

Advanced IoT-Based Autonomous Management System for Connected Infrastructure and Mobility Services

Denis Altuve¹, Karens Medrano², Kelman Belloso³

Daniel Castellanos⁴, Angel Mangandi⁵, Carlos Bran⁶, *Senior Member, IEEE*

^{1,3,4,6}Instituto de Investigación e Innovación en Electrónica (IIIE), Universidad Don Bosco (UDB), El Salvador
denis.altuve@udb.edu.sv, luis.belloso@udb.edu.sv, sm192656@alumno.udb.edu.sv, cbran@udb.edu.sv

^{2,5}Escuela de Computación, Universidad Don Bosco (UDB), El Salvador

karens.medrano@udb.edu.sv, cr221376@alumno.udb.edu.sv

Abstract—The accelerated growth of cities and communities has created the need for efficient systems to manage distributed infrastructures, facing challenges such as manual processes, limited connectivity, and low real-time data availability, especially in rural areas [1]. This paper proposes an advanced architectural framework for the AURA system (Unified Action for Tracking and Analysis), focusing on the orchestration of heterogeneous IoT data flows as its core scientific contribution. Unlike existing monolithic integrations, AURA provides a modular and scalable platform based on FIWARE, LoRaWAN, and MQTT [2]–[5], designed for real-time monitoring and tracking. To support deployments across urban–rural contexts, we rely on evidence of LoRaWAN performance in diverse scenarios [1], [6]. The novelty of the proposed architecture lies in its specialized IoT agent mediation and a hybrid storage scheme combining traditional and time series databases, which solves the interoperability and data persistence gaps found in standard IoT platforms. Experimental results demonstrate that AURA’s decoupled design outperforms traditional IoT architectures in terms of scalability and functional interoperability, establishing a comprehensive solution for intelligent connected systems with a clear projection toward predictive, evidence based decision making.

Index Terms—Application Programming Interface (API), Automation, Data Management, FIWARE, Internet of Things (IoT), IoT Architecture, IoT Platform, LoRaWAN, Message Queuing Telemetry Transport (MQTT), Monitoring, Smart City

I. INTRODUCTION

The growth of cities has intensified the need to manage distributed infrastructures such as transportation, water resources, and public services. However, many operations still rely on manual, reactive processes with limited connectivity, leading to operational inefficiencies and delayed responses to critical events.

This issue is further exacerbated in rural or remote areas due to the lack of robust communication infrastructure. Such limitations hinder the implementation of centralized solutions or continuous monitoring schemes [1].

Consequently, there is a recognized need for scalable and interoperable technological solutions that enable remote supervision and automation. Technologies like the Internet of Things (IoT) [7], FIWARE [4], LoRaWAN [8], and MQTT

[9] are key to facilitating connectivity in environments with limited infrastructure.

Within this framework, AURA systems have emerged as versatile platforms for environmental monitoring and home automation. By integrating real-time analytical capabilities, these systems enable precise tracking of variables like water and air quality, contributing to sustainability through efficient resource management [2], [3], [10], [11].

Despite these advances, common challenges such as poor interoperability and low scalability persist [12]. Recent studies emphasize the need to operationalize analytics/ML pipelines on FIWARE-based systems [5] and address stability constraints in large-scale IoT [13].

Existing platforms like ThingsBoard often lack a unified architectural model to orchestrate multi-protocol data with specialized hybrid storage. In response, AURA offers a unified approach that centralizes data tracking and analysis, improving operational efficiency and facilitating informed decision-making [14], [15].

Beyond large-scale monitoring, AURA stands out in home automation by improving energy efficiency, safety, and comfort. It utilizes contextual intelligence to anticipate user needs and optimize resource consumption [16], [17].

The scientific contribution of AURA lies in its decoupled architectural design, bridging the gap between raw multi-protocol sensing and autonomous reaction. This distinguishes it from traditional qualitative methods [18] or strictly sectoral proposals in water management [19].

This article presents the design of AURA as a benchmark technology platform, aligning with practical deployments where LoRaWAN shows robust performance in diverse terrains [6]. The proposal envisages a modular architecture that promotes interoperability between heterogeneous devices and distributed systems.

The proposed system is experimentally validated through a cloud-based deployment combined with a hybrid emulation environment, allowing the evaluation of its performance under realistic IoT operational conditions.

The methodological approach covers key stages including data acquisition, efficient transmission, and a hybrid orchestration mechanism for multi-protocol data integration and persistence. This work seeks to guide the development of adaptable infrastructures that optimize decision-making while providing a clear distinction between system implementation and research contribution.

II. SYSTEM ARCHITECTURE AND DESIGN

The development of the AURA system was based on an iterative and incremental approach, focused on the progressive integration of heterogeneous technologies and continuous validation in both simulated and real environments. To address the requirement of a clear scientific contribution, the methodology is divided into the technical framework (Implementation) and the architectural orchestration logic (Research).

The proposed system architecture ensures efficient data management by integrating components for collection, processing, and visualization. This structure is built upon open technologies from the FIWARE ecosystem and modern web tools, as illustrated in Fig. 1.

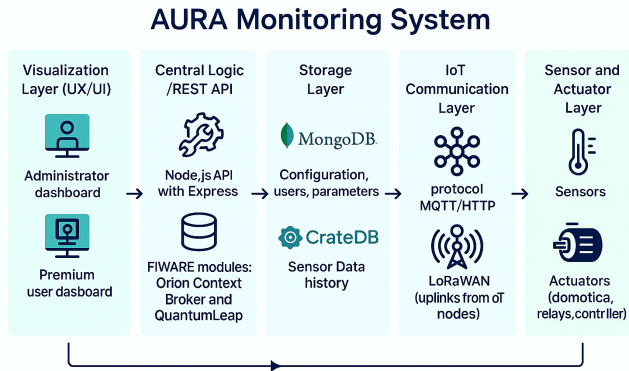


Fig. 1. Cloud architecture of the AURA monitoring system.

A. Dashboards and Visualization

We opted for a SPA (Single Page Application) architecture with Angular. Two distinct dashboards were defined according to user profiles (administrative and premium), as shown in Fig. 2. These interfaces incorporate geolocated maps for real-time tracking (Fig. 3), line charts for historical analysis (Fig. 4), and gauge charts for current sensor states (Fig. 5), allowing flexible and contextualized management.

B. Service Management

This module acts as the logical conductor using a RESTful API in Node.js. Integration with FIWARE modules, specifically Orion Context Broker and QuantumLeap, enables dynamic real-time context management and historical data persistence [4], [5]. The operational prefix management is visible in the database structure (Fig. 6).

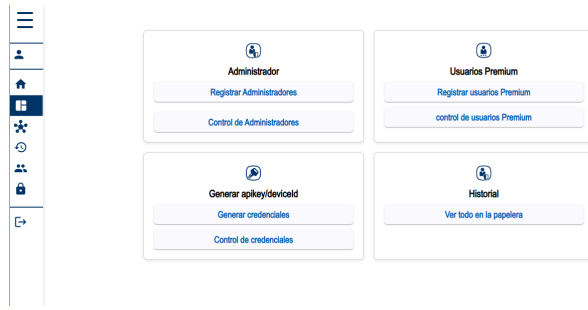


Fig. 2. Different user profiles and their corresponding access rights within the dashboard.

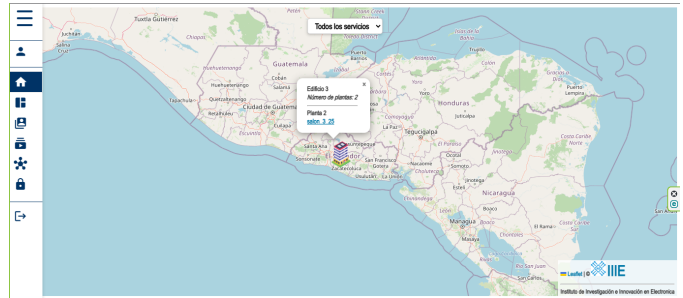


Fig. 3. Map representation of the location and status of sensors for real-time monitoring.

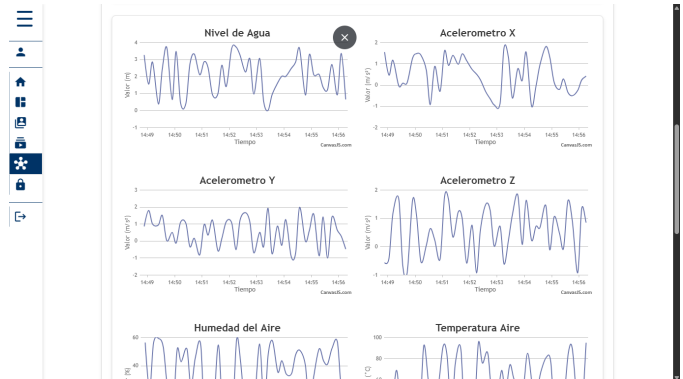


Fig. 4. Linear graphical representation for historical data analysis within a defined range in a 24-hour time interval.

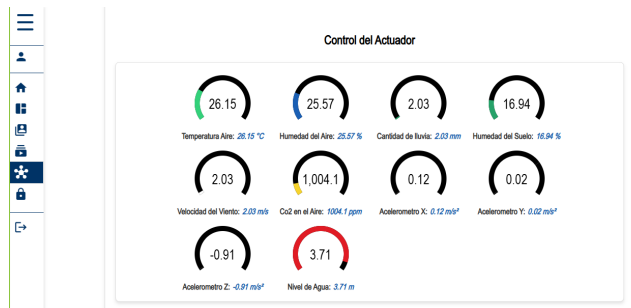


Fig. 5. Representation using a gauge of a singular sensor value for intuitive monitoring.

```

1 // 20250809103408
2 // http://localhost:8090/
3
4 {
5   "prefixes": [
6     "/v1",
7     "/",
8     "/route",
9     "/ping",
10    "/oauth2",
11    "/entities",
12    "/types",
13    "/subscriptions",
14    "/registrations"
15  ]
16 }

```

Fig. 6. Displaying the API prefix in CrateDB for contextual data management and system orchestration.

C. Sensor and Actuator Layer

This is the physical layer responsible for environmental interaction. It consists of sensors and actuators connected to IoT nodes that enable data collection and transmission to upper layers (Fig. 7). The operating status and device distribution are managed through modular monitoring, as shown in Fig. 8, ensuring the effectiveness of automated responses [2], [3].

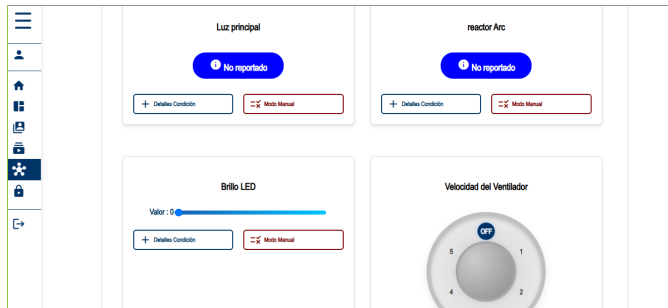


Fig. 7. Graphical representation of the distribution and status of sensors and actuators within the AURA system.

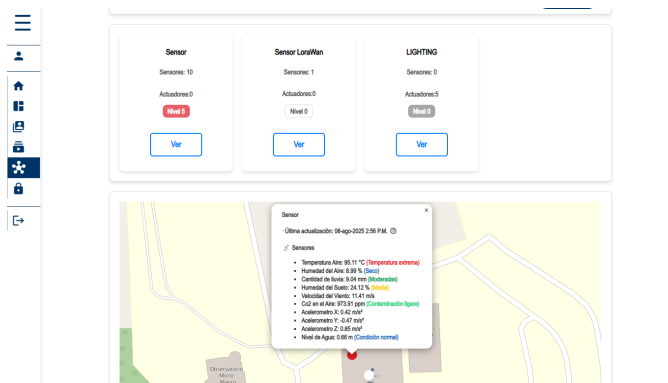


Fig. 8. Display of sensor and actuator modules, showing their operating status and the number of devices per module.

D. Hybrid Interoperability and Research Contribution

The research contribution that distinguishes AURA from standard integrations like FIWARE or ThingsBoard lies in the dual logic orchestration designed to bridge the gap between raw multi-protocol sensing and autonomous reaction.

E. Semantic Mediation Logic

Specialized IoT Agents were developed to solve the "binary to semantic" gap. Unlike generic implementations, these agents transform raw data from LoRaWAN and MQTT into standardized NGSI-LD entities. This ensures that the infrastructure acts as a single interoperable "digital twin," enabling robust connectivity in rural environments [1], [6], [9].

F. Hybrid Persistence Orchestration

AURA solves the performance bottleneck of large-scale IoT by separating metadata from telemetry. MongoDB handles semi-structured configuration data, while CrateDB manages time-series persistence through QuantumLeap (Fig. 9). The scientific novelty resides in the synchronization mechanism that maintains contextual consistency between the asset's state and its historical record, meeting the requirements of modern monitoring and control systems [20].

entity_id	entity_type	time_index	fiware_servicepath	_original_ngsi_entity	instanceid	sensors	location
urn:ngsi-ld:Sensor:000001	000001	1744103170001 (2025-08-08T17:10:00.000Z)	/fiware-ld/v1/entities/2/000001	MALL	urn:ngsi-ld:Sensor:000001	- accelerometer : 0.58 - airflow : 56.58 - humidity : 54.58 - light : 43.74 - location : 43.74 - temperature : 2.58 - velocity : 4.73 - altitude : 99.62	Type: Po Coordinate
urn:ngsi-ld:Sensor:000002	000002	1744103170002 (2025-08-08T17:10:00.000Z)	/fiware-ld/v1/entities/2/000002	MALL	urn:ngsi-ld:Sensor:000002	- accelerometer : 0.57 - airflow : 54.58 - humidity : 54.58 - light : 43.74 - location : 43.74 - temperature : 2.58 - velocity : 4.73 - altitude : 99.62	Type: Po Coordinate
urn:ngsi-ld:Sensor:000003	000003	1744103170003 (2025-08-08T17:10:00.000Z)	/fiware-ld/v1/entities/2/000003	MALL	urn:ngsi-ld:Sensor:000003	- accelerometer : 0.57 - airflow : 54.58 - humidity : 54.58 - light : 43.74 - location : 43.74 - temperature : 2.58 - velocity : 4.73 - altitude : 99.62	Type: Po Coordinate
urn:ngsi-ld:Sensor:000004	000004	1744103170004 (2025-08-08T17:10:00.000Z)	/fiware-ld/v1/entities/2/000004	MALL	urn:ngsi-ld:Sensor:000004	- accelerometer : 0.57 - airflow : 54.58 - humidity : 54.58 - light : 43.74 - location : 43.74 - temperature : 2.58 - velocity : 4.73 - altitude : 99.62	Type: Po Coordinate

Fig. 9. Visualization of modules integrated into CrateDB for historical data storage and time-series analysis.

G. Data Ingestion Workflow and Sequence Logic

To understand the dynamic behavior of the AURA platform and justify the observed end to end latency, it is essential to trace the sequential flow of information from the physical environment to the persistent storage. The data ingestion process follows a multi-stage pipeline, ensuring that each telemetry packet is validated, contextualized, and historized, as illustrated in Fig. 10:

The data flow begins with the Sensing and Uplink phase, where IoT nodes transmit encrypted packets via LoRaWAN to a Network Server through a gateway. Subsequently, during the Semantic Translation stage, the FIWARE IoT Agent retrieves the payload via MQTT and performs binary-to-semantic mapping to normalize the data according to the NGSI-LD standard.

Once transformed, Context Management is executed via an update request to the Orion Context Broker. This enables the sensor's "Digital Twin" to reflect its status in real time and activates all relevant subscriptions for downstream services.

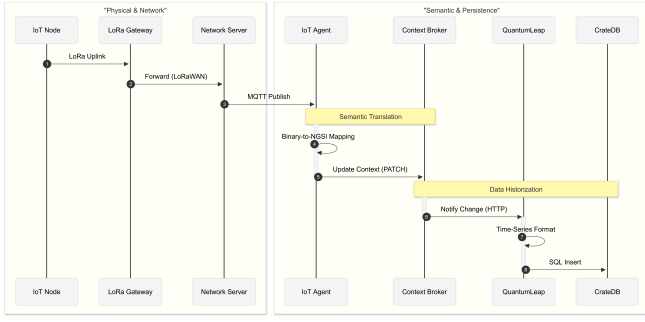


Fig. 10. Sequential workflow of data ingestion from physical sensing to historical persistence.

This event initiates Historization through the QuantumLeap service, which intercepts the context change and manages the persistence logic. Finally, the process concludes with storage in CrateDB, where information is recorded as a time series for historical analysis and dashboard visualization.

This five-stage pipeline explains the average end-to-end latency of 850 ms. Each hop introduces a processing overhead necessary to achieve total decoupling between the physical sensing layer and the analytical storage layer—a trade-off that ensures high scalability and interoperability in heterogeneous environments.

III. EXPERIMENTAL METHODOLOGY AND VALIDATION

To address the requirement of a clear scientific contribution, the evaluation of AURA is based on an iterative validation in both simulated and real environments, focusing on architectural stress-testing and operational feasibility.

A. Experimental Setup and Hybrid Validation

The performance evaluation followed a hybrid validation methodology combining physical deployment and large-scale emulation. The core software stack was deployed in a cloud environment using an AWS EC2 instance (8 vCPUs, 32 GB RAM, ARM64 architecture) to assess system behavior under high-concurrency data ingestion conditions.

To ensure both physical signal integrity and scalability validation, the data source was structured as a dual-layer configuration: (i) *Physical Layer*, consisting of a pilot deployment of five physical IoT nodes (Heltec LoRa V3 and ESP32-based devices) used to validate end-to-end telemetry transmission from sensor acquisition to semantic persistence; and (ii) *Emulation Layer*, in which the experimental environment was augmented with 42 emulated nodes replicating the same firmware logic and LoRaWAN duty-cycle constraints as the physical devices.

B. Performance Metrics

To evaluate the operational efficiency of the AURA framework, a set of quantitative performance metrics was defined. These indicators enable a reproducible and objective assessment of system behavior under varying load conditions.

End-to-end latency (L) was defined as the total elapsed time between the generation of a data packet at the sensor node and its successful registration at the context broker. For each transmitted message i , latency was computed as

$$L_i = T_{server,i} - T_{sensor,i}, \quad (1)$$

where $T_{sensor,i}$ denotes the packet creation timestamp at the IoT node and $T_{server,i}$ represents the corresponding arrival timestamp at the cloud backend. The average latency (L_{avg}) was then calculated over $N = 30$ consecutive measurement windows.

To assess transmission stability, update jitter (J) was used to quantify deviations from the nominal 30-s update interval. Jitter was computed as the absolute variation in the inter-arrival time between consecutive packets:

$$J = |(T_i - T_{i-1}) - \Delta t_{ref}|, \quad (2)$$

where $\Delta t_{ref} = 30$ s represents the reference transmission period.

In addition, network throughput (C) was estimated to characterize the bandwidth demand per device, accounting for both effective payload size and protocol overhead introduced by LoRaWAN and MQTT. Throughput was calculated as

$$C = \text{Size}_{msg} \times \text{Freq}_{tx}, \quad (3)$$

where Size_{msg} corresponds to the average message size (222 bytes) and Freq_{tx} denotes the transmission frequency (2 messages/min).

IV. RESULTS AND DISCUSSION

This section reports a quantitative performance assessment of AURA using an operational deployment of the full software stack (FIWARE Orion Context Broker, IoT Agents, QuantumLeap, MongoDB/CrateDB, Grafana, and the custom API and web dashboard). Measurements focus on latency, update rate, scalability, network consumption, and resource utilization, which are key indicators for practical IoT deployments in constrained-connectivity environments.

To contextualize the contribution, Table I summarizes the main architectural differentiators between AURA and traditional IoT approaches.

A. Quantitative Performance Analysis

The framework achieved an average end-to-end latency $L_{avg} = 850$ ms over $N = 30$ measurement windows. This is adequate for non-critical near real-time monitoring where sub-second responsiveness is beneficial but hard real-time guarantees are not required. Regarding update frequency, backend logs confirmed an inter-message jitter below ± 1 s for the 30-s cadence, confirming the system's stability in handling periodic telemetry.

TABLE I
TECHNICAL COMPARISON BETWEEN TRADITIONAL IOT ARCHITECTURES
AND THE PROPOSED AURA ARCHITECTURE

Parameter	Traditional IoT Architectures	AURA (Proposed Architecture)
Integration Capability	Limited to specific protocols	Multiprotocol support (MQTT, HTTP, LoRaWAN) with FIWARE modules
System Modularity	Monolithic or semi-modular	Fully modular and distributed
Monitoring Frequency	Occasional or low frequency	Configurable, real-time operation
Data Visualization	Basic or offline dashboards	Real-time web dashboards
Alert Management	Manual or nonexistent	Automatic and configurable alerts
Data Storage	Generally local or limited	Persistent hybrid storage (NoSQL/SQL)
Scalability	Limited by centralization	High, compatible with edge and cloud
Multidomain Support	Single use case focus	Multidomain (industrial, environmental, etc.)

Note: The comparative evaluation was performed using simulated deployment scenarios and architectural analysis, ensuring homogeneous metrics among the evaluated systems.

B. Scalability and Resource Utilization

Under a progressive load test, AURA sustained 47 concurrent devices without observable degradation, preserving packet loss below 1% and average latency below 1 s.

System monitoring on the AWS EC2 instance revealed a peak process memory usage of 49.6%. An additional 48.7% was identified as buffers/cache (effectively reclaimable), leaving more than 51% of RAM available for growth. CPU and network statistics remained stable, indicating that the deployment is not resource-saturated. Based on this measured headroom, the architecture demonstrates the capacity for significant horizontal scaling through container orchestration.

C. Network Consumption

The bandwidth demand per device was calculated based on the effective payload and protocol overhead. With an average message size of 222 bytes and a transmission rate of 2 messages/min, the consumption is defined as:

$$C = 222 \text{ bytes/msg} \times 2 \text{ msg/min} = 444 \text{ bytes/min} \approx 0.44 \text{ KB/min} \quad (4)$$

This low bandwidth demand supports massive deployments in constrained networks and complements lightweight messaging patterns commonly implemented with MQTT/HTTP bridges [21].

V. DISCUSSION AND COMPARATIVE ANALYSIS

The experimental results demonstrate the effectiveness of the proposed AURA architectural framework, particularly in addressing the long-standing gap between high-frequency telemetry acquisition and semantic context management. To

TABLE II
EXPERIMENTAL PERFORMANCE METRICS OF THE AURA DEPLOYMENT

Metric	Measured value	Interpretation
Average end-to-end latency	850 ms ($N = 30$)	Suitable for non-critical near real-time monitoring
Update interval	30 s (± 1 s jitter)	High refresh rate for IoT telemetry
Concurrent devices	47 nodes	Stable operation; scalable with horizontal scaling
Message size	222 bytes	LoRaWAN headers + payload
Network consumption	0.44 KB/min/node	Negligible bandwidth per device
RAM usage (peak)	49.6% (processes)	>51% effectively available (free+cache)

position these results within the current technological landscape, Table III presents a qualitative comparison between AURA and representative industry-standard IoT platforms.

TABLE III
QUALITATIVE COMPARISON OF AURA VERSUS STANDARD IOT SOLUTIONS

Feature	ThingsBoard / Proprietary	FIWARE (Vanilla)	AURA (Proposed)
Interoperability	Limited (internal data models)	High (NGSI-LD compliant)	Full cross-domain semantic mapping
Analytical persistence	Generic SQL/NoSQL	Requires external configuration	Optimized hybrid (context + time-series)
Critical network adaptability	Low (IP-dependent)	Medium	High (LoRaWAN payload optimization)
Deployment curve	Fast but rigid	Slow and complex	Modular and plug-and-play

A. Performance and Interoperability

As summarized in Table I, conventional IoT architectures frequently exhibit monolithic behavior, where increasing protocol heterogeneity translates into near-linear latency degradation. In contrast, the semantic mediation layer of AURA (see Section II-B) maintained a stable average end-to-end latency of approximately 850 ms, even under concurrent operation of 47 devices. This behavior indicates that the computational overhead introduced by FIWARE IoT Agents is marginal when compared to the interoperability capabilities they enable.

To support seamless cross-domain interoperability, AURA employs a semantic mapping mechanism that transforms heterogeneous raw telemetry into NGSI-LD (Next Generation Service Interface with Linked Data) entities. Unlike flat JSON-based representations, this approach adopts a formalized *Property-of-a-Property* modeling paradigm, allowing contextual metadata—including measurement units, sensor accuracy, and calibration timestamps—to be explicitly associated with each observation.

For instance, a LoRaWAN payload encoding temperature as a 12-bit integer is parsed by the IoT Agent and mapped to a URI-identified entity (e.g., `urn:ngsi-ld:Sensor:001`). This entity conforms to the Smart Data Models initiative, ensuring that the *temperature* attribute is semantically equivalent across FIWARE-enabled platforms. Such semantic alignment is critical for the construction of standardized Digital Twin representations, where data is treated not as an isolated numerical value, but as a contextualized asset linked to its spatial, temporal, and historical provenance.

B. Orchestration and Scalability

The stability observed under concurrent load conditions further validates the proposed hybrid persistence orchestration strategy, which combines MongoDB for contextual data management and CrateDB for time-series storage. This decoupled persistence approach effectively mitigates database locking and contention issues that commonly arise in single-database IoT architectures.

Experimental measurements report a peak RAM utilization of 49.6% and an average bandwidth consumption of 0.44 KB/min per node. These results indicate that the architecture can scale horizontally to support large-scale rural–urban mobility scenarios without incurring proportional increases in computational or network resource utilization.

C. Scientific Impact

Overall, the results support the central hypothesis of this work: that a decoupled, semantically orchestrated architecture—when supported by specialized agents and hybrid data persistence—effectively resolves the interoperability–performance trade-off identified in the state of the art. Consequently, AURA should be regarded not only as an operational IoT platform, but as a validated architectural reference model for next-generation autonomous infrastructure management systems.

VI. CONCLUSION

This paper proposed and validated the AURA architectural framework, an orchestrated IoT ecosystem designed to solve the interoperability and persistence bottlenecks in heterogeneous environments. By integrating FIWARE components with a specialized multi-protocol ingestion layer (MQTT, HTTP, and LoRaWAN) and a hybrid storage orchestration (MongoDB and CrateDB), the system successfully bridges the gap between raw sensing and semantic context management.

Experimental results on an AWS cloud deployment confirm the architectural efficiency of the proposal. The framework achieved an average end-to-end latency of 850 ms and sustained a stable 30-second update frequency, ensuring the responsiveness required for near real-time monitoring. Under progressive load, AURA maintained operational stability with 47 concurrent devices, preserving packet loss below 1% and average latency below 1 s. Furthermore, the negligible network consumption (0.44 KB/min per device) and optimized resource

utilization (49.6% RAM usage) demonstrate that the decoupled design provides significant headroom for horizontal scaling.

These results, together with established considerations on LoRaWAN capacity limits [22] and IoT messaging protocol performance [21], support the suitability of AURA for medium-to-large deployments across smart city, environmental monitoring, smart building, and mobility management use cases. Future work will focus on large-scale field deployments, long-term reliability evaluation, and the integration of predictive analytics (e.g., TinyML at the edge and MLOps pipelines) on top of the FIWARE ecosystem [5], leveraging the standardized NGSI-LD context model to enable autonomous, evidence-based decision-making.

REFERENCES

- [1] N. M. Obiri and H. Shikunzi, “Long-range wide area network (lorawan) connectivity and range evaluation in a rural setting,” *International Journal of Computer Applications*, vol. 185, no. 3, pp. 61–67, 2023. [Online]. Available: <https://www.ijcaonline.org/archives/volume185/number3/32689-2023922699/>
- [2] M. A. Ahmed, J. L. Gallardo, M. D. Zuniga, M. A. Pedraza, G. Carvajal, N. Jara, and R. Carvajal, “Lora based iot platform for remote monitoring of large-scale agriculture farms in chile,” *Sensors*, vol. 22, no. 8, p. 2824, 2022. [Online]. Available: <https://www.mdpi.com/1424-8220/22/8/2824>
- [3] W. A. Jabbar, T. M. Ting, M. F. I. Hamidun, A. H. C. Kamarudin, W. Wu, J. Sultan, A. A. Alsewari, and M. A. H. Ali, “Development of lorawan-based iot system for water quality monitoring in rural areas,” *Expert Systems with Applications*, vol. 242, p. 122862, 2024. [Online]. Available: <https://umpir.ump.edu.my/id/eprint/40086/1/Development%20of%20LoRaWAN-based%20IoT%20system%20for%20water%20quality%20monitoring%20in%20rural%20areas.pdf>
- [4] D. M. Berbes Villalón, L. Sánchez Jiménez, M. de la Iglesia Campos, M. E. Díaz Aguirre, and T. Delgado Fernández, “An iot architecture for smart cities based on the fiware platform,” *Revista de Ciencia y Tecnología*, vol. 38, no. 1, pp. 20–27, 2022. [Online]. Available: <https://www.redalyc.org/journal/3826/382679632003/>
- [5] J. Conde, A. Muñoz-Arcenales, A. Alonso, J. Salvachua, and G. Huecas, “Enhanced fiware-based architecture for cyberphysical systems with tiny machine learning and machine learning operations: A case study on urban mobility systems,” *IT Professional*, 2024. [Online]. Available: <https://oa.upm.es/84890/3/84890.pdf>
- [6] R. Mena, M. Ramos, L. Urquiza, and J. D. Vega-Sánchez, “Comprehensive evaluation of lorawan technology in urban and rural environments of quito,” in *Engineering Proceedings*, vol. 77, no. 1, 2024, p. 28. [Online]. Available: <https://www.mdpi.com/2673-4591/77/1/28>
- [7] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of things (iot): A vision, architectural elements, and future directions,” *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013.
- [8] M. Centenaro, L. Vangelista, A. Zanella, and M. Zorzi, “Long-range communications in unlicensed bands: the rising stars in the iot and smart city scenarios,” *IEEE Wireless Communications*, vol. 23, no. 5, pp. 60–67, 2016.
- [9] K. Lima, T. D. Oyetoyan, R. Heldal, and W. Hasselbring, “Evaluation of mqtt bridge architectures in a cross-organizational context,” *arXiv preprint arXiv:2501.14890*, 2025. [Online]. Available: <https://arxiv.org/abs/2501.14890>
- [10] A. R. Group, “Sistema aura: Monitoreo y automatización distribuida,” <https://aura-platform.org>, 2021, accedido: 7 agosto 2025.
- [11] R. Altuve and K. Medrano, “Aura: An iot platform for environmental monitoring in remote areas,” *Journal of Environmental Informatics*, vol. 35, no. 2, pp. 112–120, 2022.
- [12] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, “Internet of things for smart cities,” *IEEE Internet of Things journal*, vol. 1, no. 1, pp. 22–32, 2014.
- [13] J.-L. Wang, S. Tong, X.-Y. Li, Z. Yang, F. Xiao, and Y.-H. Liu, “On the scalability of internet of things systems,” *Journal of Computer Science and Technology*, vol. 40, pp. 1182–1194, 2025. [Online]. Available: <https://link.springer.com/article/10.1007/s11390-025-5178-5>

- [14] A. Analytics, "Plataforma de análisis de datos en tiempo real para iot," <https://aura-analytics.org>, 2023, accedido: 7 agosto 2025.
- [15] A. Edge, "Nodos inteligentes para el procesamiento local en sistemas distribuidos," <https://aura-edge.tech>, 2023, accedido: 7 agosto 2025.
- [16] J. Smith and M. Garcia, "Smart home automation: A literature review," *Automation in Construction*, vol. 124, p. 103386, 2021.
- [17] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context-aware computing for the internet of things: A survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 1–34, 2020.
- [18] L. Johnson and R. Martinez, "Integration of legacy systems in modern monitoring architectures," *Journal of Systems Architecture*, vol. 57, no. 6, pp. 532–543, 2011.
- [19] S. Ali, M. Khan, and M. Imran, "Iot-based smart water management systems: a review," *Environmental Monitoring and Assessment*, vol. 194, p. 314, 2022.
- [20] L. González, A. Gonzales, S. González, and A. Cartuche, "A low-cost iot architecture based on lpwan and mqtt for monitoring water resources in andean wetlands," *SN Computer Science*, vol. 5, no. 144, 2024. [Online]. Available: <https://link.springer.com/article/10.1007/s42979-023-02515-4>
- [21] S. Singh and R. Raj, "Performance comparison of mqtt and http protocols for iot," *International Journal of Computer Applications*, vol. 176, no. 3, pp. 1–5, 2020.
- [22] F. Adelantado, X. Vilajosana, P. Tuset-Peiró, B. Martinez, J. Melia-Seguí, and T. Watteyne, "Understanding the limits of lorawan," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 34–40, 2017.