Intelligent Cloud Infrastructure Monitoring and Predictive Performance Management for Saudi Banking Systems under Vision 2030: Integrating OCI Observability, Automated Alerts and Resource Optimisation

Abstract

Saudi banking increasingly depends on interconnected digital channels, databases, payment gateways and resilient electricity supplies. Consequently, monitoring cloud infrastructure requires more than displaying processor utilisation or issuing isolated threshold alarms. This critical review examines how Oracle Cloud Infrastructure (OCI) observability can support predictive performance management through integrated metrics, logs, distributed traces, risk scoring and governed remediation. A structured appraisal of thirty scholarly publications from 2020–2025 is combined with primary technical and regulatory documentation. The synthesis identifies complementary strengths: tracing explains transaction dependencies, learning-based forecasts reveal emerging capacity constraints, and event correlation can reduce repetitive notifications. However, published models rarely establish transferability to regulated Saudi banking workloads, and prediction accuracy does not automatically imply better service continuity. The proposed architecture connects application service-level indicators with database behaviour, electricity supply, uninterruptible power, cooling and operational decision controls. Evaluation is framed around breach lead time, incident detection precision, recovery performance, resource efficiency and auditability rather than unsupported claims of achieved savings. For Vision 2030, the principal opportunity is not autonomous optimisation at any cost but evidence-led operations that preserve payment continuity, financial data protection and infrastructure resilience under explicit human accountability. The paper concludes with implementation controls, research gaps and a testable evaluation agenda for Saudi financial institutions.

Keywords: *Oracle Cloud Infrastructure; observability; predictive monitoring; Saudi banking; AIOps; alert governance; energy resilience; Vision 2030*

1. Introduction

Digital banking turns infrastructure disruptions into customer-facing failures within seconds. A delayed authentication request can interrupt mobile onboarding; an exhausted database connection pool can amplify payment latency; an unavailable network path can isolate an otherwise healthy application. Saudi Arabia’s Financial Sector Development Program intensifies the requirement for reliable digital services as banks and fintech providers expand electronic transactions, customer access and interoperable financial platforms. Research on national fintech development recognises

UNDER PEER REVIEW IN IJAR

the importance of infrastructure maturity and enabling regulation, but provides less detail about operational control mechanisms [21,22].

Conventional dashboards often treat host health as a proxy for business health. That assumption is weak in distributed systems: a virtual machine may report acceptable average CPU use while the ninety-ninth percentile response time deteriorates because of downstream contention. An industrial investigation of microservice tracing found that organisations frequently collect telemetry without exploiting its diagnostic potential [1]. Survey research similarly identifies difficulties in diagnosing service chains whose topology changes during deployment and scaling [9,13]. The more relevant question is whether monitoring can provide useful, timely, and defensible decisions about service risk.

Cloud performance management also has an electrical dimension. Compute availability depends upon utility feeds, transformers, switchgear, generators, uninterruptible power supply systems and environmental controls. Grid transition can reshape the disturbances against which critical facilities must engineer resilience; recent Saudi frequency-stability studies investigate these conditions without implying that ordinary outages are inevitable [19,20]. Banking observability must therefore join information-technology signals to facilities and supplier risk, even when those signals are supplied by different owners.

1.1. Research aim, objectives and contribution

The aim is to establish a technically credible, governance-aware framework for predictive cloud infrastructure monitoring in Saudi banking systems using OCI-native observability capabilities. Four objectives guide the analysis: distinguish monitoring, diagnosis and prediction; evaluate evidence for automated alerts and workload forecasting; integrate digital-service resilience with electrical dependencies; and define measurable implementation and assurance criteria for financial institutions. The contribution is a cross-layer decision architecture, not an empirical claim that OCI deployment has already improved any identified bank.

The paper answers three questions. Which telemetry best anticipates degraded banking transactions? Under what conditions can forecast-led action outperform threshold-based intervention? How should operational and electricity-related risks be incorporated without weakening regulatory control? Each answer is considered alongside uncertainty, data accessibility, intervention costs and the possibility that automation itself causes an incident.

2. Materials and methods

A structured critical literature review was undertaken rather than a statistical meta-analysis. Relevant scholarly studies published between 2020 and 2025 were identified through targeted

UNDER PEER REVIEW IN IJAR

searches of publisher records and academic indexes covering observability, distributed tracing, anomaly detection, incident management, autoscaling, workload forecasting, cloud energy demand, Saudi grid stability and financial technology. Bibliographic metadata and DOI landing records were checked for the thirty selected publications. This process supports a transparent synthesis but does not warrant claims of exhaustive database retrieval, dual-reviewer screening or pooled treatment effects.

Eligible works had a verifiable DOI, publication within the specified interval and an explicit contribution to at least one analytical strand. Foundational older methods were discussed conceptually without adding older works to the reference list. Preference was given to peer-reviewed journal articles, alongside selected archival conference papers where they provided concrete evidence on distributed operations. Studies that only marketed proprietary tools, lacked traceable methods or asserted performance improvements without identifiable evaluation settings were not treated as independent validation.

Evidence was organised into five dimensions: observability coverage and trace context; anomaly or root-cause methodology; prediction and resource orchestration; electricity and energy resilience; and the Saudi banking operating context. Each paper was examined for its design, workload, metrics, reported contribution and constraints. Particular attention was paid to whether experiments used synthetic injections, open benchmark traces or real production telemetry. This distinction matters because a model that recognises artificially injected anomalies may perform differently against rare payment incidents or unlabelled infrastructure events.

The analysis does not involve customer datasets, execute cloud experiments or test a live bank. Proposed service indicators and architectures are normative, and all numerical operating targets presented later are illustrative acceptance criteria rather than observed findings. OCI feature descriptions were cross-checked against Oracle's Monitoring and Observability documentation; Saudi governance considerations were cross-checked against the Saudi Central Bank's published cyber-security and continuity frameworks. Neither vendor documentation nor regulatory policy should be mistaken for independent evidence of predictive model effectiveness.

A methodological limitation is that academic publication dates, online-first dates and journal-volume dates do not always coincide. The bibliography adopts the reported scholarly publication year, and the DOI is retained as the stable identifier. Comparisons across studies therefore use their substantive evaluation period and technical design rather than assuming that adjacent publication years represent a controlled technology progression.

UNDER PEER REVIEW IN IJAR

3. Critical evidence synthesis

Distributed traces offer an important bridge from infrastructure symptoms to transaction experience. Bento and colleagues show how trace-derived dependency graphs can reveal anomalous services and workflows [2], whereas Kohyarnejadfard and colleagues use sequence analysis to detect unusual span behaviour [3]. ServiceAnomaly combines tracing with profiling signals [4]; TraceGra and graph variational approaches retain structural information overlooked by simple threshold series [5,6]. Their methodological variety is useful, but reported scores are not directly comparable because datasets, injected faults and labels differ. In a Saudi banking context, success should mean reliably localising the component responsible for a failed authorised payment, not merely identifying an unusual span.

Execution comparison techniques can recognise deviations from a baseline trace [7], and application-aware tracing can classify several categories of anomaly [8]. Yet an apparent performance deviation may reflect legitimate payroll-day demand, a software release or a planned failover. Reviews of incident management show the continued difficulty of correlating multiple alerts into actionable incidents [9,10]. Alert fatigue is therefore an outcome of weak context and decision processes as much as insufficient machine learning. The DevOps literature adds an important lesson: defect detection before production complements, rather than replaces, live observability [11].

Cloud-native systems expose additional non-stationarity through deployments, ephemeral services and dynamic scheduling [13]. Forecasting studies demonstrate that temporal and dependency-aware learning can model resource variations [16,29]. However, improved forecasting error on historical traces is not proof that an intervention will preserve banking service-level objectives. Model evaluation must include the number of preventable breaches, the interval between a warning and a real incident, and the damage from unnecessary changes. A low average error can conceal severe underprediction during rare spikes, precisely when payment services are most vulnerable.

Autoscaling studies further challenge simplistic optimisation. Pintye and colleagues demonstrate learning-based orchestration methods [14], while reinforcement-learning and feedback-control approaches study resource allocation under changing demand [24,25]. Edge-cluster research and a practical microservice autoscaler show that control decisions depend on system-specific constraints [26,27]. Serverless resource management creates related opportunities but adds cold-start and service-limit concerns [28]. None of these studies establishes that unconstrained automated scaling is appropriate for a core banking ledger or a protected settlement interface.

UNDER PEER REVIEW IN IJAR

3.1. Evidence quality and transferability

Table 1 distinguishes the principal evidence streams. Laboratory performance figures should not be transplanted to financial institutions as forecasts of achievable savings. Public workload traces seldom encode Saudi regulatory boundaries, cross-region recovery approvals, end-to-end payment correctness, confidential financial data or shared electrical failure modes. The strongest studies justify candidate mechanisms and experimental designs; they do not eliminate local validation. A credible adoption case should compare matched baseline and intervention periods, test seasonal workload changes and document adverse outcomes, including missed incidents and increased monitoring costs.

Table 1. Critical comparison of literature streams and their applicability to Saudi banking.

Evidence stream	Representative studies	Supported contribution	Transferability constraint
Tracing and diagnosis	[1–9,12]	Service dependencies and anomaly localisation	Topology drift; limited payment labels
Alerts and operations	[10,11]	Incident correlation and intervention workflow	Alert burden; uncertain causal attribution
Forecasting and control	[14–17,24–29]	Anticipatory scaling and capacity management	Synthetic traces; cost and SLO trade-offs
Electrical resilience	[18–20]	Power-dependency models and grid scenarios	Provider facility telemetry not public
Saudi digital finance	[21,22]	Policy and adoption context for reliable services	No controlled OCI banking outcomes

4. Proposed OCI observability architecture

The architecture in Figure 1 adopts a business-first hierarchy. At the top sit customer journeys: sign-in, balance enquiry, card authorisation, funds transfer, clearing and settlement. Each journey is assigned service-level indicators for availability, transaction success, tail latency and completion integrity. The middle layer maps these journeys to application services, integration gateways, queues, databases, network links and identity services. Infrastructure metrics become useful when they can be linked to this causal dependency model rather than interpreted in isolation.

OCI Monitoring supplies time-series metrics and alarms, including threshold or missing-data conditions; Notifications and Streaming provide delivery mechanisms. Logging and Logging Analytics support event investigation, while Application Performance Monitoring contributes user experience and distributed transaction views. Stack Monitoring and Database Management add dependencies and resource condition, and Operations Insights provides historical capacity analysis. These documented capabilities form building blocks, not a single automatically trained forecasting model. A bank must decide which telemetry is exported, joined, retained and evaluated, and which forecasts require additional controlled analytical components.

A practical collection design records timestamps, resource identifiers, deployment versions, environment, application journey, dependency edges and alarm ownership. It uses consistent correlation identifiers but avoids inserting account numbers, credentials or transaction payloads into ordinary observability streams. Sampling policy should preserve high-value error traces without indiscriminately retaining sensitive records. Centralised ingestion requires encryption, role-based access, region-specific data handling and auditable retention schedules. Integration with existing ticketing and security operations should distinguish performance incidents from possible intrusion or fraud signals [23].

The predictive tier transforms telemetry into engineered features: slope and variability of memory pressure, p95 and p99 latency, connection-pool occupancy, queue ageing, storage I/O waiting, error bursts, network retransmissions and anomalies in electrical or cooling state. It combines interpretable statistical baselines with multivariate forecasting where the extra complexity demonstrably helps. Dependency graphs support hypothesis ranking, but causal diagnosis requires change history, topology accuracy and controlled investigation; correlation alone cannot establish the root cause of failure [9,12,30].

Alerts should express a consequence, context and decision. A useful alert states the affected banking journey, predicted breach horizon, confidence calibration, dominant supporting signals, known maintenance windows, recommended action, rollback plan and accountable operator. Event deduplication must not suppress independent faults. The overall workflow moves from measurement to warning, service impact assessment, approval and post-action verification. Every automated outcome produces evidence for subsequent review rather than disappearing into an undocumented operations script.

UNDER PEER REVIEW IN IJAR

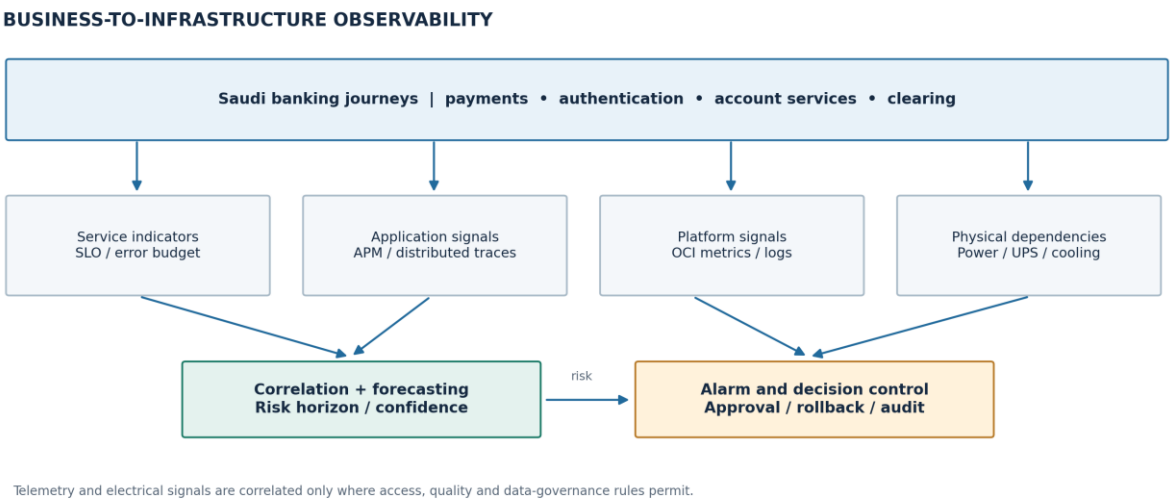


Figure 1. Original cross-layer observability architecture connecting banking service objectives, OCI telemetry, facilities dependencies and governed operational decisions. The diagram is conceptual; no production-bank topology is represented.

4.1. Forecast-led control rather than uncontrolled self-healing

The decision layer separates observation, recommendation and execution. Low-risk actions might increase diagnostic sampling, pre-warm an approved stateless service or schedule a capacity review. Higher-risk activities, such as resizing a production database, altering routing, suppressing security controls or initiating failover, require pre-authorised runbooks and explicit change authority. Workload forecasting can guide capacity reservations; it should not bypass segmentation, transaction integrity controls or tested recovery procedures.

A constrained control rule can be expressed as: choose an action only when expected avoided service-loss cost exceeds action cost, the probability of a service-level breach passes an approved threshold, and all security, recovery and electricity constraints remain satisfied. Prediction horizon must exceed approval plus execution time. When that inequality fails, the system should escalate uncertainty instead of promising prevention. Figure 2 shows these governance gates and illustrates how technical recommendations are rejected or deferred when operational safeguards cannot be demonstrated.

UNDER PEER REVIEW IN I

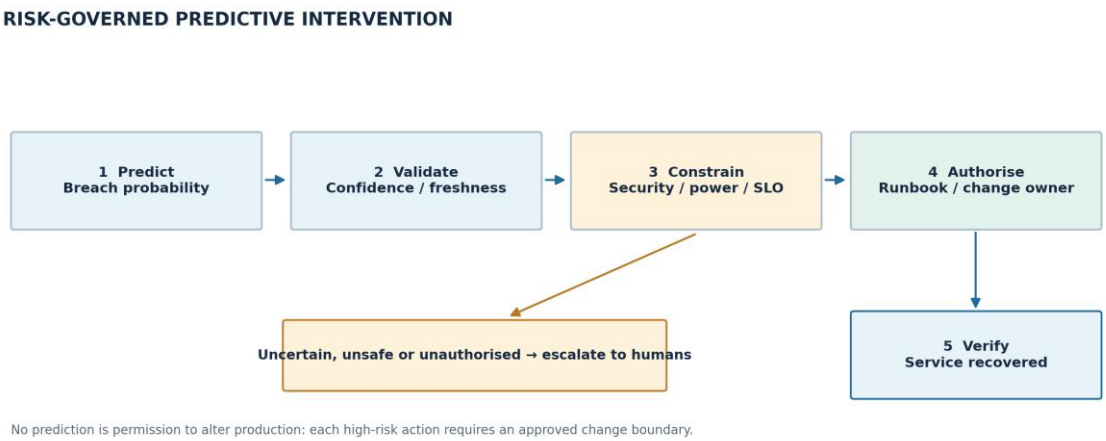


Figure 2. Original control-gate model for forecast-led interventions, including rejection or escalation where confidence, security, electricity continuity or change approval are insufficient.

5. Predictive analytics, alerts and evaluation

Forecast development should begin with a modest baseline: seasonal medians, exponentially weighted moving averages and supervised models trained only on past data. Neural or graph-based approaches may be appropriate for interacting services, but their operational benefit must be compared against simple alternatives. Resource-pattern forecasting research indicates that distribution shifts can undermine historical relationships [16]; the appropriate defence includes rolling validation, drift monitoring, retraining approvals and a monitored fallback model. Data leakage is prevented by separating training and test periods and excluding information that becomes available only after an incident.

A practical prediction target is the probability that a named service-level objective will be breached within a fixed horizon, such as the next fifteen minutes. That probability should be calibrated, not treated as a certainty. For rare events, conventional accuracy is misleading; the evaluation should report precision, recall, false alerts per operational shift, warning lead time and detection latency. Cost-sensitive assessment also measures customer impact minutes, operator effort and the consequences of a missed failure. Studies of alert management reinforce the importance of notification quality rather than alert volume [10].

Model testing needs credible negative controls. Forecasts should be evaluated on payroll peaks, Ramadan-related demand changes where applicable, public holidays, software deployments, partial network failures, database saturation and simulated loss of electrical supply. Tests should preserve the distinction between planned and unexpected events. Comparative trials should use shadow mode first: the predictor generates alerts and recommendations without changing production. Only after independent evaluation should a narrowly bounded subset of low-risk actions enter controlled automation.

UNDER PEER REVIEW IN IJA

Table 2 translates this approach into operational measurements. SLO breach minutes should be measured at the customer journey rather than solely from host uptime. Mean time to detect and restore are valuable but vulnerable to inconsistent incident definitions; the bank must record start, detection, mitigation and verified recovery timestamps. Capacity utilisation is useful only alongside headroom, contention and customer latency. Likewise, energy-saving claims require measured workload-normalised consumption and evidence that optimisation did not increase failure exposure.

Operational monitoring itself can fail. Missing metrics, clock drift, suppressed alarms, misconfigured roles, ingestion delays and regional network partitions can create false confidence. A second monitoring path should supervise telemetry freshness and the availability of notification destinations. Periodic game days can inject safe faults, validate escalation routes and check that dashboards survive the incident under test. For critical services, monitoring resilience should be treated as a separately assessed system requirement rather than an incidental platform feature.

Table 2. Proposed verification metrics and risk-governed operating decisions.

Indicator	Definition / unit	Required decision evidence	Safeguard
Journey availability	Successful transaction minutes / eligible minutes	Customer-visible impact by service	Independent synthetic checks
Tail latency	p99 end-to-end time, milliseconds	SLO trend and downstream correlation	Do not average away spikes
Prediction utility	Precision, recall and useful lead time	Matched baseline; false alerts per shift	Shadow deployment first
Incident response	Detection and verified restoration times	Time-stamped change and recovery record	Human approval for high risk
Resource efficiency	Capacity headroom and cost per transaction	Performance under normal and degraded load	Preserve recovery reserves
Electrical resilience	UPS, cooling and power-event state	Facilities events matched to service risk	Never expose provider control systems

REVIEW IN IJAR

6. Electrical infrastructure and resource optimisation

An electrically informed approach recognises that the cloud consumer may not directly observe the power systems underlying a provider's facility. Nevertheless, a bank can monitor its own data centres, branch connectivity, local networking, power events and contractual resilience commitments. Applicable facility telemetry includes grid-feed availability, UPS charge and transfer status, generator readiness, distribution-board alarms, temperature, cooling efficiency and maintenance windows. These signals can be joined with digital-service dependency records without implying privileged access to a cloud provider's internal facility controls.

Saudi grid research highlights the technical importance of frequency response as the generation mix evolves [19,20]. These studies inform resilience scenario design; they do not quantify the outage probability of a particular OCI region. Data-centre digital-twin research shows how electrical control logic may be modelled and tested before deployment [18]. The practical implication is to treat common-mode loss of power, cooling or telecommunications as a credible scenario, even where application services appear distributed across independent virtual hosts.

Resource optimisation consequently has two competing objectives: reduce unnecessary compute and electrical demand while protecting redundancy and recovery headroom. Predictive consolidation and energy-aware forecasting provide promising methods [15,17], but aggressive consolidation can reduce the buffer available for power anomalies or transaction surges. A bank should establish minimum resilient capacity, separate energy-saving actions from failover reserves and require load tests under degraded-site conditions. Workload migration is not a substitute for dual-path electrical engineering, reliable cooling or recovery exercises.

Where electricity-consumption data are available, the appropriate measure is energy per successful transaction alongside total kWh, availability and emissions-accounting assumptions. Power usage effectiveness alone does not capture application efficiency or transaction criticality. Capacity planning should consider summer cooling demand, exceptional settlement workloads and spare-power constraints. Green optimisation is defensible when it preserves service obligations; it is not defensible when savings are inferred from a forecast without metered verification.

7. Saudi banking governance and implementation

The Saudi Central Bank's published cyber-security framework treats cloud arrangements, data location, due diligence, access, continuity and auditability as important governance matters. Its continuity framework also emphasises round-the-clock availability of financial operations. Implementation must therefore begin with a documented applicability assessment of current rules, outsourcing approvals, contractual responsibilities and the bank's own risk appetite. The paper

UNDER PEER REVIEW IN IJAR

does not assume that selecting an OCI service, region or deployment pattern by itself establishes regulatory compliance.

A staged operating model is preferable. First, inventory critical journeys and map infrastructure ownership. Second, standardise service indicators, trace sampling and alert records. Third, build historical baselines and validate forecast performance in shadow mode. Fourth, introduce risk-tiered approval gates and incident playbooks. Finally, connect facilities resilience, supplier testing and quarterly management review. The programme should preserve a complete evidential chain from predicted condition to authorised action, observed outcome and subsequent model change.

Governance responsibilities cannot be delegated to an alerting algorithm. Infrastructure operations own telemetry quality; application teams own service semantics; security teams approve data handling; business continuity teams define recovery objectives; facilities engineers own physical-power controls; and accountable banking managers approve material production interventions. An independent assurance function should test whether models remain calibrated, recommendations are explainable and emergency overrides are effective. This separation reduces the risk that a technically impressive monitoring initiative creates a poorly governed decision system.

8. Research gaps and future directions

The largest evidence gap concerns transferability. Research datasets seldom contain synchronised metrics, traces, incident labels, facilities telemetry and regulator-relevant change records from a Saudi bank. Future work should establish privacy-preserving benchmark scenarios with defined workloads, failure injections, missing-data patterns and explicit recovery outcomes. Cross-organisation collaboration could publish reproducible methodology without releasing confidential payment records or topology details.

A second gap is causal evaluation. Most learning-based monitoring studies assess prediction or classification, whereas the practical question is whether a recommendation prevented a measurable customer-impact event. Prospective trials should compare baseline alarms, hybrid statistical prediction and advanced models under matched workloads, with predeclared intervention rules. Outcomes should include net avoided loss, operator workload, model drift, electricity consumption and any production harm caused by automation.

Third, electrical and cyber-physical dependencies remain weakly linked to application observability. Joint fault-tree and dependency-graph models could examine whether proposed multi-domain resilience survives shared power, cooling, identity or telecommunications failures. Fourth, research must address explainability under regulatory inspection: auditors need reconstructed decision pathways, not only anomaly scores. Human-in-the-loop designs should test

UNDER PEER REVIEW IN IJAR

whether explanations genuinely improve decisions during stressful incidents. These priorities create a research agenda directly aligned with trustworthy Saudi digital finance expansion.

Evaluation dossiers should document thresholds, comparison periods, data exclusions, model revisions and governance exceptions.

9. Conclusions

OCI observability provides an important foundation for understanding cloud resource condition, application performance and service dependencies, yet trustworthy predictive performance management requires further work. The literature supports trace-guided diagnosis, contextual alerting and forecast-assisted resource control, while warning against uncritical transfer of benchmark results to mission-critical banking. The proposed framework links customer service-level objectives to telemetry, prediction, governed action and post-event assurance.

For Saudi Vision 2030, dependable digital finance depends equally upon cyber governance, resilient electricity, operational expertise and disciplined evaluation. The recommended next step is a controlled, auditable pilot focused on selected banking journeys and a small number of high-consequence failure modes. Credible progress should be demonstrated through independently measured service outcomes, not the number of dashboards, algorithms or automated responses deployed.

References

[1] Li, B., Peng, X., Xiang, Q., et al. (2022). Enjoy your observability: an industrial survey of microservice tracing and analysis. Empirical Software Engineering, 27, 25. https://doi.org/10.1007/s10664-021-10063-9

[2] Bento, A., Correia, J., Filipe, R., et al. (2021). Automated analysis of distributed tracing: challenges and research directions. Journal of Grid Computing, 19, 9. https://doi.org/10.1007/s10723-021-09551-5

[3] Kohyarnejadfard, I., Aloise, D., Azhari, S.V., et al. (2022). Anomaly detection in microservice environments using distributed tracing data analysis and NLP. Journal of Cloud Computing, 11, 25. https://doi.org/10.1186/s13677-022-00296-4

[4] Panahandeh, M., Hamou-Lhadj, A., Hamdaqa, M. and Miller, J. (2024). ServiceAnomaly: an anomaly detection approach in microservices using distributed traces and profiling metrics. Journal of Systems and Software, 209, 111917. https://doi.org/10.1016/j.jss.2023.111917

[5] Chen, J., Liu, F., Jiang, J., et al. (2023). TraceGra: a trace-based anomaly detection for microservice using graph deep learning. Computer Communications. https://doi.org/10.1016/j.comcom.2023.03.028

UNDER PEER REVIEW IN IJAR

[6] Xie, Z., Xu, H., Chen, W., et al. (2023). Unsupervised anomaly detection on microservice traces through graph VAE. Proceedings of the ACM Web Conference, 2874–2884. https://doi.org/10.1145/3543507.3583215

[7] Meng, L., Ji, F., Sun, Y. and Wang, T. (2021). Detecting anomalies in microservices with execution trace comparison. Future Generation Computer Systems, 116, 291–301. https://doi.org/10.1016/j.future.2020.10.040

[8] Khanahmadi, M., Shameli-Sendi, A., Jabbarifar, M., et al. (2023). Detection of microservice-based software anomalies based on OpenTracing in cloud. Software: Practice and Experience, 53, 1681–1699. https://doi.org/10.1002/spe.3208

[9] Soldani, J. and Brogi, A. (2023). Anomaly detection and failure root cause analysis in (micro)service-based cloud applications: a survey. ACM Computing Surveys, 55, article 59. https://doi.org/10.1145/3501297

[10] Yu, Q., Zhao, N., Li, M., et al. (2024). A survey on intelligent management of alerts and incidents in IT services. Journal of Network and Computer Applications, 224, 103842. https://doi.org/10.1016/j.jnca.2024.103842

[11] Fawzy, A.H., Wassif, K. and Moussa, H. (2023). Framework for automatic detection of anomalies in DevOps. Journal of King Saud University – Computer and Information Sciences, 35, 8–19. https://doi.org/10.1016/j.jksuci.2023.02.010

[12] Wu, L., Tordsson, J., Elmroth, E. and Kao, O. (2020). MicroRCA: root cause localization of performance issues in microservices. NOMS 2020 IEEE/IFIP Network Operations and Management Symposium. https://doi.org/10.1109/NOMS47738.2020.9110353

[13] Deng, S., Zhao, H., Huang, B., et al. (2024). Cloud-native computing: a survey from the perspective of services. Proceedings of the IEEE, 112, 12–46. https://doi.org/10.1109/JPROC.2024.3353855

[14] Pintye, I., Kovács, J. and Lovas, R. (2024). Enhancing machine learning-based autoscaling for cloud resource orchestration. Journal of Grid Computing, 22, 68. https://doi.org/10.1007/s10723-024-09783-1

[15] Zheng, H., Lu, Y., Sun, Z., et al. (2024). Energy optimisation in cloud datacentres with MC-TIDE: mixed channel time-series dense encoder for workload forecasting. Applied Energy, 374, 123903. https://doi.org/10.1016/j.apenergy.2024.123903

[16] Li, J., Yang, D., Guo, H., Mo, X. and Wu, W. (2024). Forecasting resource usage pattern changes in clouds via contrast graph-evolution learning. Future Generation Computer Systems, 154, 373–383. https://doi.org/10.1016/j.future.2024.01.013

UNDER PEER REVIEW IN IJAR

[17] Rozehkhani, S.M., Mahan, F. and Pedrycz, W. (2024). Efficient cloud data center: an adaptive framework for dynamic virtual machine consolidation. Journal of Network and Computer Applications, 103885. https://doi.org/10.1016/j.jnca.2024.103885

[18] Galkin, N., Ruchkin, M., Vyatkin, V., et al. (2023). Automatic generation of data centre digital twins for virtual commissioning of their automation systems. IEEE Access. https://doi.org/10.1109/ACCESS.2023.3234804

[19] Alqahtani, S., Shaher, A., Garada, A. and Cipcigan, L. (2023). Impact of the high penetration of renewable energy sources on the frequency stability of the Saudi grid. Electronics, 12, 1470. https://doi.org/10.3390/electronics12061470

[20] Kahouli, O. (2025). Frequency stability analysis of power grid during photovoltaic generation in Hail region (Arabia Saudi). Frontiers in Energy Research, 12, 1323154. https://doi.org/10.3389/fenrg.2024.1323154

[21] Tayfor, A.E. (2025). Enhancing FinTech to support Saudi Arabia's Vision 2030: the role of the Saudi Central Bank in fostering a sustainable and innovative economy. WSEAS Transactions on Business and Economics, 22, 1622–1634. https://doi.org/10.37394/23207.2025.22.130

[22] Alrsheedi, A. and Iskandar, Y.H.P. (2025). Key factors influencing fintech adoption among Saudi banks: a conceptual framework. Humanities and Social Sciences Communications, 12, 1194. https://doi.org/10.1057/s41599-025-05532-1

[23] Jacob, S., Qiao, Y., Ye, Y. and Lee, B. (2022). Anomalous distributed traffic: detecting cyber security attacks amongst microservices using graph convolutional networks. Computers & Security, 118, 102728. https://doi.org/10.1016/j.cose.2022.102728

[24] Chen, X., Zhu, F., Chen, Z., et al. (2022). Resource allocation for cloud-based software services using prediction-enabled feedback control with reinforcement learning. IEEE Transactions on Cloud Computing, 10, 1117–1129. https://doi.org/10.1109/TCC.2020.2992537

[25] Zhang, Y., Hua, W., Zhou, Z., et al. (2021). Sinan: ML-based and QoS-aware resource management for cloud microservices. Proceedings of ASPLOS, 167–181. https://doi.org/10.1145/3445814.3446693

[26] Toka, L., Dobreff, G., Fodor, B. and Sonkoly, B. (2021). Machine learning-based scaling management for Kubernetes edge clusters. IEEE Transactions on Network and Service Management, 18, 958–972. https://doi.org/10.1109/TNSM.2021.3052837

UNDER PEER REVIEW IN IJAR

[27] Hossen, M.R., Islam, M.A. and Ahmed, K. (2022). Practical efficient microservice autoscaling with QoS assurance. Proceedings of HPDC, 240–252. https://doi.org/10.1145/3502181.3531460

[28] Li, Y., Lin, Y., Wang, Y., Ye, K. and Xu, C. (2023). Serverless computing: state-of-the-art, challenges and opportunities. IEEE Transactions on Services Computing, 16, 1522–1539. https://doi.org/10.1109/TSC.2022.3166553

[29] Patel, Y.S. and Bedi, J. (2023). MAG-D: a multivariate attention network based approach for cloud workload forecasting. Future Generation Computer Systems, 142, 376–392. https://doi.org/10.1016/j.future.2023.01.002

[30] Wang, T., Zhang, W., Xu, J. and Gu, Z. (2020). Workflow-aware automatic fault diagnosis for microservice-based applications with statistics. IEEE Transactions on Network and Service Management, 17, 2350–2363. https://doi.org/10.1109/TNSM.2020.3022028

UNDER PEER REVIEW IN IJAR