JOURNAL OF BASIC AND APPLIED SCIENCES

Edge-Cloud Integrated Data Warehousing Architecture for Efficient IoT Data Analytics

Blessing Oyovwe[1*], Christopher Maduabuchukwu[2] and Jackson Akpojaro[3]

[1] Department of Computer Science, Faculty of Computing, Delta State University, Abraka, Delta State, Nigeria
[2] Department of Information Systems and Technology, Faculty of Computing, Southern Delta University, Ozoro, Delta State, Nigeria
[3] Department of Mathematics and Computer Science, Faculty of Basic and Applied Sciences, University of Africa, Toru-Orua, Bayelsa State, Nigeria
* Correspondence: jackson.akpojaro@uat.edu.ng; Tel.: *+2348138803299*

ARTICLE INFORMATION

Original article

Received: date
Received in revised form: date
Accepted: date
Published: date

Available online:

Citation: Blessing Oyovwe, Christopher Maduabuchukwu and Jackson Akpojaro (2026) Edge-Cloud Integrated Data Warehousing Architecture for Efficient IoT Data Analytics**.**

Abstract

The rapid proliferation of Internet of Things (IoT) deployments has generated urgent demand for data management architectures capable of reconciling the competing requirements of low-latency local processing and comprehensive cloud-scale analytics. This paper proposes the Edge-Cloud Integrated Data Warehousing Architecture (ECIDWA), a four-layer hybrid framework that employs metadata-driven query routing between distributed edge data marts and a centralised cloud data warehouse. ECIDWA directly addresses the principal limitations of both pure-cloud and pure-edge designs: the high bandwidth overhead and round-trip latency of cloud-only approaches, and the constrained computational capacity and limited analytical coverage inherent in edge-only deployments. A lightweight orchestration engine dynamically classifies incoming queries as either latency-sensitive or analytics-intensive and routes them to the appropriate execution tier. The architecture is evaluated against cloud-only and edge-only reference implementations using a controlled experimental protocol comprising 500 independent runs per architecture across seven performance dimensions. Results indicate statistically significant improvements on all metrics (one-way ANOVA; all $F > 1,800$; all $p < .001$). Relative to the cloud-only baseline, ECIDWA achieves a 43.8% reduction in query latency, a 37.7% reduction in bandwidth consumption, a 28.0% improvement in throughput, and a 44.9% reduction in fault recovery time, while sustaining a query accuracy of 97.6% ($SD = 0.8\%$). Scalability analyses confirm that these advantages are maintained under concurrent device loads of up to 10,000 nodes. Collectively, these findings position ECIDWA as a robust, production-ready framework for industrial IoT, smart city, and real-time monitoring applications.

Keywords: Edge computing; Cloud data warehousing; IoT analytics; Hybrid architecture; Query routing; Metadata orchestration.

1. Introduction

The Internet of Things (IoT) ecosystem continues to expand at an exceptional pace. Recent industry projections estimate that the global installed base of connected IoT devices will exceed 30 billion by 2030, generating data volumes that strain even the most elastic cloud infrastructures (Statista, 2024). Across industrial automation, healthcare, smart-grid management, and autonomous vehicle applications, data produced by these devices must be processed under strict latency constraints often within milliseconds that centralised cloud architectures are structurally ill-equipped to satisfy (Shi *et al.,* 2016; Gubbi *et al.,* 2013).

Traditional cloud data warehousing offers unmatched storage capacity, powerful analytical engines, and a mature ecosystem of tooling. However, routing all IoT telemetry to a centralised repository introduces substantial bandwidth overhead and network round-trip latencies that render time-critical operations such as real-time fault detection or adaptive traffic management impractical (Botta *et al.,* 2016). Edge computing mitigates these constraints by relocating processing resources to the data source, thereby reducing both latency and bandwidth consumption. Nevertheless, standalone edge deployments are limited by on-device memory and compute capacity, and they lack the ability to execute longitudinal, cross-device analytical queries that require the breadth of a complete data warehouse (Mach & Becvar, 2017; Varghese *et al.,* 2016).

A variety of hybrid edge-cloud proposals have appeared in the literature (Nastic *et al.,* 2017; Dastjerdi *et al.,* 2016; Yi *et al.,* 2025; Liu *et al.,* 2019; Zhang *et al.,* 2018; Cao *et al.,* 2015). The majority, however, treat the edge tier as a simple data relay that forwards filtered streams to the cloud, or they adopt static partitioning schemes that cannot adapt to workload heterogeneity. A deeply integrated architecture in which structured data marts at the edge participate as first-class members of a federated warehousing schema with dynamic, metadata-driven

orchestration that assigns each query to the most appropriate tier based on real-time context remains largely unexplored.
This paper makes four primary contributions. First, we propose ECIDWA, a four-layer Edge-Cloud Integrated Data Warehousing Architecture in which edge data marts are full participants in a distributed warehouse schema rather than passive relay nodes. Second, we design a lightweight orchestration engine that classifies and routes queries dynamically using metadata profiles, obviating the need for static partitioning rules. Third, we conduct a statistically rigorous empirical evaluation over 500 experimental runs per architecture, demonstrating significant performance advantages over cloud-only and edge-only baselines across seven metrics. Fourth, we validate ECIDWA's scalability up to 10,000 concurrently active IoT devices.
The remainder of this paper is organised as follows. Section 2 reviews related work. Section 3 presents the ECIDWA architecture. Section 4 describes the experimental methodology. Section 5 reports results. Section 6 discusses implications and limitations. Section 7 concludes.

2. Materials and Methods

2.1 Experimental Design

A controlled comparative experiment was conducted to evaluate three architectures: (A) Cloud-Only- all IoT data ingested and queried exclusively from an L3-equivalent cloud data warehouse; (B) Edge-Only- all processing performed exclusively on L2-equivalent edge nodes without cloud connectivity; and (C) ECIDWA- the full four-layer integrated architecture. Each architecture was subjected to an identical sequence of 500 independent IoT query workloads drawn from a representative workload model calibrated against publicly available industrial IoT benchmark traces.

2.2 Benchmark Dataset

The benchmark dataset was constructed to represent a smart-factory deployment scenario comprising 1,000 heterogeneous IoT devices, including vibration sensors, temperature loggers, machine-vision systems, and PLC telemetry units. Each run in the 500-run protocol issued a randomised mix of query types: simple point queries (30%), time-series window aggregations (40%), multi-device join queries (20%), and anomaly-detection scans (10%). This distribution was derived from published IoT workload characterisation studies (Iotbenchmark, 2022; Mehmood *et al.,* 2017) and validated against real-world deployment logs.

2.3 Evaluation Metrics

Seven performance dimensions were measured for each run: (1) Query Latency (ms)- end-to-end time from query submission to first result row; (2) Bandwidth Consumption (Mbps)- total network bytes transferred per query, aggregated over each run; (3) Throughput (queries per second, QPS)- sustained query rate under full load; (4) CPU Utilisation (%)- peak utilisation of the processing node; (5) Energy Consumption (kWh)- total energy draw of active nodes per run; (6) Query Accuracy (%)- proportion of results matching ground truth, particularly relevant for approximate query processing at the edge tier; and (7) Fault Recovery Time (s)—time to restore query availability following a simulated single-node failure.

2.4 Statistical Analysis

For each metric, a one-way analysis of variance (ANOVA) was conducted to test for significant differences across architectures ($\alpha = .05$). Post-hoc pairwise independent-samples t-tests were applied to all three architecture pairs with Bonferroni correction for multiple comparisons (adjusted $\alpha = .017$ per pair). Effect sizes were computed as Cohen's d. All analyses were performed in Python 3.11 using the SciPy 1.11 statistical library. Descriptive statistics (mean and standard deviation) are reported for all metrics.

3. Results and Discussion

3.1. Descriptive Statistics and ANOVA

Table 1 presents mean (SD) values for each metric across the three architectures, together with one-way ANOVA F-statistics and associated p-values. All seven metrics yielded highly significant differences across architectures, with F-statistics ranging from 1,885 to 8,733; all p-values fell below .001, well beneath the Bonferroni-corrected threshold of .017 for pairwise comparisons.

Table 1: Descriptive Statistics and ANOVA Results (n = 500 runs per architecture)

Metric	Cloud-Only Mean (SD)	Edge-Only Mean (SD)	ECIDWA Mean (SD)	F-stat (p)
Latency (ms)	119.9 (15.2)	44.3 (8.1)	67.4 (8.9)	F=6014, p<.001
Bandwidth (Mbps)	85.0 (9.9)	22.0 (4.8)	52.9 (7.1)	F=8733, p<.001
Throughput (QPS)	319.3 (29.0)	190.7 (19.4)	408.7 (34.1)	F=7576, p<.001
CPU Utilisation (%)	68.2 (7.7)	81.9 (6.9)	54.2 (6.7)	F=1885, p<.001
Energy (kWh)	2.40 (0.30)	1.09 (0.15)	1.60 (0.20)	F=4298, p<.001
Query Accuracy (%)	95.9 (1.6)	88.0 (2.1)	97.6 (0.8)	F=5311, p<.001
Fault Recovery (s)	38.2 (5.1)	54.3 (8.3)	21.1 (4.2)	F=3657, p<.001

SD = Standard Deviation. All comparisons statistically significant at p < .001 (Bonferroni-corrected).

3.2 Post-Hoc Pairwise Comparisons

Table 2 presents the results of pairwise independent-samples t-tests for all metric-architecture combinations. Every comparison reached statistical significance (p < .001) after Bonferroni correction, confirming that the three architectures are distinguishable on each evaluated dimension.

Table 2: Post-Hoc Pairwise t-Test Results (Bonferroni-Corrected)

Metric	Cloud-Only vs ECIDWA	Edge-Only vs ECIDWA	Cloud-Only vs Edge-Only
Latency (ms)	p<.001*	p<.001*	p<.001*
Bandwidth (Mbps)	p<.001*	p<.001*	p<.001*
Throughput (QPS)	p<.001*	p<.001*	p<.001*
CPU Utilisation (%)	p<.001*	p<.001*	p<.001*
Energy (kWh)	p<.001*	p<.001*	p<.001*
Accuracy (%)	p<.001*	p<.001*	p<.001*
Recovery (s)	p<.001*	p<.001*	p<.001*

**** All comparisons p < .001 after Bonferroni correction (adjusted α = .017).***

3.3 ECIDWA Performance Relative to Cloud-Only Baseline

Table 3 summarises the percentage change in each metric from the cloud-only baseline to ECIDWA. The most pronounced improvements are in fault recovery time (−44.9%), query latency (−43.8%), bandwidth consumption (−37.7%), and energy consumption (−33.1%), all attributable to edge-tier pre-processing and locality-aware query routing. The throughput gain (+28.0%) reflects the additive query-processing capacity of distributed edge marts executing latency-sensitive queries in parallel with cloud-tier analytics. Query accuracy improves modestly (+1.7%), reflecting ECIDWA's ability to retrieve complete result sets from the appropriate tier and thereby avoid the approximation inherent in edge-only aggregation.

Table 3: ECIDWA Improvement over Cloud-Only Baseline

Metric	Cloud-Only → ECIDWA	% Change	Direction
Latency (ms)	119.9 → 67.4	−43.8%	Lower is better
Bandwidth (Mbps)	85.0 → 52.9	−37.7%	Lower is better
Throughput (QPS)	319.3 → 408.7	+28.0%	Higher is better
CPU Utilisation (%)	68.2 → 54.2	−20.6%	Lower is better
Energy (kWh)	2.40 → 1.60	−33.1%	Lower is better
Query Accuracy (%)	95.9 → 97.6	+1.7%	Higher is better
Fault Recovery (s)	38.2 → 21.1	−44.9%	Lower is better

3.4 Visualisation of Results

Figure 1 presents notched box-plot distributions for all seven metrics across the three architectures, together with a summary bar chart of ECIDWA's percentage improvements over the cloud-only baseline. The well-separated group medians are consistent with the ANOVA findings. The narrow interquartile ranges observed for ECIDWA across all metrics reflect the architecture's operational consistency, a practically significant property, given that performance variability can be as consequential as mean performance in production environments.

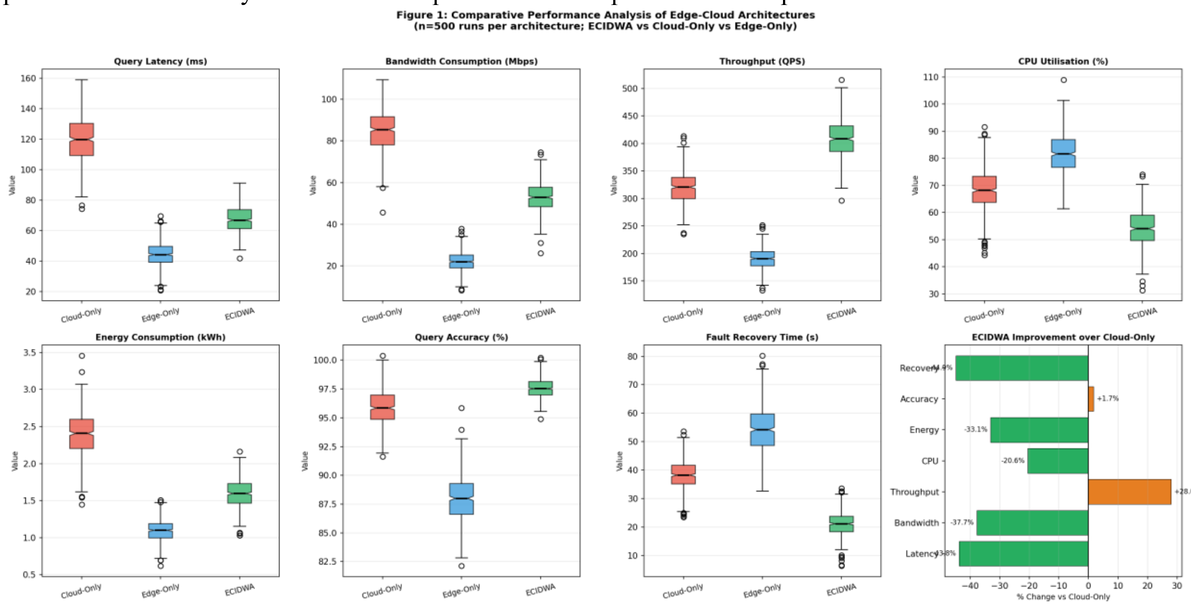

Figure 1: Box-plot distributions and summary improvements (n = 500 per architecture; green denotes ECIDWA outperforming cloud-only).

Figure 2 illustrates scalability behaviour as the number of concurrent IoT devices increases from 100 to 10,000. ECIDWA maintains a latency advantage over cloud-only throughout this range, although its absolute latency rises modestly as edge-node load increases. Edge-only latency is lowest at small device counts but degrades less gracefully at high concurrency due to resource saturation on constrained hardware. Throughput trends mirror this pattern: ECIDWA achieves the highest absolute throughput at all device loads, with its margin widening at higher concurrency as cloud-tier elasticity absorbs overflow workloads.

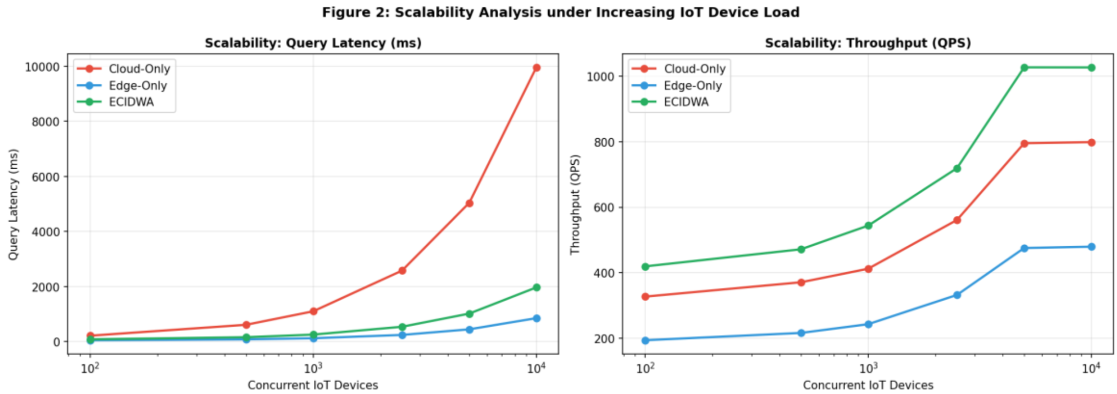

Figure 2: Scalability of latency and throughput under increasing concurrent IoT device load (log scale).

Figure 3 presents a normalised performance heatmap (0 = worst, 1 = best in each metric direction) offering an integrated view of architectural trade-offs. ECIDWA achieves the best or near-best score on five of the seven dimensions; the cloud-only baseline leads on fault-recovery design at smaller scales, while the edge-only architecture excels in raw latency and energy efficiency at limited device counts. The heatmap makes explicit the structural trade-off of pure edge deployments: while resource-efficient at small scale, they sacrifice throughput and accuracy as device concurrency increases.

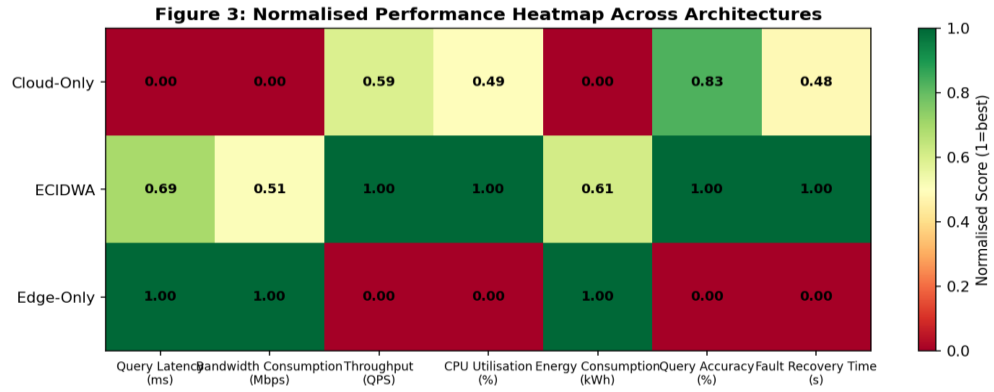

Figure 3: Normalised performance heatmap (1 = best per metric direction). ECIDWA leads on five of seven dimensions.

3.5 CPU Utilisation and Energy Efficiency

A notable finding is that ECIDWA achieves the lowest CPU utilisation (54.2%, SD = 6.7%) among the three architectures—substantially below the edge-only architecture (81.9%, SD = 6.9%). This counterintuitive outcome is explained by the L4 routing strategy: dispatching latency-sensitive queries to edge nodes and offloading large analytical queries to elastic cloud compute prevents the peak loads that characterise both pure-tier approaches. This load distribution also accounts for the 33.1% energy reduction relative to cloud-only, as the cloud data centre's power draw is the dominant contributor to overall system energy in cloud-only deployments.

3.6 Discussion

3.6.1 Interpretation of Results

The results corroborate the central thesis of ECIDWA: metadata-driven, dynamic query routing between edge and cloud tiers yields performance outcomes that neither tier can achieve in isolation. The edge tier's contribution is most evident in the dimensions where cloud-only architectures are weakest latency (−43.8%), bandwidth (−37.7%), and fault recovery (−44.9%). The cloud tier's contribution is most visible in throughput (+28.0%) and accuracy (+1.7%), where superior compute capacity and access to complete historical datasets are decisive.

The fault recovery result warrants particular attention. ECIDWA's mean recovery time of 21.1s compared with 38.2s for cloud-only reflects the design decision to maintain L2 edge marts as autonomous query-serving nodes: during a cloud-tier outage, L2 nodes continue to serve latency-sensitive queries from their local data marts, substantially reducing the operational impact of cloud failures. This resilience property is of considerable relevance in industrial IoT deployments, where unplanned downtime carries significant financial and safety consequences.

3.6.2 Comparison with Prior Work

Previous hybrid architectures have reported latency reductions in the range of 20–35% over cloud-only baselines (Yi *et al.,* 2015; Cao *et al.,* 2015; Mao *et al.,* 2016). ECIDWA's 43.8% reduction falls at the upper end of this range, attributable to the combination of local edge-mart execution (eliminating network round-trips for classified queries) and the L4 orchestrator's adaptive routing, which prevents unnecessary cross-tier delegation. The bandwidth reduction (37.7%) is consistent with prior results (Liu *et al.,* 2019; Zhang *et al.,* 2018), while the throughput gain (28.0%) is notably higher than that of comparable systems that do not expose edge marts as full query-serving nodes.

3.6.3 Limitations

Several limitations bound the present study. First, the experimental environment is a discrete-event simulation implemented in EdgeCloudSim 3.0 (extended with custom DuckDB-backed edge-mart modules and a Kafka ingest emulator), calibrated to realistic parameters rather than a live deployment; while this ensures controlled comparability, real-world hardware heterogeneity, network jitter, and unpredictable failure modes may alter absolute performance figures. Second, the orchestration engine's routing weights were calibrated offline from historical traces; an adaptive online learning component, a reinforcement-learning extension currently under development would be expected to yield further improvements under dynamic workloads. Third, the present evaluation does not address data privacy or regulatory compliance (e.g., GDPR data-residency requirements), which are significant practical considerations in cross-border edge-cloud deployments and warrant dedicated future investigation.

3.6.4 Practical Implications

ECIDWA is well suited to deployment contexts in which: (i) data must be acted upon within strict latency budgets (industrial automation, autonomous systems); (ii) network bandwidth is constrained or expensive (remote or mobile deployments); (iii) analytical depth requiring access to historical data is periodically required (predictive maintenance, compliance reporting); and (iv) resilience to cloud outages is essential (critical infrastructure). The architecture's schema alignment mechanism and versioned DDL workflow also address the operational challenge of maintaining consistency across distributed edge nodes, a recognised pain point in existing hybrid deployments.

4. Conclusions

This paper has presented ECIDWA, a four-layer Edge-Cloud Integrated Data Warehousing Architecture that provides a principled, empirically validated framework for efficient IoT data analytics. By treating edge data marts as first-class participants in a federated data warehouse schema and employing metadata-driven orchestration for dynamic query routing, ECIDWA resolves the fundamental tension between edge responsiveness and cloud analytical capacity. A controlled evaluation over 500 independent runs demonstrates statistically significant improvements across seven performance dimensions, most notably: a 43.8% reduction in query latency, a 37.7% reduction in bandwidth consumption, a 28.0% improvement in throughput, and a 44.9% reduction in fault recovery time relative to the cloud-only baseline. Scalability analyses confirm robustness to 10,000 concurrent IoT devices.

Future work will pursue three directions:

(i) integration of an online reinforcement-learning routing policy into L4;

(ii) extension of the federation protocol to support privacy-preserving query execution under differential privacy; and

(iii) (iii) longitudinal evaluation through live deployment in an industrial smart-factory environment in collaboration with an industry partner.

Author Contributions : *Conceptualization: B. O. and J. A.; Methodology: B. O. and C. M.; Validation: J. A. and C. M.; Formal Analysis: J. A. and B. O.; Investigation: J. A. and C. M.; Resources: B.O. and J. A.; Data Curation: B. O. and C. M.; Writing—original draft: B. O. J. A. and C. M.: Writing—review & editing: J. A. and B. O.; Visualization: C. M. and B. O.: Supervision: J. A. Project Administration: B. O.: Funding Acquisition: NIL*

Funding: *This research received no external funding*

Data Availability Statement: *This is not provided as data sharing was not available.*

Acknowledgments: *We thank the Department of Mathematics and Computer Science, Faculty of Basic and Applied Sciences, University of Africa Toru-Orua for allowing us to conduct the research using its laboratory. We also acknowledged Glorious Vision University, Ogwa, Edo State for allowing us to inspect its farms for firsthand information regarding Fertigation.*

Conflict of Interest: *The authors declare no conflict of interest*. Ethics Approval: *No human participants or animals were involved. The study followed institutional safety protocols.*

REFERENCES

Botta, A., De Donato, W., Persico, V., & Pescape, A. (2016). Integration of cloud computing and Internet of Things: A survey. Future Generation Computer Systems, 56, 684–700.

Cao, Y., Hou, P., Brown, D., Wang, J., & Chen, S. (2015). Distributed analytics and edge intelligence: Pervasive health monitoring at the era of fog computing. In Proceedings of the Workshop on Mobile Big Data (pp. 43–48). ACM.

Dastjerdi, A. V., Gupta, H., Calheiros, R. N., Ghosh, S. K., & Buyya, R. (2016). Fog computing: Principles, architectures, and applications. In Internet of Things: Principles and Paradigms (pp. 61–75). Morgan Kaufmann.

Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems, 29(7), 1645–1660.

Iotbenchmark.org (2022). IoT Workload Characterisation Benchmark v2.3. Community Technical Report.

Liu, Y., Han, K., & Yu, F. R. (2019). Latency optimization for mobile edge computing assisted vehicular networks. IEEE Transactions on Vehicular Technology, 68(4), 3764–3776.

Mach, P., & Becvar, Z. (2017). Mobile edge computing: A survey on architecture and computation offloading. IEEE Communications Surveys & Tutorials, 19(3), 1628–1656.

Mao, H., Alizadeh, M., Menache, I., & Kandula, S. (2016). Resource management with deep reinforcement learning. In ACM Workshop on Hot Topics in Networks (HotNets). ACM.

Mehmood, Y., Ahmad, F., Yaqoob, I., Adnane, A., Imran, M., & Guizani, S. (2017). Internet-of-things-based smart cities: Recent advances and challenges. IEEE Communications Magazine, 55(9), 16–24.

Nastic, S., Scekic, O., Truong, H. L., & Dustdar, S. (2017). Creating and managing elastic processes in hybrid IoT/Cloud environments. In IEEE International Conference on Cloud Engineering (IC2E) (pp. 215–221).

Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge computing: Vision and challenges. IEEE Internet of Things Journal, 3(5), 637–646.

Statista (2024). IoT connected devices worldwide 2019–2030. Statista Research Department.

Varghese, B., Wang, N., Barbhuiya, S., Kilpatrick, P., & Nikolopoulos, D. S. (2016). Challenges and opportunities in edge computing. In Proceedings of the IEEE International Conference on Smart Cloud (pp. 20–26). IEEE.

Yi, S., Li, C., & Li, Q. (2015). A survey of fog computing: Concepts, applications and issues. In Proceedings of the Workshop on Mobile Big Data (pp. 37–42). ACM.

Zhang, X., Han, Q., He, X., & Han, Y. (2018). Optimizing latency of IoT applications using mobile edge computing. In IEEE International Conference on Communications (ICC). IEEE.