SAP ERP SOFTWARE FOR MANAGING LABORATORY SUPPLIES, PHARMACY STOCK, AND SURGICAL EQUIPMENT INVENTORY

Luka Kovac
Croatia

ABSTRACT

SAP ERP software supports the management of laboratory supplies, pharmacy stock, and surgical equipment inventory through an integrated enterprise platform. The system records item quantities, batch numbers, expiry dates, storage locations, suppliers, purchase details, and departmental usage. Real-time inventory tracking helps hospitals identify low stock, excess quantities, delayed deliveries, and urgent replenishment requirements. Automated workflows improve purchase requests, approvals, goods receipt, stock transfers, and invoice verification. Batch and serial number tracking strengthens medicine traceability and surgical equipment control. Integration with laboratories, pharmacies, operating theatres, procurement, finance, and warehouses improves data accuracy and supply coordination. Dashboards allow administrators to monitor consumption trends, inventory value, wastage, and supplier performance. Overall, SAP ERP can prevent shortages, reduce expired stock, control procurement costs, and improve hospital inventory efficiency.

keywords: SAP ERP; Laboratory Supply Management; Pharmacy Stock Control; Surgical Equipment Inventory; Healthcare Inventory Management.

1. Introduction

Enterprise applications depend on reliable diagnostics to detect runtime errors, failed workflows, API issues, slow transactions, and integration faults [1-2]. Logs are the main evidence source for understanding what happened before, during, and after an incident [3-4]. Effective application diagnostics require structured logging, traceability, contextual metadata, error classification, and operational visibility [5-7]. However, fixed logging strategies often create two problems. Low-detail logs may miss important causes [8], while high-detail logs may generate excessive storage cost and alert noise [9].

The main challenge is that diagnostic needs change during runtime [10]. A normal transaction may require only basic logs, but a failing workflow, repeated exception, or security-sensitive event may need deeper traces [11-12]. Adaptive logging improves this process by changing log depth according to system condition, transaction risk, and error behavior [13-15]. AI-assisted monitoring can identify when additional diagnostic detail is needed and when logging can return to normal levels [16-18]. This article proposes an adaptive logging framework for enterprise application diagnostics.

2. Methodology

The proposed framework captures log metadata from application servers, API gateways, databases, background jobs, authentication modules, and integration services [19-20]. It records event type, severity, timestamp, user role, transaction ID, request path, error frequency, response time, affected component, and workflow state [21-22]. These features are converted into diagnostic context profiles [23]. Adaptive logging requires context because the same event may be low-risk in one workflow but critical in another [24-25]. The logging controller classifies runtime states as normal, warning, error-prone, performance-risk, security-sensitive, workflow-critical, or incident-active [26]. Based on this classification, it adjusts log level, trace depth, parameter capture, correlation ID tracking, and diagnostic enrichment [27-28]. Lightweight scoring and policy rules are used so logging decisions remain fast and explainable [29-30]. Sensitive fields are masked before storage, and transaction-critical events are linked with workflow and database identifiers for easier diagnosis [31].

The framework is evaluated using simulated enterprise applications involving APIs, database workflows, authentication events, scheduled jobs, and service integrations [32]. Static logging is compared with adaptive logging under repeated exceptions, slow requests, failed jobs, API timeout, suspicious login, and workflow interruption scenarios [33-36]. Evaluation metrics include diagnosis time, log volume reduction, root-cause evidence quality, missed-event rate, storage overhead, alert usefulness, and administrator acceptance rate. Feedback from resolved incidents, ignored logs, and administrator notes is used to refine logging policies over time.

3. Results and Discussion

The results show that adaptive logging improves diagnostics by capturing richer evidence only when runtime risk increases. In the baseline condition, static logs either lack enough detail or produce large volumes of low-value records. With the proposed framework, detailed traces are activated for high-risk events and reduced during normal execution. The strongest improvement appears in intermittent failures, where deeper logs are needed only around the failure window. The discussion shows that adaptive logging must balance diagnostic depth with privacy and storage cost. Excessive logging can expose sensitive data or overload monitoring systems, while weak logging may hide failure causes. The main limitation is that log-level decisions depend on accurate runtime classification. Regular policy review and masking controls are necessary for safe long-term use.

4. Conclusion

This article presented an adaptive logging framework for enterprise application diagnostics. The framework adjusts log detail using runtime context, risk scoring, workflow state, and error behavior. It improves troubleshooting by increasing diagnostic evidence during abnormal conditions while reducing unnecessary log noise during normal operation. The study shows that adaptive logging can strengthen enterprise diagnostics when combined with structured metadata, privacy controls, and continuous incident feedback.

References

1. Kavuluri, V. V. R., Bandla, S., Gorumutchu, M. R., Mandapatti, J. K., & Jagadhabi, N. (2022). Execution Trace Forensics in Low-Code Platforms for Critical Workflows. Journal of Grid, Machines and Drives, 9, 1-8.
2. Kande, R., kanth Thottempudi, K., Kotla, C., Nithyanandam, S., & Anjuru, P. (2022). AI-Driven HIPAA Compliance Enforcement with Terraform and Azure Policy for Continuous Regulatory Monitoring. Emerging Digital Systems, 9, 29-36.
3. Keshireddy, S. R. (2022). Low-Latency Data Lake Ingestion Using Adaptive Partitioning and Query-Aware Layouts. Cross-Disciplinary Knowledge Systems, 9, 8-14.
4. Kavuluri, H. V. R. (2020). Feature Selection for Machine Learning Fault Prediction in Software Modules. Journal of Privacy-Aware Computing Systems, 7, 7-12.
5. Nithyanandam, S., & Anjuru, P. (2023). Fault-Tolerant Design for High-Availability EV Charging Networks. High-Performance Distributed Computing Review, 10, 1-8.

6. Jagadhabi, N., Mandapatti, J. K., Bandla, S., Gorumutchu, M. R., & Kavuluri, V. V. R. (2022). Semantic Degradation in Long-Lived Machine-Learned Data Pipelines. Transactions on Smart Electrical and Computational Systems, 9, 33-40.
7. Gorumutchu, M. R., Mandapatti, J. K., Jagadhabi, N., Kavuluri, V. V. R., & Bandla, S. (2024). Data Lineage Preservation Under Automated Schema Evolution. Journal of Grid, Machines and Drives, 11, 1-7.
8. Kavuluri, H. V. R. (2020). Software Reliability Prediction with Weibull Failure Models. Emerging Digital Systems, 7, 7-12.
9. Kavuluri, H. V. R. (2022). Data Quality Scoring for Streaming Pipelines with Hybrid Validation. Cross-Disciplinary Knowledge Systems, 9, 34-41.
10. Jagadhabi, N., Gorumutchu, M. R., Bandla, S., Mandapatti, J. K., & Kavuluri, V. V. R. (2023). Control Plane Saturation in Metadata-Driven Platforms. Journal of Privacy-Aware Computing Systems, 10, 36-43.
11. Kavuluri, V. V. R., Gorumutchu, M. R., Jagadhabi, N., Mandapatti, J. K., & Bandla, S. (2021). Schema Volatility Propagation in AI-Driven Data Architecture Pipelines. High-Performance Distributed Computing Review, 8, 1-7.
12. Keshireddy, S. R. (2024). Fault-Tolerant Data Engineering for Real-Time ETL in Multi-Source Enterprise Systems. Transactions on Smart Electrical and Computational Systems, 11, 9-15.
13. Keshireddy, S. R. (2019). Oracle APEX Integration with RESTful Services for Lightweight Enterprise Data Exchange. Journal of Grid, Machines and Drives, 6, 7-12.
14. Kavuluri, H. V. R. (2023). Change Data Capture for High-Volume Transaction Systems with Consistency Validation. Emerging Digital Systems, 10, 1-8.
15. kanth Thottempudi, K., Kande, R., & Kotla, C. (2024). Real-Time Hyper-Personalization at Scale: Contextual Bandit Architecture for Dynamic Customer Experience in Omnichannel Retail. Cross-Disciplinary Knowledge Systems, 11, 28-34.
16. Kavuluri, H. V. R. (2023). PostgreSQL for AI-Driven Applications: Performance Tuning for Machine Learning Pipelines. Perception, Navigation and Intelligent Devices, 10, 37-48.
17. Kavuluri, H. V. R. (2023). Metadata-Driven Engineering of Wind Turbine Logs for Condition Monitoring. Journal of Privacy-Aware Computing Systems, 10, 17-25.
18. Mandapatti, J. K., Bandla, S., Kavuluri, V. V. R., Gorumutchu, M. R., & Jagadhabi, N. (2024). State Explosion Risks in Rule-Driven Execution Engines. High-Performance Distributed Computing Review, 11, 1-8.
19. Anjuru, P., Nithyanandam, S., kanth Thottempudi, K., & Kande, R. (2024). Predictive Reliability Modeling in Distributed EV Charging Systems. Transactions on Smart Electrical and Computational Systems, 11, 1-8.
20. Anjuru, P., & Nithyanandam, S. (2021). Adaptive Fault Detection in EV Charging Cyber-Physical Systems. Journal of Grid, Machines and Drives, 8, 12-19.

21. kanth Thottempudi, K., Kande, R., & Kotla, C. (2024). Latency-Aware Decision Intelligence: Neural Architecture Search for Auto-Optimizing ML Inference Pipelines Under SLA Constraints in Retail Analytics. Emerging Digital Systems, 11, 29-36.
22. kanth Thottempudi, K., Kotla, C., & Kande, R. (2025). Cognitive Data Fabric: Foundation Model Architecture for Self-Organizing Enterprise Knowledge, Schema Evolution, and Zero-Query Insight Discovery in Retail Analytics. Cross-Disciplinary Knowledge Systems, 12, 29-36.
23. Nithyanandam, S., & Anjuru, P. (2021). Low-Latency Communication for Real-Time EV Charging Control Systems. Perception, Navigation and Intelligent Devices, 8, 31-38.
24. Anjuru, P., & Nithyanandam, S. (2023). Digital Twin-Based Predictive Maintenance for EV Charging Networks. Journal of Privacy-Aware Computing Systems, 10, 1-8.
25. Kavuluri, H. V. R. (2022). Adaptive Storage Tiering for Mixed Analytical and Operational Workloads in Data Engineering. High-Performance Distributed Computing Review, 9, 9-16.
26. Keshireddy, S. R. (2019). Spreadsheet-Based Record Migration to Oracle APEX Database Applications. Transactions on Smart Electrical and Computational Systems, 6, 1-5.
27. Kavuluri, H. V. R. (2020). Oracle Autonomous Database vs Open-Source Solutions: An Overview. Journal of Grid, Machines and Drives, 7, 26-39.
28. Mandapatti, J. K., Bandla, S., Kavuluri, V. V. R., Gorumutchu, M. R., & Jagadhabi, N. (2021). Error Propagation Topologies in AI-Assisted Construction Pipelines. Emerging Digital Systems, 8, 39-45.
29. Kavuluri, H. V. R. (2021). Distributed Feature Stores for Machine Learning Pipelines with Versioned Synchronization. Cross-Disciplinary Knowledge Systems, 8, 9-16.
30. Anjuru, P., & Nithyanandam, S. (2025). Adaptive Self-Learning Control for Autonomous EV Charging Operations. Perception, Navigation and Intelligent Devices, 12, 29-36.
31. Kotla, C., kanth Thottempudi, K., & Kande, R. (2021). Deep Learning Anomaly Detection for HIPAA-Regulated Azure Cloud Threat Monitoring. Journal of Privacy-Aware Computing Systems, 8, 28-34.
32. Keshireddy, S. R. (2019). Audit Trails in Oracle APEX with Database Triggers and Session Metadata. High-Performance Distributed Computing Review, 6, 1-6.
33. Nithyanandam, S., & Anjuru, P. (2025). Edge-Cloud Collaboration for Real-Time EV Charging Control. Transactions on Smart Electrical and Computational Systems, 12, 29-36.
34. Keshireddy, S. R. (2021). Oracle APEX Form Validation for Data Entry Error Reduction in Administrative Systems. Journal of Grid, Machines and Drives, 8, 1-6.
35. Keshireddy, S. R. (2020). Package-Based PL/SQL Architecture for Business Logic Separation in Oracle APEX. Emerging Digital Systems, 7, 1-6.
36. Kande, R., kanth Thottempudi, K., & Kotla, C. (2021). Autonomous DevSecOps: A Quantitative Maturity Model and ML-Driven Security Orchestration for Continuous Compliance. Cross-Disciplinary Knowledge Systems, 8, 1-8.