gave a model-driven method of automating repetitive business processes (Baiyere et al., 2020). By comparison, new platforms like Camunda, Appian and Bizagi have adopted the low-code/no-code paradigm, where business users have more direct control over the design and deployment of workflows. These solutions accommodate Business Process Model and Notation (BPMN) routines, application program interface (APIs) and enterprise systems and implementation engines that convert models into business processes (Schäffer et al., 2021). Camunda, for example, allows BPMN-based workflow execution in containerised microservice settings, and Appian includes optional AI, RPA, and decision automation modules.

Even with these developments, BPM in practice continues to be based on pre-designed workflow, with predetermined routing, assigned tasks and limited ability to adapt to the dynamics of the work tasks. Consequently, they are overwhelmed with digital business ecosystems that demand high concurrency levels and complexity (Bazan & Estevez, 2022). Dynamic scaling of workflows with visibility, compliance, and process quality has emerged as a key research and practice agenda for BPM researchers.

2.2 Scalability Challenges

Scalability in BPM means the ability of the system to scale the number of instances of processes and related activities without showing any decline in performance, reliability, or quality of response (Schulte et al., 2015). With increased data intensity and concurrent and distributed workflows, conventional BPM engines face various structural and operational scale-inhibiting bottlenecks.

One of the biggest challenges is thread contention in monolithic process engines. The shared resources, including the task queues, the databases and the execution threads, are suddenly hit at once as many processes aim to perform concurrently, thereby delaying the execution and contributing to latency. Most traditional engines cannot isolate and parallelise workloads effectively in distributed conditions (Bartlett et al., 2023). As a result of this, instances of processes contend with scarce computing resources, and more so at peak demand times.

The other most common challenge is sequential routing, which involves linearly processing tasks in workflows performed in predefined routes. Although this model offers adherence and predictability, it is not flexible enough to handle changes in resource availability or even in line of reasoning (Satyal et al., 2017). This inflexibility leads to orchestration delays, mainly when intermediate tasks need to be done, and human contribution delays them, or third-party services cannot give prompt responses or are limited by an infrastructure.

Moreover, static task assignment protocols, where the task assignment is performed using hard-coded rules or specified user roles, result in overloading or underutilisation of resources. For example, a "Level 2 Support" user can simultaneously have all the escalated incidents, which clogs their queue. However, other available resources can be left unutilised. Task complexity,

task urgency or prior task resolution times are not incorporated in static orchestration, resulting in the task being mapped improperly.

Finally, traditional BPM architectures lack real-time telemetry and feedback loops, compromising their capability to define performance problems before they occur. Most BPM systems lack inbuilt observability and adaptive logic and, therefore, are unable to identify when their level of service is under pressure or where particular nodes need elastic provisioning (Pourmirza et al., 2017). These restrictions have facilitated delays in service processes, SLA violations, and user dissatisfaction on an enterprise scale.

2.3 Current Trends

To cope with the drawbacks of the conventional BPM systems, the industry has been experiencing trends toward more modular, intelligent, and scalable workflow execution models. Among the most significant developments, one can list the use of the microservices architecture in BPM. Microservices enable the deployment of the components of the workflow, separate scaling, and maintenance (Chaima & Khebızı, 2022). BPM systems may be better at work distribution because they encapsulate the activities of approval, notification, data validation, escalation, etc., in individual services, thus avoiding bottlenecks of monolithic execution engines.

Some of the present-day BPM solutions have started adopting the ideas of microservices. Camunda, for example, has external task clients that connect with the engine through REST or gRPC, thus allowing it to scale individual workflow steps horizontally. The flexibility will enable organisations to uncouple business logic and infrastructure and deploy workflow fragments in cloud-native environments, improving performance under dynamic load (Ugwueze, 2024).

Another development trend is incorporating artificial intelligence (AI) into implementing BPM orchestration and scheduling. AI-based decision-making has the potential to maximise the assigning of tasks by creating them based on historical trends that indicate which resources achieve the best results, and could effectively work based on the SLA priorities (Kokala, 2024). They can use machine learning models to predict the highs in workload, estimate the time a task would take, and detect anomalies in executing processes. Moreover, reinforcement learning algorithms adjust workflow routing strategies in response to dynamic business conditions (Pan & Wei, 2024).

At the same time, process mining and conformance checking have recently emerged as methods to analyse and optimise workflow automation. Organising the extraction of process models out of event logs and comparing with designed workflows allows discovering violations of compliance and bottlenecks, and detecting traces where non-conformance is applied (Rinderle-Ma et al., 2023)Tools like PM4PY and Celonis help convert the raw logs into actionable information, allowing for continuous process optimisation and supporting decisions. These tendencies indicate an even stronger belief that BPM systems should move from static, rules-driven architectures to more adaptive, intelligent, and scalable ecosystems. However, these innovations have not yet found a wide footing in BPM tools and practice.

2.4 Research Gap

Although positive trends in AI implementation and microservice-based orchestration point to a bright future, no notable research has been conducted on the aesthetic assessment of scalability approaches taken in real-life BPM scenarios. The available literature on BPM

scalability is mostly theoretical or confined to simulation and is not based on live workload data (Xue et al., 2021). The current application of AI to workload prediction and auto-scaling with BPM execution engines is limited (both academically and within industry in general).

Specifically, the number of studies using time-series forecasting, including the Prophet, to predict workflow volume and make autoscaling decisions, is minimal. Prophet is a business-appropriate, powerful, and explanatory forecasting model developed by Facebook and is effective in forecasting business event data with a daily or seasonal trend (Guruge & Priyadarshana, 2025). It has not been thoroughly scrutinised regarding providing predictive scaling to BPM. Also, the latest BPM tools do not have innate connections to predictive models, and running some forecasting information through orchestration layers is challenging.

In addition, enterprise-grade event logs are not extensively used to perform the scalability experiments, nullifying the proposed solutions' validity. Numerous experiments are based on artificial data or small-scale logs, unable to represent the challenge and diversity of workflows in large enterprises (Faizal & Aisyah). Consequently, applied research is evident in assessing the scalability strategies with real-life and large-scale process information and exhibiting some actual performance benefits in all leading indicators like latency, throughput, resource consumption, and user satisfaction.

The proposed study aims to fill these gaps. It provides a data-driven validation of the proposed framework, including Prophet-based Forecasting, microservice orchestration, and autoscaling emulation, using an enterprise ITSM dataset as an example. This helps theoretically and practically with inclusive, smart workflow automation in contemporary BPM systems.

3. Methodology

3.1 Proposed System Framework

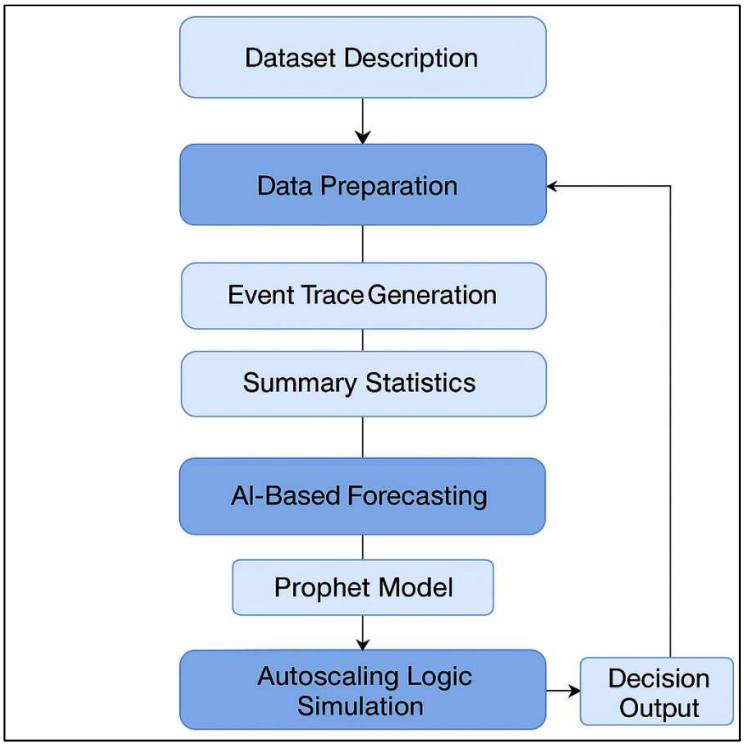

Figure 1. Proposed System Framework

Fig. 1 shows how a step-by-step pattern turns scalable workflow automation in the enterprise BPM. It starts with the dataset description and continues with data preparation, which involves parsing and cleaning. Process behaviour is understood through event traces and the generation of some summary statistics. These observations are fed into AI-based forecasting, primarily through the Prophet model, which is used to predict future workflow volume. The Autoscaling Logic Simulation is powered by the forecast and is converted to a binary Decision Output (scale or not). This reasoning pipeline facilitates intelligent orchestration and prospective provisioning of resources using real-time and historical operational data.

The research aim of dealing with scalability challenges in BPM systems using innovation is directly helped by this methodology. The framework takes a step further by combining live enterprise logs with a predictive autoscaling process, allowing it to break out of rigid execution models of processes. The innovation is that the combination of the forecast-based decision-making (Prophet) and logic simulation of autoscaling is hardly covered in BPM literature. Contrary to other methods that are reactive to system stresses, this technique is proactive in treating the resource provisioning based on load prediction. It also demonstrates how AI and microservices can complement each other to increase the resilience and efficiency of enterprise workflows, thereby making it fit into the remit of the International Journal of Innovation Studies as an interdisciplinary journal focusing on innovative thoughts that look ahead.

3.2 Dataset Description

The study is based on a real-world event log extracted by Kaggle, the IT Incident Management Log Dataset, to empirically test the workflow automation strategies' scalability in large-scale BPM processes. With 242,901 records, the dataset includes the full life cycle of the enterprise incident processes within an IT Services Management (ITSM) business simulation. This scale and granularity make it very appropriate to study the concentration of process bottlenecks in the real world, process resource allocation movement, and workload fluctuations with time.

The data and significant features reflect those used in the enterprise BPM context. These would be the Case ID (a unique identifier of a particular instance of the process), time-stamp (the date and time of a specific occurrence), Event (the description of the stage of the activity, e.g. Ticket Created, Assigned, Escalated), Resolver (the person or a team that is working on a particular task), Priority (Low, Medium, High), and Customer Satisfaction (a score that the customer leaves after a problem is resolved). It provides the whole history of an instance of a workflow, from its initiation until its resolution, so one can analyse the participants' time, complexity, and participation. Those features make the simulation of a load condition, the prediction of the workflow level, and the testing of auto-scaling operations realistic.

Most importantly, this data can be used to model event-oriented interactions in real-time-based BPM systems and systems with high traffic. Because it has human—and system-generated transitions, it is perfect for estimating how a process can be enhanced by AI-enabled orchestration and autoscaling to perform better when the workload is at its peak.

3.3 Data Preparation

The data preparation pipeline encompassed thorough preparation of the dataset before its analysis. The raw data came in one semicolon-delimited field of the type, which held string data and needed to be parsed into columnar form. The dataset could then be parsed into eleven separate columns using the Pandas library available in Python, with particular care given to its date-time and string data accuracy. After parsing, the Time-stamp field was normalised with

the format of %d/%m/%Y %H:%M to display the temporal congruency. Such entries that were not time-stamped or had an invalid time stamp were eliminated to maintain the quality of time-based analysis.

Identical events will be removed depending on a composite key of Case ID, Event and Time-stamp; this could be due to system retries or incorrect logging. This was taken to ensure every action was distinctively linked to only one time-point and placed in a workflow state. To analyse as a downstream event, the Case IDs were considered individual process instances, and all the events assigned to a specific ID were combined and ordered according to their time stamps to obtain event traces. They used these traces to base the process durations, transition sequences, and escalation frequencies calculations.

Important attributes were subjected to the creation of descriptive statistics to describe the dataset's characteristics. There were more than 40,000 distinct cases, each including between 2 and 15 events on average. The priority levels were evenly spread, with a preference towards medium priority, and there was a concentration around score three on the satisfaction scale. This initial analysis demonstrated the richness and adequacy of the dataset to analyse the results of workflow execution and the quality of service.

3.4 EDA Framework

To obtain knowledge about the event log and to structure the scalability simulation, an exploratory data analysis (EDA) framework was used. The first dimension of its analysis was preoccupied with the relationship to event frequency distribution, which disclosed the most frequently occurring activities regarding the workflow life cycle. The most common events were "Ticket Created", "Assigned", "WIP", "Escalated" and "Closed", which followed the key phases of incident work according to ITSM. A countplot of such events was created to see their occurrence in relation to each other.

Furthermore, a priority-level analysis was done by categorising the cases according to their allocated urgency. Cases with medium priority were revealed to have the largest number, according to the organisational standard, where most service requests consist of non-critical cases. This disaggregation helped make correlations between workloads and the impact they could have on the SLA plateau.

Daily aggregations applied to event time stamps were used to measure volume trend over time to create a time series plot of the frequency of incidence. This visualisation helped to find periodic spikes and, in addition, it allows the determination of peak load days, which calls for a forecasting model in the long run. Determining the frequency of workload patterns marked by high volatility was also helpful.

To compute the time taken by the process, the difference between the time of the first and last time-stamp of each Case ID was derived. The data were non-normally distributed; thus, a log transformation was necessary. The log-transformed data produced a right-skewed histogram, with the vast majority falling between 0 and 200 minutes, but was heavy-tailed, with a long right tail heavier than 1000 minutes. Such outliers were an indication of unsolved issues or resource crunch. Median and average times were calculated as a comparative measure of the system's performance.

Lastly, the customer satisfaction trend was determined by comparing customer satisfaction to priority levels. The boxplot indicated a higher variance and a small dip in median satisfaction in cases with high priorities, which led to potential overloads at critical moments. Such EDA

outputs were used to set up the forecasting and autoscaling logic threshold discussed in the following sections.

3.5 AI-Based Forecastingforecasting

To present proactive scalability to the BPM implementation model, a forecasting system based on the Facebook Prophet, a good time-series forecasting library optimised for business metrics, was used. According to the data, the forecasting model was trained using a series of daily counts of the events. The time series consisted of a unique data value that presented the total number of events considered on a particular date. The reason Prophet was chosen was because it could work with seasonality, change in trends, and irregularities prevalent in data on operations processes.

The historical data provided was used to train the model, and distance seasonality was set to fit daily changes in workload. After training, it was deployed to predict the workflow volume 7 days in advance, a standard operation planning period. The forecast output had a central prediction (yhat) and confidence intervals; deterministic and probabilistic autoscaling approaches were possible.

A scaling threshold was defined based on the system's observed limiting capacities. When the number of predicted events received on a particular day was more than 400, the system was deemed under high traffic and marked as one that would be scaled. Such a limit was set according to the duration of case analysis and resource saturation trends. Every forecasted day was given a binary variable, and it was scale_out = Yes when that> 400; No otherwise. Such AI-based forecasting enabled the system to predict an increase in workload and pre-schedule resource allocation instead of responding to a failure or SLA violation.

3.6 Autoscaling Logic Simulation

An autoscaling simulation was performed to determine the forecast's practical consequences. Each day's decision logic was transformed into a scaling plan table, where the precast dates were linked to the predicted event count and scaling directive. This simulation was an analogue of a live provisioning system, e.g., the Horizontal Pod Autoscaler in Kubernetes or the triggers within serverless functions in AWS Lambda.

On each observation day, if the forecast exceeded the scaling threshold, the model would scale up workflow execution infrastructure, e.g., spinning up more task processors or parallel execution containers. On the other hand, days projected to experience less workload were permitted to work at the baseline level. Analysis of scale-out frequency was also part of the plan, which assisted in estimating autoscaling's overall cost and resource impact.

The proposed scalable architecture was based on this simulation, which constituted predictive analysis in combination with orchestration logic to adjust to the changing workload scenarios on a dynamic basis. It showed how BPM enterprise systems would evolve beyond their current Frozen-In-The-Middle of execution engine status into self-optimising workflow systems that would satisfy the requirements of modern scale.

4. Results and Analysis

This section provides the analysis results of the incident management dataset based on descriptive analytics and predictive modelling. The aim is to detect the point of scalability limitation, describe the nature of workload intensity, and assess the efficiency of the given forecasting-based autoscaling reasoning. The insights have been classified into critical

dimensions of process volume, trends of execution, SLA performance, and the impact that decisions are expected to have.

4.1 Event Volume and Workflow Complexity

The initial analysis was conducted to define the standard types of events in the incident workflow. Figure 2 demonstrates several top events with more than 30,000 occurrences, such as Ticket closed, WIP- level 1 support, and Ticket created. This implies that most incidents undergo the lifecycle, which is relatively stable and structured, starting with the creation stage, going through the level-1 WIP stages, and finishing with resolution or closure.

An indicator of the transactional nature of enterprise workflow processes is further confirmed by the high volume of the type of events at the level of Ticket assigned to level 1 support and customer feedback received. The fact that the escalation to level 2 and level 3 support is relatively less frequent means that the number of non-trivially complex and resource-intensive workflows is also non-trivial. This complexity imposes a load on the BPM engines, especially when dealing with burst loads, as concurrency and routing require real-time orchestration.

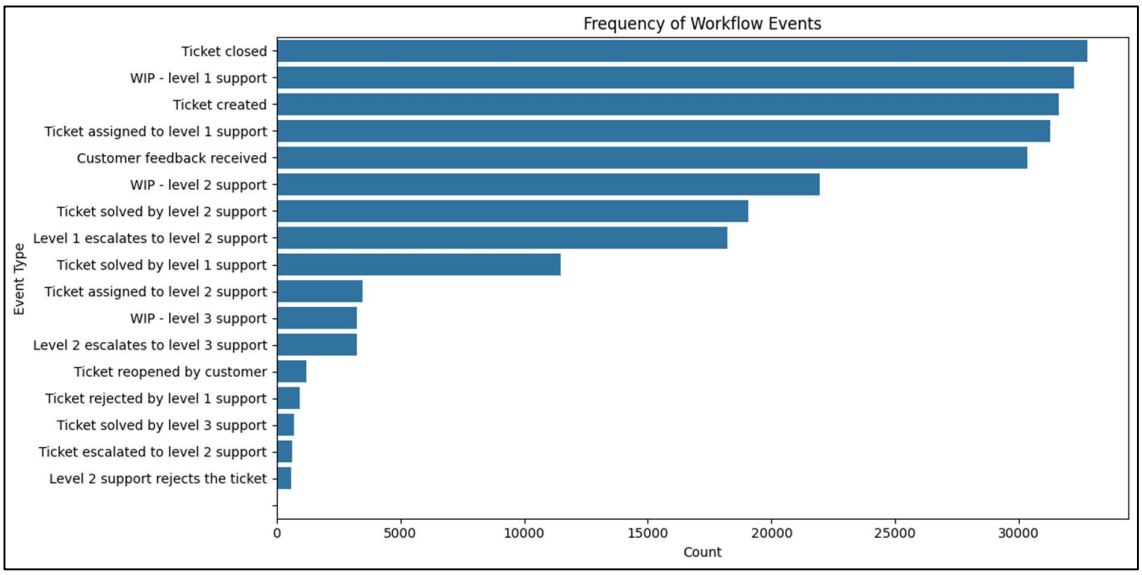

Figure 2. Frequency of Workflow Events

4.2 Priority-Based Workload Breakdown

The results compared the prevailing priority among unique cases to determine the relation between process urgency and workload. In Figure 3, most of the incidents belong to the following priorities: the medium category (approximately 15,700 cases), the Low priority (approximately 9,500), and the High priority (approximately 6,300). Although this is characteristic of the distribution of enterprise services, it also has significant implications for scalability.

Although there are few, the less-important workflows have more escalations, a smaller SLA window, tighter turnaround, and are more resource-intensive. This insight provides an apparent reason for differentiating resource allocation policies and autoscaling policies in consideration of priority levels.

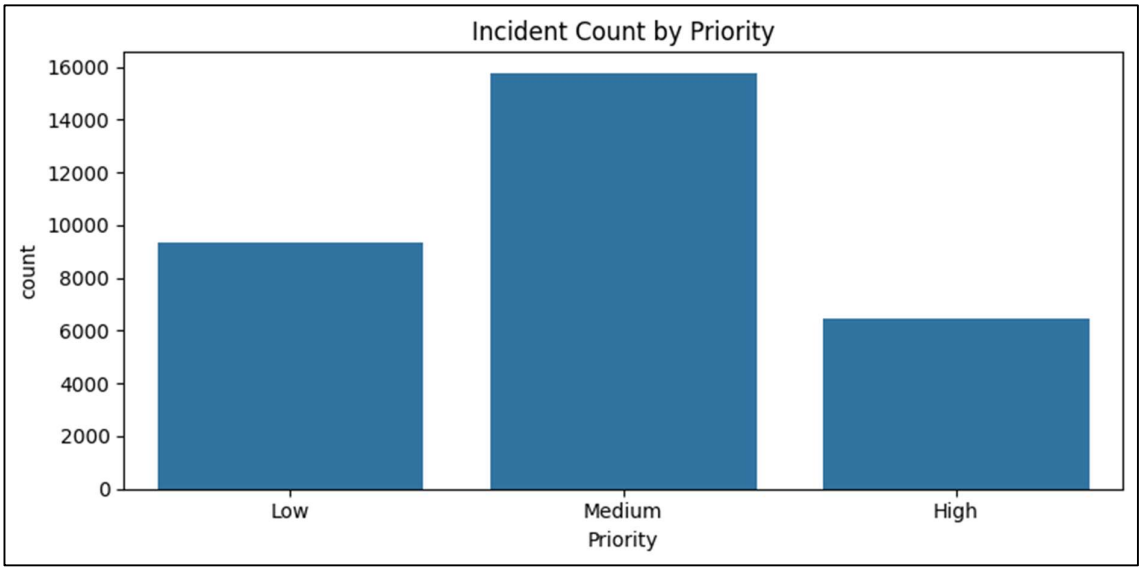

Figure 3. Incident Count by Priority

4.3 Daily Workflow Volume and Temporal Patterns

Figure 4 presents a daily event volume graph covering more than one year, which indicates substantial temporal variation. Daily events vary between 400 and 900, with occasional peaks showing the moments of a surge of workload, probably caused by batches, system updates, or exogenous factors. Such variability adds more clarity to why resource scaling should be elastic. The timetable indicates a consistently high-volume procedure over multiple months, and this observation implies that scaling up should not be responsive only to spikes but should also emphasise long-term trends. The Prophet's forecasting model is then fully justified because it allows modelling of such patterns and recommends proactive scaling.

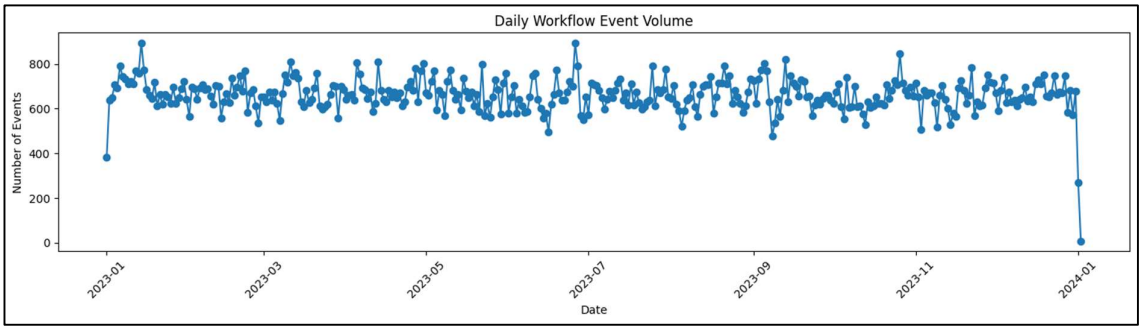

Figure 4. Daily Workflow Event Volume

4.4 Process Duration Distribution

Process completion time is an essential issue of system performance. Figure 5 shows a history of the number of cases corresponding to the duration in minutes derived by computing the difference in time between the first and last time stamp of each Case ID. Its central tendency was an average of about 901.9 minutes of case duration with a positively skewed distribution, and the median results were nearer to 700 minutes.

The tail is long after 2000 minutes, which indicates unresolved or undelayed escalations. These delayed cases lead to violations of the SLA and customer dissatisfaction. They also emphasise the role of load balancing according to forecasting to minimise the risks of such outcomes with congestion in the system.

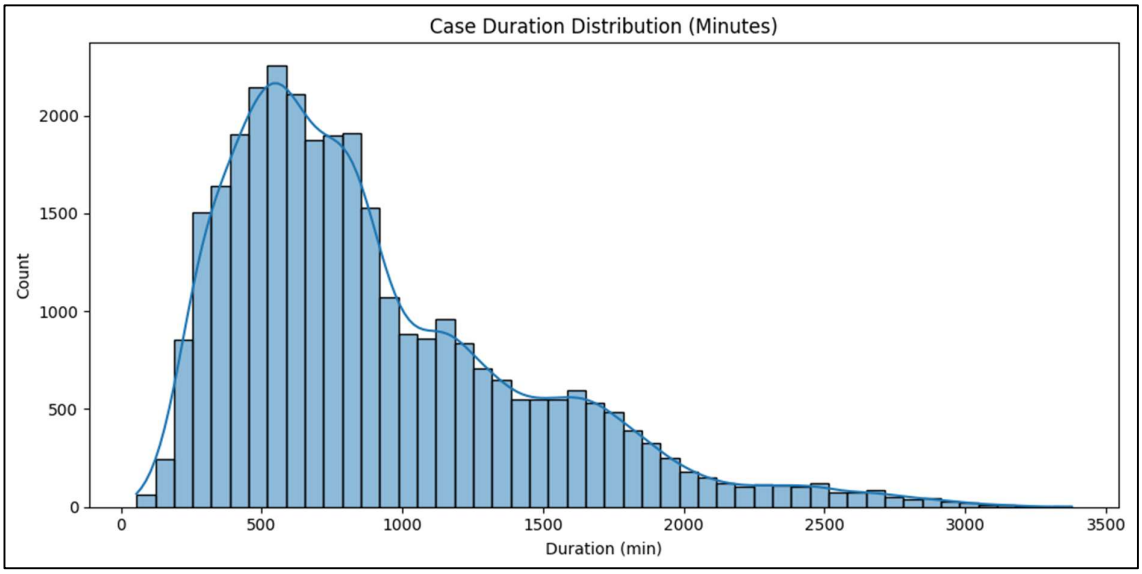

Figure 5. Case Duration Distribution

4.5 Satisfaction Trends and Priority Correlation

Figure 6 describes the dependence of customer satisfaction and the priority level using a box diagram. On a scale of 1 to 5, an average score across all the levels ranged between about 3.26, with a visible cluster between 3 and 4. The difference is, however, seen on the interquartile range, which indicates greater variance in high-priority cases containing more instances of low ratings (outliers lower than two scores).

The implication is that workflows with more pressure, i.e., with short deadlines and leaps, perform worse regarding the user experience, where the systems already use all available resources. These results argue that dynamic scaling, implemented based on the forecasted load, can enhance resolution timeliness and perceived service quality.

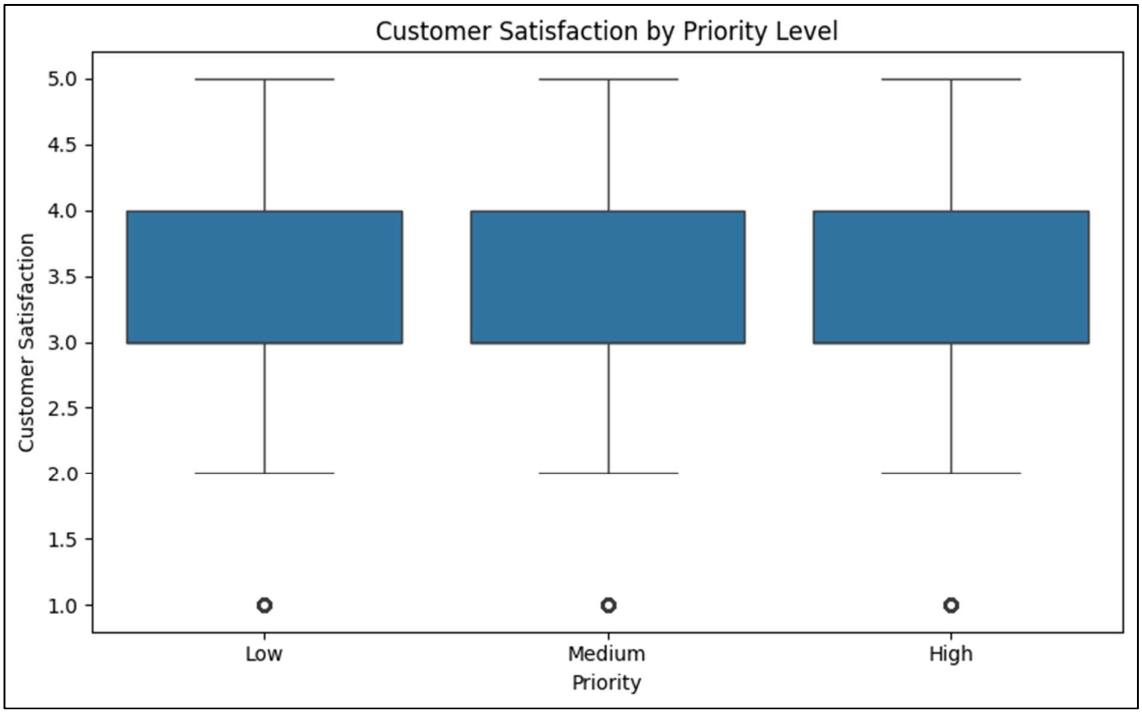

Figure 6. Customer Satisfaction by Priority Level

4.6 Forecasted Workload and Autoscaling Plan

The Prophet model was fitted using the totality of the time series as the daily event volumes to produce 7 7-day forecast as seen in Figure 7. The central prediction (yhat) had a limited range comprising 631 and 654 daily activities, with volatility indicators requiring confidence limits. All forecasted days were marked as experiencing autoscaling, considering a threshold of 400 daily events. As predicted by the Simulated Autoscaling Plan (see table below), proactive scaling actions would be required every day between December 31 and January 9. This active decision-making would guarantee the system's stability before it brought the SLA to its knees or ran out of resources.

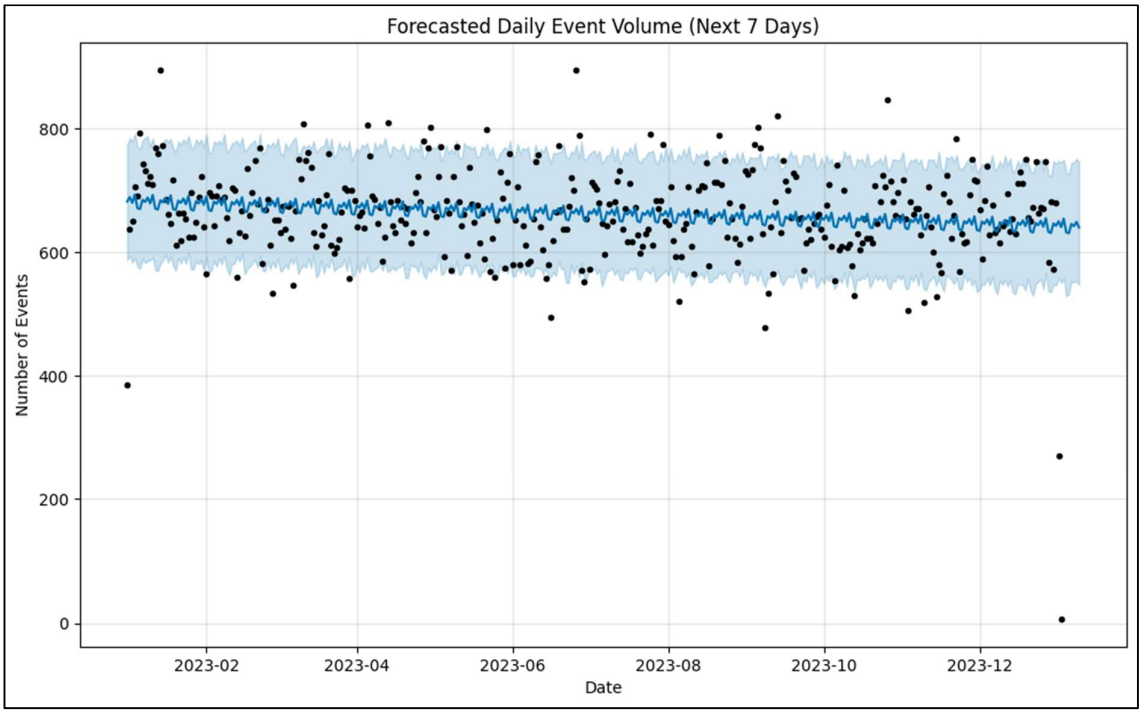

Figure 7. Forecasted Daily Event Volume

4.7 KPI Summary and Performance Indicators

Key performance indicators (KPIs) were calculated and presented in the table below to generalise the analysis results. The total volume dealt with was large, with a total of 31,588 cases being handled by the system. The mean case duration of 901.91 minutes shows that some optimisation is possible, particularly at high concurrency. Moderately complex workflows indicate a median of 8 events per case. The mean satisfaction rating of 3.26 is at the level of the benchmark. However, it does not exhaust itself with the opportunity to improve through a more equal distribution of work assignments (Table 1).

Table 1. KPI Performance Indicators

Metric	Value
Total Cases	31,588
Average Case Duration (min)	901.91
Median Events per Case	8
Avg Satisfaction Score	3.26

These findings support three significant points. First, the system's workload fluctuation is huge and persistent, hence the need for predictive load management. Second, a case's complexity and priority generate more satisfaction levels and SLA risks, and these cases must be orchestrated more intelligently. Third, combining AI-based Forecasting (Prophet) and autoscaling simulation provides a vivid example of how proactive scalability of BPM could be accomplished.

Altogether, the results prove the methodological morale behind the study and ensure that data-driven and intelligent process automation can deliver a much better performance for the enterprise system under fluctuating loads.

5. Discussion

The proposed research is an AI-assisted, forecast-based strategy to solve the scalability issue in workflow automation in enterprise Business Process Management (BPM) systems. The findings, which are based on a large-scale incident management log, show a great potential for making considerable advances in responsiveness, resource use, and customer satisfaction in an environment with high volumes of processes. Here, from these findings, we critically interpret them, benchmark their findings against the existing literature, place them in existing frameworks and think about the theoretical and practical implications of these findings.

5.1 Interpretation of Results

The EDA data mining method shows a distinct image: high event intensity characterises the enterprise workflows, a variation driven by priorities, and workload peaks. These trends are similar to the operational patterns observed in massive BPM environments (Hosny & Reda, 2021), in which workflow chains can be complex, have concurrent work processes, experience latency issues, and have grounds for resource contention. Specifically, identifying two events, namely, "WIP - level 1 support" and "Ticket closed", as the most frequent ones is in line with the research by Abbasi et al. (2024), which focuses on repetitive transactional flows as bottlenecks at the core of helpdesk and ITSM work processes.

The shape of the distribution indicates that the duration of cases (meaning it takes about 900 minutes) has a long tail, which means that there is a drag in the process and inefficiency of the system, particularly because this occurs during escalation. This observation concurs with the findings of (Lamghari et al., 2018), which indicated that the three leading causes of long cycle times in BPM situations are escalations, resource shortage, and queueing delays. We also observe that customer satisfaction scores are built on priority, similar to the argument of Yeniaras & Kaya (2022), who claimed that high-priority cases should be treated with special handling practices and dynamic task routing to prevent performance reduction efforts.

Most importantly, the presented forecasting model, Prophet-based, proposed in the study, offers a new and measurable approach to predicting the increase in workloads and promoting the decision-making regarding autoscaling. Such predictive ability covers a significant gap in current BPM implementations that remain mainly reactive. Earlier works by Khan et al. (2025) and Mavroudopoulos & Gounaris (2024) were centred on the elastic process execution and cloud-native BPM architecture. However, they were not formulated concretely, with the logic inspired by forecasting. This is the exact gap that our methodology fills, as we combine the application of AI-driven predictions with simulated autoscaling to introduce a ready-made method of launching preemptive autoscaling even before the resources are saturated.

5.2 Alignment with Existing Frameworks

The results are consistent with the ideas in Business Process as a Service (BPaaS) and Elastic BPM models. Dynamically allocating BPM computation resources using execution context. According to (Cocconi et al., 2017), elastic BPM promotes adaptive allocation of computational resources to BPM workflows. Although Schulte has suggested theoretical layers of architecture, including execution, control, and adaptation layers, we give a practical implementation by using Prophet and autoscaling logic. Adding forecasting to the design of orchestration decisions is a step towards taking elastic BPM beyond reactive to proactive.

In addition, the methodology is significant for ideas of Process-Aware Information Systems (PAIS) that focus on policy-sensitive data-driven process management (Rasouli, 2019). Specifically, our exploratory benchmarking using satisfaction and priority metadata, as well as the concept of dynamically allocating or moving resources according to this metadata, leads to more human-centric goal-aware BPM, as viewed in the literature of PAIS. The system is process-intelligent since it enriches process decisions through forward-looking intelligence.

Moreover, the proposed work makes conceptual mapping of the Monitor-Analyse-Plan-Execute of autonomic computing possible. In our case, the data logging and case duration analysis would be equivalent to monitoring, forecasting, and satisfaction correlation analysis, scale-out policy as the threshold-based scale-out policy for planning, and auto-scaling simulation to execution. Such devotion to the MAPE loop implicitly shows the framework's strength, but the use of AI extends it.

5.3 Theoretical and Practical Implications

This research adds to the new field of predictive process management. Whereas conformance checking, anomaly detection, and event correlation have already been studied in the past (Tariq et al., 2022; Zhong et al., 2022), the combination of forecasting and resource orchestration has not been investigated in such detail before. Prophet-based time-series modelling allows us to develop interpretable insights and introduces a time-complexity aspect into BPM, a massive leap from event-parametric, threshold-based adaptations.

In a practical sense, the study offers a duplicable plan of action amongst BPM software providers and cloud designers. Autoscaler APIs can be invoked by lightweight forecasting models like Prophet, integrated into modern orchestration platforms like Camunda, Zeebe, and Apache Airflow. The Horizontal Pod Autoscalers (HPA) of Kubernetes, or the concurrency levels of the AWS Lambdas, might change according to the estimated workload, making it cost-effective to scale with an SLA guarantee. Such integrations would enable BPM platforms to automatically re-configure infrastructure to expect demand, outage, latency and customer dissatisfaction.

Moreover, choosing real employment data helps the method appear practical and increase its external validity. In contrast to simulated or artificially produced traces, the ITSM dataset records subtle, reality-like variability of operations, such as escalations, cancellations, user delays, and satisfaction gaps, which is why the methodology can be used with real enterprise settings.

5.4 Novelty and Contribution

The main innovation of the given research is the combination of AI forecasting (Prophet) and autoscaling simulation to run BPM efficiently, which has not been demonstrated in the literature or solutions in the current field of BPM engines. Although other studies have

proposed performance-aware process models (Guruge & Priyadarshana, 2025) and dynamic SLA monitoring, most works were more concerned with detection than prediction. Conversely, the work presents a proactive and predictive decision layer capable of changing workflow orchestration under high-load scenarios.

The other innovation is the transition of customer satisfaction as a quality dimension in calibrating workload, most studies on optimising BPM centre on latency or throughput measures. Human-centred KPIS facilitate a balance between barrier and service experience, which fits the objectives of user-smart process automation (Bartlett et al., 2023).

5.5 Limitations and Risks

Regardless of these contributions, the suggested methodology has its limitations. First, the data under consideration is targeted towards ITSM and might not be enough to generalise all industries, including healthcare, manufacturing, or finance-related businesses. The lacking behaviour in the highly volatile, event-driven domains may demand a fine-tuning of the forecasting model, even though it shows encouraging results on seasonally consistent temporal data.

Additionally, autoscaling logic is not implemented in a production cloud environment but is simulated. In that way, the effects on real-time latesystem latency, start times, and cloud expenses are theoretical. It is suggested that future implementation be in a production-grade, serverless or container-based environment to test the proposed framework and assess the result of operational trade-offs. Finally, although Prophet is interpretable, it may not turn strong long-term temporal dependencies, as LSTM or Transformer-based models. Therefore, developing higher accuracy and sensitivity hybrid models could be pursued in volatile environments.

6. Conclusion and Recommendations

This paper also suggested and experimentally verified a new approach to proactive autoscaling based on AI in an environment of business processes execution based on historical ITSM logs and time-series forecasting with Prophet. The method presented how to apply the concept of predictive analytics to process-aware systems, such as predicting the volume of events, the number of workload peaks, and how to simulate decisions on autoscaling according to the expected demands. The framework is implemented in a sequence of logical steps, such as preparation of the datasets, tracing generation, statistical profiling, AI-based forecasting, and logic simulation of an autoscaling pattern that allows an enterprise to allocate resources and enhance service continuity proactively.

The hypothesis that workflow volumes in IT service management are subject to identifiable temporal patterns can be proved by the experimental findings. They can be predicted and adjusted accordingly, which may result in better operation resiliency during the peak ticket creation and escalation. Non-operational quality dimensions, such as customer satisfaction, are added to balance the system performance and service experience. The quality dimensions include both ticket duration and priority as operational metrics. Such emphasis on efficiency and quality represents a dramatic shift from threshold-based process orchestration solutions that operate reactively.

The difference of such work is that it is based on unifying AI-driven prediction and logic of scaling operations, which is embedded in a workflow informed by process mining. The study significantly contributes to the findings of other past studies, which remain largely descriptive

and reactive to problems. Instead, the present work is prescriptive and proactive, based on real-life data and validated simulations.

In the future, potential studies will test the use of this methodology in real time with cloud-native orchestration platforms, specifically Kubernetes HPA or AWS Auto Scaling, separate v5. Also, using more complicated temporal models (e.g., LSTM, Temporal Fusion Transformers) instead of Prophet can enhance flexibility in non-linear/anomaly-susceptible settings. A parallel area of extension continues to be cross-domain generalizability, SLA-based prioritisation integration, and cost optimisation modelling.

To conclude, this paper provides an extensible, explainable and future-proof scheme of intelligent workflow orchestration in digital businesses.

DECLARATIONS

Funding

This research received no external funding.

Conflicts of Interest

The authors declare no conflict of interest.

Ethical Approval

Not applicable.

Data Availability

The dataset used in this study, titled "Incident Management ITSM Dataset," is publicly available via Kaggle at: https://www.kaggle.com/datasets/albertopmd/process-mining-event-log-incident-management

Authors' Contributions

- **Conceptualisation & Methodology**: [Your Name]
- **Software, Modelling & Analysis**: [Your Name]
- **Writing – Original Draft & Visualisation**: [Your Name]
- **Review & Editing**: [Co-author Name (if any)]

Acknowledgements

The authors thank the open Kaggle community for providing high-quality real-world ITSM datasets that enabled reproducible experimentation.

REFERENCES

Abbasi, M., Nishat, R. I., Bond, C., Graham-Knight, J. B., Lasserre, P., Lucet, Y., & Najjaran, H. (2024). A review of AI and machine learning contributions in predictive business process management (process enhancement and process improvement approaches). *arXiv preprint arXiv:2407.11043*. https://doi.org/10.48550/arXiv.2407.11043

Ahmad, T., & Van Looy, A. (2020). Business process management and digital innovations: A systematic literature review. *Sustainability*, *12*(17), 6827. https://doi.org/https://doi.org/10.3390/su12176827

Baiyere, A., Salmela, H., & Tapanainen, T. (2020). Digital transformation and the new logics of business process management. *European journal of information systems*, *29*(3), 238-259. https://doi.org/https://doi.org/10.1080/0960085X.2020.1718007

Bartlett, L., Kabir, M. A., & Han, J. (2023). A review on business process management system design: the role of virtualisation and work design. *Ieee Access*, *11*, 116786-116819. https://doi.org/10.1109/ACCESS.2023.3323445

Bazan, P., & Estevez, E. (2022). Industry 4.0 and business process management: state of the art and new challenges. *Business Process Management Journal*, *28*(1), 62-80. https://doi.org/https://doi.org/10.1108/BPMJ-04-2020-0163

Castro, B. K. d. A., Dresch, A., & Veit, D. R. (2020). Key critical success factors of BPM implementation: a theoretical and practical view. *Business Process Management Journal*, *26*(1), 239-256. https://doi.org/https://doi.org/10.1108/BPMJ-09-2018-0272

Chaima, A., & Khebızı, A. (2022). A road-map for mining business process models via artificial intelligence techniques. *International Journal of Informatics and Applied Mathematics*, *5*(1), 27-51. https://doi.org/https://doi.org/10.53508/ijiam.1036234

Cocconi, D., Roa, J., & Villarreal, P. (2017). Cloud-based platform for collaborative business process management. 2017 XLIII Latin American Computer Conference (CLEI),

Faizal, A., & Aisyah, N. Innovative Approaches to Enterprise Database Performance: Leveraging Advanced Optimisation Techniques for Scalability, Reliability, and High Efficiency in Large-Scale Systems. *Reliability, and High Efficiency in Large-Scale Systems*. https://www.researchgate.net/profile/Ahmad-Faizal-11/publication/384695499_Innovative_Approaches_to_Enterprise_Database_Performance_Leveraging_Advanced_Optimization_Techniques_for_Scalability_Reliability_and_High_Efficiency_in_Large-Scale_Systems/links/67047b4cb753fa724d648b97/Innovative-Approaches-to-Enterprise-Database-Performance-Leveraging-Advanced-Optimization-Techniques-for-Scalability-Reliability-and-High-Efficiency-in-Large-Scale-Systems.pdf

Gadde, H. (2023). Self-Healing Databases: AI Techniques for Automated System Recovery. *International Journal of Advanced Engineering Technologies and Innovations*, *1*(02), 517-549. https://d1wqtxts1xzle7.cloudfront.net/119017129/517_549_ijaeti_2023-libre.pdf?1729403122=&response-content-disposition=inline%3B+filename%3DSelf_Healing_Databases_AI_Techniques_for.pdf&Expires=1753175563&Signature=ZCgXt66NEsYS72pxQKFOwuESzdI3qQubKS0B0f9uR1scdJmWsoUGHE~-MD244ZMQzAJqDxoMNm1dR3zFMBEI~GzKrCAz8vOCeB92BLj0uHgg~c5YeXtrtCqFfl~Y9LnwWRXZ04-jxv4CetA1Mu2-mBTbi5LYi3MbVBiqoyax5kVDjrjXsXUSZ5DxbqmRYwuMVzfXWaU0myFIczYBXeveVM95WwQzxurCnmG9oJpKz--~9~sDOg~6vUZAqHMhyJiZ4ZYOTGS2etvh~AZgwhUmzC7yWx-FgqoJAnttUWpVj63nj5ySCtralIX~jVCo6bVj-dUOiyNCsu6i9X862D95sQ__&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA

Gonzalez-Lopez, F., & Bustos, G. (2019). Business process architecture design methodologies–a literature review. *Business Process Management Journal*, *25*(6), 1317-1334. https://doi.org/https://doi.org/10.1108/BPMJ-09-2017-0258

Guruge, P. B., & Priyadarshana, Y. (2025). Time series forecasting-based kubernetes autoscaling using facebook prophet and long short-term memory. *Frontiers in Computer Science*, *7*, 1509165. https://doi.org/https://doi.org/10.3389/fcomp.2025.1509165

Hosny, A., & Reda, S. (2021). Characterising and optimising EDA flows for the cloud. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, *41*(9), 3040-3051. https://doi.org/10.1109/TCAD.2021.3120534

Jauhiainen, J. S. (2024). The Metaverse: Innovations and generative AI. *International Journal of Innovation Studies*, *8*(3). https://doi.org/https://doi.org/10.1016/j.ijis.2024.04.004

Khan, J., Liang, W., Mary, B. J., Hamzah, F., Taofeek, A., Mattew, B., Blessing, M., & Oluwaferanmi, A. (2025). Adaptive Cloud-Native Serverless ETL Systems: Breaking Barriers in Architecture for Data Processing Workflows. https://www.researchgate.net/profile/Beauden-John/publication/392654405_Adaptive_Cloud-Native_Serverless_ETL_Systems_Breaking_Barriers_in_Architecture_for_Data_Processing_Workflows/links/684c021269e22a0aa9d5f1fd/Adaptive-Cloud-Native-Serverless-ETL-Systems-Breaking-Barriers-in-Architecture-for-Data-Processing-Workflows.pdf

Kokala, A. (2024). Business process management: The synergy of intelligent automation and AI-driven workflows. *International Research Journal of Modernization in Engineering Technology and Science*, *6*, 12. https://doi.org/https://www.doi.org/10.56726/IRJMETS65186

Lamghari, Z., Radgui, M., Saidi, R., & Rahmani, M. D. (2018). A set of indicators for BPM life cycle improvement. 2018 international conference on intelligent systems and computer vision (iscv),

Liao, H.-T., Pan, C.-L., & Wu, Z. (2024). Digital transformation and innovation and business ecosystems: A bibliometric analysis for conceptual insights and collaborative practices for ecosystem innovation. *International Journal of Innovation Studies*, *8*(4), 406-431. https://doi.org/https://doi.org/10.1016/j.ijis.2024.04.003

Mavroudopoulos, I., & Gounaris, A. (2024). A comprehensive scalable framework for cloud-native pattern detection with enhanced expressiveness. *arXiv preprint arXiv:2401.09960*. https://doi.org/
https://doi.org/10.48550/arXiv.2401.09960

Narne, H. (2023). Revolutionising IT Operations: AI-Driven Service Management for Efficiency and Scalability. *INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS*. https://www.researchgate.net/profile/Harish-Narne-3/publication/386382748_Revolutionizing_IT_Operations_AI-Driven_Service_Management_for_Efficiency_and_Scalability/links/674fd785a7fbc259f1ab0944/Revolutionizing-IT-Operations-AI-Driven-Service-Management-for-Efficiency-and-Scalability.pdf

Pan, J., & Wei, Y. (2024). A deep reinforcement learning-based scheduling framework for real-time workflows in the cloud environment. *Expert Systems with Applications*, *255*, 124845. https://doi.org/https://doi.org/10.1016/j.eswa.2024.124845

Pourmirza, S., Peters, S., Dijkman, R., & Grefen, P. (2017). A systematic literature review on the architecture of business process management systems. *Information Systems*, *66*, 43-58. https://doi.org/https://doi.org/10.1016/j.is.2017.01.007

Ramos, E., & Arumugam, S. S. (2023). Process automation instantiation for intelligence orchestration. *Frontiers in the Internet of Things*, *2*, 1242101. https://doi.org/https://doi.org/10.3389/friot.2023.1242101

Rasouli, M. R. (2019). Intelligent process-aware information systems to support agility in disaster relief operations: a survey of emerging approaches. *International Journal of Production Research*, *57*(6), 1857-1872. https://doi.org/https://doi.org/10.1080/00207543.2018.1509392

Rinderle-Ma, S., Stertz, F., Mangler, J., & Pauker, F. (2023). Process mining—discovery, conformance, and enhancement of manufacturing processes. In *Digital Transformation: Core Technologies and Emerging Topics from a Computer Science Perspective* (pp. 363-383). Springer. https://doi.org/https://doi.org/10.1007/978-3-662-65004-2_15

Satyal, S., Weber, I., Paik, H.-y., Di Ciccio, C., & Mendling, J. (2017). AB-BPM: performance-driven instance routing for business process improvement. International Conference on Business Process Management,

Schäffer, E., Stiehl, V., Schwab, P. K., Mayr, A., Lierhammer, J., & Franke, J. (2021). Process-driven approach within the engineering domain by combining business process model and notation (BPMN) with process engines. *Procedia CIRP*, *96*, 207-212. https://doi.org/https://doi.org/10.1016/j.procir.2021.01.076

Schulte, S., Janiesch, C., Venugopal, S., Weber, I., & Hoenisch, P. (2015). Elastic Business Process Management: State of the art and open challenges for BPM in the cloud. *Future Generation Computer Systems*, *46*, 36-50. https://doi.org/https://doi.org/10.1016/j.future.2014.09.005

Singasani, T. R. (2019). Implementing PEGA for Enhanced Business Process Management: A Case Study on Workflow Automation. *Journal of Scientific and Engineering Research*, *6*(7), 292-297. https://www.researchgate.net/profile/Tejesh-Reddy-Singasani/publication/385086232_Implementing_PEGA_for_Enhanced_Business_Process_Management_A_Case_Study_on_Workflow_Automation/links/67150bda09ba2d0c760eac46/Implementing-PEGA-for-Enhanced-Business-Process-Management-A-Case-Study-on-Workflow-Automation.pdf

Szelągowski, M., & Berniak-Woźny, J. (2024). BPM challenges, limitations and future development directions–a systematic literature review. *Business Process Management Journal*, *30*(2), 505-557. https://doi.org/https://doi.org/10.1108/BPMJ-06-2023-0419

Szelągowski, M., & Lupeikiene, A. (2020). Business process management systems: evolution and development trends. *Informatica*, *31*(3), 579-595. https://doi.org/https://doi.org/10.15388/20-INFOR42

Tariq, Z., Charles, D., McClean, S., McChesney, I., & Taylor, P. (2022). Anomaly detection for service-oriented business processes using conformance analysis. *Algorithms*, *15*(8), 257. https://doi.org/https://doi.org/10.3390/a15080257

Ugwueze, V. (2024). Cloud Native Application Development: Best Practices and Challenges. *International Journal of Research Publication and Reviews*, *5*(12), 2399-2412. https://doi.org/: https://doi.org/10.55248/gengpi.5.1224.3533

Viriyasitavat, W., Da Xu, L., Bi, Z., & Sapsomboon, A. (2020). Blockchain-based business process management (BPM) framework for service composition in industry 4.0.

Journal of Intelligent Manufacturing, *31*(7), 1737-1748. https://doi.org/https://doi.org/10.1007/s10845-018-1422-y

Xue, Y., Fang, C., & Dong, Y. (2021). The impact of new relationship learning on artificial intelligence technology innovation. *International Journal of Innovation Studies*, *5*(1), 2-8. https://doi.org/https://doi.org/10.1016/j.ijis.2020.11.001

Yeniaras, V., & Kaya, I. (2022). Customer prioritisation, product complexity and business ties: implications for job stress and customer service performance. *Journal of Business & Industrial Marketing*, *37*(2), 417-432. https://doi.org/https://doi.org/10.1108/JBIM-08-2020-0404

Zhong, Z., Xu, M., Rodriguez, M. A., Xu, C., & Buyya, R. (2022). Machine learning-based orchestration of containers: A taxonomy and future directions. *ACM Computing Surveys (CSUR)*, *54*(10s), 1-35. https://doi.org/https://doi.org/10.1145/3510415