Designing a FIWARE-based architecture for Electric Vehicle charging stations management platform

TSVETELIN STEFANOV[1], ELENA BLAGOEVA[2,3], DENIS CHIKURTEV[2,3*]
[1]Institute of Robotics, Bulgarian Academy of Sciences, SOFIA, BULGARIA
University of Library Studies and Information Technologies
Sofia,
BULGARIA

[2]Institute of Information and Communication Technologies
Bulgarian Academy of Sciences
Sofia,
BULGARIA

[3]Center of Competence for Mechatronics and Clean Technologies "Mechatronics, Innovation, Robotics, Automation and Clean Technologies" – MIRACle,
Sofia,
BULGARIA

*ORCID: 0000-0003-4903-7544

Abstract: - This paper proposes two architectural solutions for Electric Vehicle Management Systems (EVMS), based on open-access FIWARE components. The first architecture implements a simple configuration for charging station management, while the second builds upon the first by expanding its functionalities by adding historical data support. Modeling a charging station in the context of FIWARE requires a smart data model. . The paper introduces an improved smart data model, used for the management of a physical charging station. To validate the proposed architectures, the article first presents the results of a charging session performed with the charging station and management platform. A simulation of ten charging sessions for each architecture demonstrates its capabilities and enables a comparative assessment. The obtained results confirm that the proposed architecture provides better scalability and performance compared to the standard system, based on HTTP communication.

Key-words: - EVMS, IoT, FIREWARE, Communication, Charging Stations, Architectures, Smart data models

Received: April 26, 2025. Revised: August 24, 2025. Accepted: September 28, 2025. Published: December 31, 2025.

1 Introduction

As electric vehicles become increasingly woven into the fabric of daily life, one of the major hurdles that cities face is building out a robust charging infrastructure. In the context of smart urban development, charging stations—and the platforms that manage them—are no longer standalone systems. Instead, they've evolved into key components of a broader, interconnected urban network. These stations form the backbone of a dependable system that prioritizes accessibility and ease of use. At the same time, they're designed to interact seamlessly with other city services through the Internet of Things (IoT). Over the past ten years, we've seen a sharp rise in the number of charging stations, both within cities and along major transit routes. This growth has largely been fueled by government policies and European programs that champion green innovation and digital transformation. As a result, manufacturers have increased the production of various charging station models, creating an urgent need for Electric Vehicle Management Systems using smart, IoT-based platforms that can unify how these stations are monitored and managed across different urban settings. The software and communication protocols behind these platforms are essential—they allow for real-time data monitoring during charging sessions and support user account management. Integrated into the IoT ecosystem, these systems also define how data is collected and processed. When connected with other smart city technologies—like energy grids, traffic flow systems, and renewable power sources—they become powerful tools. The implementation of such platforms opens new opportunities for large-scale data analysis, predictive load forecasting, and dynamic load management within electrical grids, aligning with the principles of sustainable and intelligent urban development. The present study focuses on the development of a charging station management system, examining existing solutions of this type, their advantages and disadvantages, as well as the challenges associated with device homogenization and the standardization of the data they exchange. In this way, the study positions such systems not only as a key element in the evolution of charging infrastructure but also as essential components in the integration of charging infrastructure within smart city and IoT environments.

2 Related works

Recent research on electric vehicle charging technologies emphasizes the importance of combining advanced management strategies with interoperable architectures. The operator-centric architecture presented in [1], outlines a smart, scalable charging framework using real-time monitoring and predictive scheduling to optimize charging efficiency and operator profitability. Scalable IoT-based platform solutions, [2], and zero-trust security architectures, [3], provide frameworks that regulate the charging process while maintaining system security. A reference model for interoperable EV-charging reservation systems has also been introduced. It aligns stakeholders through standardized data models, interfaces, and communication protocols, and has been tested using interoperability scoring, agent-based simulations, and a proof-of-concept implementation, [4]. One proposed solution is a decentralized Cloud-SDN architecture that combines SDN and NFV, supports real-time coordination and dynamic pricing in EV and building energy management. This approach has shown better load balancing and greater microgrid stability during periods of high demand, [5].

Reference, [6], introduces a set of software design patterns aimed at building sustainable operational platforms, though it stops short of detailing real-world deployment scenarios. Reliability concerns are tackled in [7], while international efforts like the ACES project, [8], delve into service innovation and the potential for replicating solutions across different environments. While multi-layered system architectures for smart and connected vehicles are the focus of [9], similar frameworks for EV charging technologies - including configurations and system designs - are explored in [10], [11], [12]. These studies highlight various strategies for control and management, particularly within microgrids and larger charging infrastructures.

In [1], adaptive network models are shown to improve load balancing dynamically, and, [13], complements this with distributed control algorithms that enhance system responsiveness. On the communication front, the publish/subscribe model discussed in [14], offers more flexible reservation and resource allocation mechanisms. Meanwhile, [15], presents a model-based interoperability framework that helps manage complexity across systems - an essential step toward building scalable, standards-compliant EV charging networks ready for future demands.

Several studies examine interoperability and data management through NGSI-LD-based solutions. The study, [16], investigates how NGSI-LD data enrichment can reshape smart-city applications by enhancing semantic interoperability across heterogeneous IoT systems. Complementing this, [17], presents a connector that integrates NGSI-LD data into open data portals, enabling standardized publishing workflows and improving data accessibility for external services. The role of NGSI protocols in transportation is highlighted in [18], where NGSI-enabled mechanisms support IoT-enhanced mobility systems through interoperable data exchange and real-time context management. Finally, [19], provides an overview of FIWARE as a standard-based open-source IoT platform, detailing its modular architecture and emphasizing its suitability for scalable, interoperable smart-city and mobility solutions.

Building on these findings, the comparative analysis in Table 1 (Appendix) shows that among different deployment models, FIWARE-based components stand against basic standard EV charging platforms with their open-source components, modularity, and strong support for interoperability and context management.

While literature emphasizes the transformative potential of NGSI-LD-based solutions for achieving semantic interoperability, real-time context management, and standardized data exchange in smart-city and mobility applications, the comparative evaluation of existing EV management platforms Table 2 (Appendix) demonstrates that these capabilities are still only partially implemented across current solutions.

As Table 2 (Appendix) shows, most EV platforms rely primarily on API-based middleware-level interoperability, offering only *partial* or *non-semantic* data exchange, whereas communication is dominated by syntactic protocols such as OCPP and

OCPI. Event-driven architectures and advanced context-management mechanisms, central features of NGSI-LD and FIWARE, remain inconsistently adopted, and semantic interoperability is either proprietary or absent. This stands in contrast to FIWARE-based components, which, according to prior studies, provide native NGSI and NGSI-LD compliance, open semantic models, modular architecture, and context-aware processing through the Orion Context Broker (OCB). FIWARE's open-source context-centric approach offers a higher degree of interoperability, extensibility, and architectural transparency than standard EV charging platforms, which continue to operate within closed, partially interoperable, and API-constrained ecosystems. As a result, the table offers empirical support for the argument that FIWARE-based architectures are better positioned to enable scalable, interoperable, and semantically integrated EV charging infrastructures. This comparative analysis aligns with earlier findings that highlight NGSI-LD's importance for achieving interoperability and context-awareness, showing that FIWARE's modular and open architecture effectively addresses the limitations observed in standard EV platforms.

In Section 3, two FIWARE-based architectural solutions for charging station management are presented: a minimal architecture centered on the Orion-LD context broker and an improved one capable of supporting the growing number of charging stations, storing and visualizing their short-term historical data, and managing their contextual data flows.

3 Proposed architectures

Designing an effective FIWARE-based solution requires thoughtful selection and integration of components, depending on the scale and functional needs of the application. Since FIWARE components implement separate functionalities on their own, they are not mandatory to be included in every FIWARE architecture. According to the official documentation provided by the FIWARE team, the minimum architecture consists of only one mandatory element, the context broker, which is the heart of the system and is essential for the operation of all other FIWARE-based components. To implement charging station management through FIWARE-based architecture, two architectural solutions are proposed in this article. The first architecture represents a simple communication solution that uses only the Orion-LD context broker, while the second architecture is designed to be scalable, managing a larger number of charging stations and the contextual information they generate and send.

3.1 Simplified FIWARE-based architecture for charging station management

The proposed architecture illustrated in Figure 1 represents a minimal functional setup, designed to implement simple communication between FIWARE components and EV charging stations.

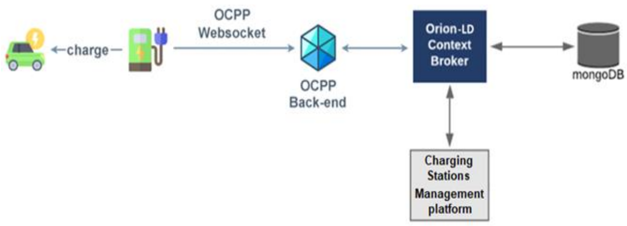

Fig. 1. Simplified FIWARE-based architecture
Source: created by the authors

The main component of the architecture is the Orion-LD Context Broker, whose role is to manage communication between the individual components of the system. Orion-LD uses the open-source, non-relational database MongoDB to store the current state of context data, along with information about subscriptions and registrations. As the context broker does not support bidirectional WebSocket communication required for interacting with charging stations via the OCPP protocol, it is necessary to implement a middleware service to perform this task. The architectural component, named OCPP Back-end, is responsible for communication between charging stations and the context broker. It implements some of the main functionalities of the OCPP 1.6 protocol: BootNotification, Heartbeat, Authorize, StartTransaction, StopTransaction, RemoteStartTransaction, RemoteStopTransaction, MeterValues, and StatusNotification. The OCPP back-end subscribes to the context broker to automatically receive the latest value of the command property of the charging station entity. The command property of a charging station entity can be start, stop, or status.

1) The start command is used to begin a charging session by using the "remote_start_transaction" method.

2) The stop command is used to end the ongoing charging session by using the "remote_stop_transaction" method.

3) The status command is used to retrieve the current charging status, transaction ID, and meter values from the charging station and returns them in JSON format.

When the command value of a given charging station entity is updated, the corresponding OCPP 1.6 method is triggered in the OCPP Back-end, which also updates the entity property dataValues used for storing the returned context information from the status command {"success": True,"data": { "status": config.status, "transaction_id": config.transaction_id, "meter_value": config.meter_value}}. In this architecture, charging stations are managed through a web application that sends HTTP PATCH requests to the charging station entities registered in the context broker, updating their command values via a graphical user interface (Figure 2).

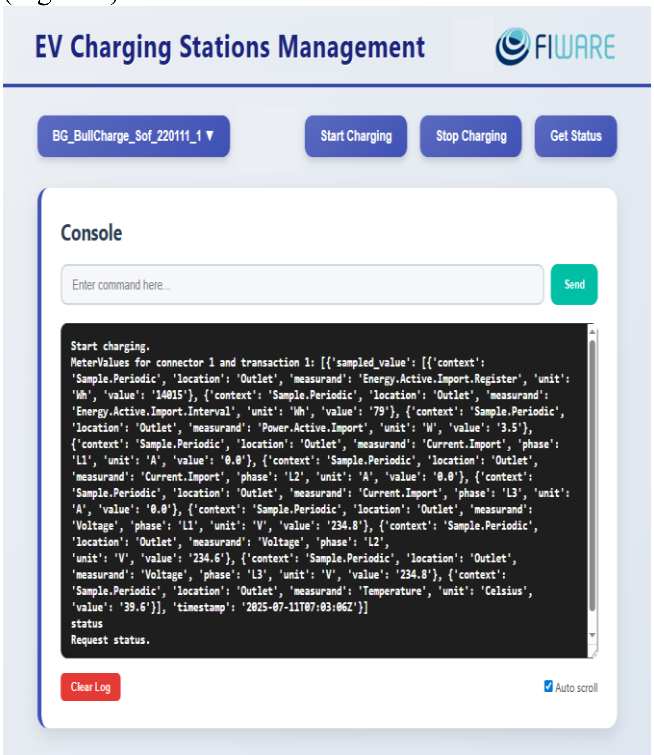

Fig. 2. User interface of the charging stations management platform Source: created by the authors

3.2 An improved FIWARE-based architecture for charging station management

The previous architecture introduced a simplified method for communicating with each registered charging station by updating their entity values in Orion-LD via HTTP requests. Building upon this architecture, the new system enhances the Orion-LD Context Broker by integrating additional components, while still maintaining it as the core of the architecture (Figure 3).

As the number of charging stations increases, managing the growing volume of context data they generate becomes increasingly challenging. For this reason, FIWARE introduced an IoT Agent JSON, which is a component that lets a group of devices send their data to and be managed from a Context Broker using their own native protocols. Both the Orion-LD Context Broker and the IoT Agent JSON rely on MongoDB to store the latest context information. The IoT Agent JSON communicates in this scenario with a Mosquitto MQTT Broker using the publish-subscribe pattern, acting as a subscriber to specific MQTT topics where charging stations publish data, allowing the agent to receive messages and translate them into context updates for Orion-LD.

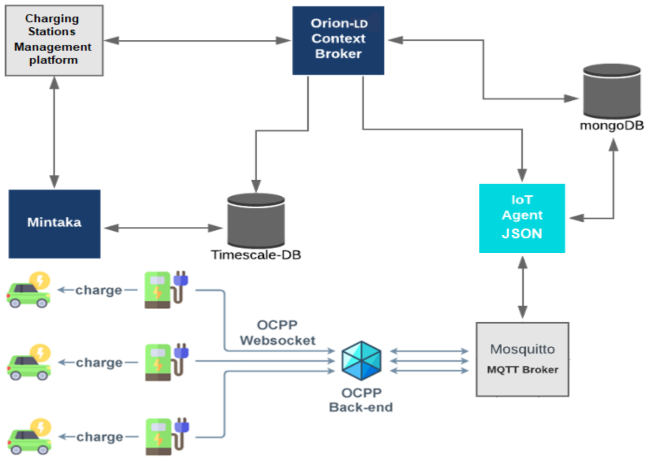

Fig. 3. Improved FIWARE-based architecture Source: created by the authors

In this architecture, the OCPP Back-end once again serves as a middleware mechanism, but this time it acts between the charging stations and the Mosquitto MQTT broker. When a command such as start, stop, or status is received on the MQTT topic „/json/apiKey/EVChargingStation:001/attrs/cmd/“, the OCPP Back-end forwards the received command from the topic via WebSocket to trigger the corresponding control method in OCPP. When a charging station sends its payload object as a result of the status command on the topic “/json/apiKey/EVChargingStation:001/attrs/dataValues/”, the dataValues property of the charging station’s entity is updated with the received object. The charging stations are once again managed through the Charging Stations Management Platform service, which now includes functionality for querying historical data, enabled by the Mintaka service, not just the latest value of a specific entity, as in the previous architecture. Management commands for charging stations are sent from the management platform as PATCH requests addressed to the respective charging station entity registered in Orion-LD.

In the context of FIWARE-based platforms, there are two directions of data transfer: northbound and southbound. Northbound typically refers to information sent directly to the context broker (in this case, the patch requests), while southbound refers to information sent to the context broker via the MQTT broker and IoT Agent JSON. In the proposed architecture, management commands are sent via the northbound path, while feedback from the charging stations is received by the context broker via the southbound path. The Mintaka add-on services the temporal interface and is also responsible for persisting the context. One advantage of the temporal interface is that it is provided directly by the context broker, eliminating the need for subscriptions and reducing HTTP traffic. Furthermore, the temporal interface can be queried across all context entities, not just those that satisfy a subscription. A Timescale time-series database, built on PostgreSQL, is used for persisting historical context data, which can also be queried directly via SQL queries to the database. The charging stations management platform uses the Mintaka API to query the short-term historical data of a specific entity. When the corresponding charging station registered in Orion-LD is selected from a drop-down list, its short-term historical data is displayed directly in the interface terminal and dynamically loaded on scroll using the mouse.

4 Designing a smart data model for Electric Vehicle Charging Stations

Charging stations used in modern electromobility represent a highly diverse class of devices. The market offers numerous models that differ in design, power capacity, communication protocols, connector types, functionalities, and intended use. This diversity is a natural outcome of the rapid development of the industry, the coexistence of different national standards, and the continuous efforts of manufacturers to introduce innovative solutions. At the same time, this heterogeneity creates significant challenges in integrating charging stations into unified networks and platforms. Because of the wide spectrum of technical parameters and features, the need for a more standardized, abstract model capable of representing most existing charging stations comparably and coherently. Such a model should rely on a set of fundamental characteristics shared across all types of charging stations, regardless of manufacturer or specific implementation. For this purpose, an analysis of the technical characteristics of different types of charging stations was performed (see Table 3 (Appendix)). The insights derived from this analysis form the basis for defining a unified representation that can model the essential attributes of a heterogeneous charging infrastructure. After presenting the characteristics of the charging stations under consideration, the common characteristics between them are identified, which can be used to build a unified smart data model that can be applied to a larger number of charging stations. The FIWARE Foundation also offers smart data models for charging stations, which are developed primarily by an active developer community. The main problems with these open smart data models, available in GitHub repositories, are often too general and miss many important technical details about the physical charging station, making it difficult to model their behavior. To address these issues, this paper presents an improved and more practical smart data model for a charging station that implements the main features extracted from the analysis of each charging station model from the table above (see Table 4 (Appendix)). The table below provides a real-life example with data from a charging station that will be used to validate the proposed architectures in the next section of the paper. The proposed smart data model for a charging station can serve as a practical example that can also be used for other types of charging stations.

5 Validation of the Proposed Platforms through a Charging Session and System Load Analysis

For this experiment, we used the Chargestorm Connected 2 charging station, an AC electric vehicle charger equipped with a single Type 2 outlet. This station supports adjustable charging power in the range of 1.4kW to 11kW across three phases, with a current range of 6–16A. It incorporates built-in AC and DC ground fault detection (30mA AC, 6 mA DC), a MID-certified energy meter, and offers load balancing functionality via the NanoGrid system. The charger can connect to networks by using Ethernet or optional 3G, 4G modules, and supports the OCPP 1.6 communication protocol, which allows integration into remote monitoring and control systems. The charger is suitable for various environments, with an operating temperature range of -30°C to +50°C.

5.1 Results from the charging session

The experimental charging session, conducted on 06 April 2025, from 15:07:23 to 16:07:17, lasted approximately 59 minutes and 54 seconds. During this period, the station supplied an estimated 3.46kWh of energy, operating in single-phase mode with a nominal maximum capacity of 3.7kW. The power curve, illustrated in Figure 4 (Appendix), reveals a rapid stabilization phase, after which the system maintained a nearly constant charging power of 3.4kW-3.6kW. Periodic, short-lived interruptions, evident as sharp drops to zero, occurred throughout the session. These events were synchronized across the monitored power, current, and voltage, which suggests that they represent brief negotiation or regulation cycles rather than a malfunction. The current profile closely mirrors the power trend, peaking at 15.8A and maintaining an average of 15.25A. The current interruptions match the timing of the power drops, confirming their systemic origin within the charging control loop. The voltage dynamics in Figure 5 (Appendix) indicate moderate variability, with values ranging between 223.4V and 227.9V and averaging 225.16V. Such fluctuations are consistent with normal grid behavior under load, and the brief dips are temporally correlated with the adaptive interruptions observed in the power profile.

5.2 Simulation of charging sessions for system load evaluation

After conducting a charging session with a real charging station, the information provided during the session is stored in order to simulate a similar load on the proposed systems. The testing methodology remains identical for both platforms, but the way in which each platform receives data differs. For the first platform, which is based solely on the Orion-LD context broker, each update to the charging station entity is performed via an HTTP PATCH request. Accordingly, for every row in the dataset, one PATCH request is issued to the context broker. The original dataset contains 936 rows. Since the proposed scenarios include 10 simulated charging stations, this results in a total of 936 × 10 = 9,360 HTTP PATCH requests sent to the first platform during the simulated charging sessions. Each session lasts 1 hour and 34 seconds, meaning that Orion-LD receives these updates continuously throughout this period. The purpose of this experiment is to determine the system's reliability and responsiveness under a high-volume update load. For the second platform, which extends Orion-LD by incorporating an IoT Agent and an MQTT Broker, the same dataset is used, however, the charging stations transmit their updates via MQTT messages instead of HTTP requests. Because MQTT is a lightweight publish–subscribe protocol designed for high-frequency IoT communication, it is expected to reduce communication overhead and improve system responsiveness under load. During the experimental load testing of both proposed charging station management systems, 10 simultaneously active charging sessions were simulated. When comparing the Orion-LD–only platform to the extended Orion-LD + IoT Agent + MQTT platform, the graphical and numerical behavior of the three monitored metrics, latency, memory usage, and CPU load, clearly shows the advantages of the second architectural model. As shown in Figure 6 (Appendix), the single-layer Orion-LD platform maintains an average latency of approximately 120–150 ms, but exhibits recurring spikes reaching 500–650 ms, indicating significant variability and sensitivity to the large number of HTTP PATCH requests. In contrast, the extended platform using MQTT maintains latencies concentrated around 70–90 ms, with rare deviations and maximum peaks that seldom exceed 150–180 ms, representing more than 2.5× lower peak latency compared to the baseline architecture. A similar pattern is observed in the use of RAM, as can be seen on (Figure 7 (Appendix)). On the first Orion-LD-based platform, the average load consistently hovers around 75–85%, with frequent local excursions above 90%, while the improved platform operates stably in the 55–65% range, without significant and prolonged peaks, which indicates more efficient management of states and buffers through MQTT mediation. Processor-load measurements (Figure 8 (Appendix)) offer the clearest evidence of the difference. While the first Orion-LD-based platform operates at a consistently elevated CPU load of 80–95%, with spikes nearing full saturation, the architecture employing Orion-LD together with an IoT Agent and MQTT Broker remains within the markedly lower interval of 45–60%, despite an equivalent volume of simulated charging operations. In summary, the numerical parameters from the examined three figures clearly demonstrate that the second architecture not only reduces peak loads by over 40–60% in all metrics, but also provides a significantly more stable and predictable operating profile under identical experimental conditions.

6 Future works

The test results confirmed that the charging session was executed successfully, and the proposed platform did not encounter any difficulties during the production phase. The proposed architectures and smart data model suggest several directions for future research. In future work, tests will be conducted with other types of charging stations, because the current model was built from a limited set of AC charging stations and tested with a single device, so it should be checked against a wider variety of AC and DC fast chargers, including multi-point and high-power public hubs, to see how well it generalizes and what extra attributes may be needed. The OCPP backend will also be enhanced by adding functionalities available in the OCPP 2.0.1 protocol, since the charging station used is compatible only with OCPP 1.6. Also, the charging station management platform was deployed locally on a low-capacity server. Because only a single charging station was available, ten stations were simulated, all using a single dataset acquired from the real charging session. In future work, additional test sessions will be carried out to generate more diverse charging profiles, thereby enabling more comprehensive testing of the platform. Since both platforms currently rely exclusively on FIWARE-based technologies, future work will include benchmarking them against comparable solutions built on alternative architectures.

7 Conclusion

The proposed solutions demonstrate that FIWARE provides a robust and open platform for managing EV charging stations, combining flexibility, compliance with standards, and scaling capabilities. The two architectures show a practical path: a minimal installation with Orion-LD and OCPP back-end for basic management, and an enhanced architecture with IoT Agent JSON and MQTT broker. The proposed smart data model captures detailed technical characteristics of real charging stations. The model was derived from a comparative analysis of multiple commercial chargers and instantiated for a specific AC station, demonstrating how heterogeneous specifications can be represented in a unified NGSI-LD structure. The conducted experimental charging session is confirmed as successful and demonstrates stable behavior of the station and the exchange of contextual data through the platform. The two architectural variants with a basic Orion-LD configuration and an extended solution including IoT Agent and MQTT broker were compared and evaluated in terms of their performance through simulated charging sessions. The results showed that the extended architecture achieves lower latency and more efficient resource usage while preserving scalability.

Acknowledgment

This work has been supported by contract number BG-RRP-2.017-0031-C01, "Research and development of a smart Energy system for eco-charging of electric vehicles, using renewable energy sources" – RENEW and contract number BG-RRP-2.005-0003, “Strengthening the scientific excellence and innovation capacity in the area of smart city through twinning” funded by European Union-NextGenerationEU through the National Recovery and Resilience Plan of the Republic of Bulgaria.

This work was accomplished by the Center of Competence for Mechatronics and Clean Technologies "Mechatronics, Innovation, Robotics, Automation and Clean Technologies" – MIRACle, with the financial support of contract No. BG16RFPR002-1.014-0019-C01, funded by the European Regional Development Fund (ERDF) through the Programme "Research, Innovation and Digitalisation for Smart Transformation" (PRIDST) 2021–2027.

References:

[1] Lee, Z. J., Lee, G., Lee, T., Jin, C., Lee, R., Low, Z., Chang, D., Ortega, C., & Low, S. H. (2021). Adaptive charging networks: A framework for smart electric vehicle charging. IEEE Transactions on Smart Grid, 12(5), 4339-4350. https://doi.org/10.1109/tsg.2021.3074437

[2] Beránek, M., Feuerlicht, G., Kucera, O., & Kovár, V. (2023). An Architecture for a Large-Scale IoT e-Mobility Solution. In ICEIS (1) (pp. 733-740). https://doi.org/10.5220/0011827100003467

[3] Carroll, T. E., Chang, L. H., & Wright-Hamor, C. L. (2024). The Design and Evaluation of Zero Trust Architecture for Electric Vehicle Charging Infrastructure: EVs@ Scale Series on EV Charging Station Cybersecurity (No. PNNL-36824). Pacific Northwest National Laboratory (PNNL), Richland, WA (United States). https://doi.org/10.2172/2462754

[4] Basmadjian, R., Kirpes, B., Mrkos, J., & Cuchý, M. (2020). A reference architecture for interoperable reservation systems in electric vehicle charging. Smart Cities, 3(4), 1405-1427. https://doi.org/10.3390/smartcities3040067

[5] Chekired, D. A., Khoukhi, L., & Mouftah, H. T. (2017). Decentralized cloud-SDN architecture in smart grid: A dynamic pricing model. IEEE Transactions on Industrial Informatics, 14(3), 1220-1231. https://doi.org/10.1109/tii.2017.2742147

[6] Meheden, M., Musat, A., Traciu, A., Viziteu, A., Onu, A., Filote, C., & Răboacă, M. S. (2021). Design Patterns and Electric Vehicle Charging Software. Applied Sciences, 11(1), 140. https://doi.org/10.3390/app11010140

[7] Keith, D., & Womack, J. (2023). Building and sustaining reliable public EV charging in the United States. Environmental Research Letters, 18(1), 011004. https://doi.org/10.1088/1748-9326/acae39

[8] Marinelli, M., Thingvad, A., & Calearo, L. (2020). Across Continents Electric Vehicles Services Project: Final Report. Technical

University of Denmark. https://orbit.dtu.dk/en/publications/across-continents-electric-vehicles-services-project-final-report/ [Access date: 23/01/2026]

[9] Yu, D., & Xiao, A. (2022). The Digital Foundation Platform - A Multi-layered SOA Architecture for Intelligent Connected Vehicle Operating System. SAE Technical Paper 2022-01-0107. https://doi.org/10.4271/2022-01-0107

[10] Acharige, S. S., Haque, M. E., Arif, M. T., & Hosseinzadeh, N. (2022). Review of electric vehicle charging technologies, configurations, and architectures. arXiv preprint arXiv:2209.15242. https://doi.org/10.48550/arXiv.2209.15242

[11] Ayoade, I. A., & Longe, O. M. (2024). A comprehensive review on smart electromobility charging infrastructure. World electric vehicle journal, 15(7), 286. https://doi.org/10.3390/wevj15070286

[12] Savio Abraham, D., Verma, R., Kanagaraj, L., Giri Thulasi Raman, S. R., Rajamanickam, N., Chokkalingam, B., Sekar, K. M., & Mihet-Popa, L. (2021). Electric vehicles charging stations' architectures, criteria, power converters, and control strategies in microgrids. Electronics, 10(16), 1895. https://doi.org/10.3390/electronics10161895

[13] Nimalsiri, N. I., Mediwaththe, C. P., Ratnam, E. L., Shaw, M., Smith, D. B., & Halgamuge, S. K. (2019). A survey of algorithms for distributed charging control of electric vehicles in smart grid. IEEE Transactions on Intelligent Transportation Systems, 21(11), 4497-4515. https://doi.org/10.1109/tits.2019.2943620

[14] Cao, Y., Wang, N., Kamel, G., & Kim, Y. J. (2015). An electric vehicle charging management scheme based on publish/subscribe communication framework. IEEE Systems Journal, 11(3), 1822-1835. https://doi.org/10.1109/jsyst.2015.2449893

[15] Kirpes, B., Danner, P., Basmadjian, R., Meer, H. D., & Becker, C. (2019). E-mobility systems architecture: a model-based framework for managing complexity and interoperability. Energy Informatics, 2(1), 15. https://doi.org/10.1186/s42162-019-0072-4

[16] González, V., Martín, L., Santana, J. R., Sotres, P., Lanza, J., & Sánchez, L. (2024). Reshaping smart cities through ngsi-ld enrichment. *Sensors*, *24*(6), 1858. https://doi.org/10.3390/s24061858

[17] Martín, L., Lanza, J., González, V., Santana, J. R., Sotres, P., & Sánchez, L. (2024). A Connector for Integrating NGSI-LD Data into Open Data Portals. *Sensors*, *24*(5), 1695. https://doi.org/10.3390/s24051695

[18] Bouloukakis, G., Zeginis, C., Papadakis, N., Zervakis, P., Plexousakis, D., & Magoutis, K. (2022, November). Enabling IoT-enhanced transportation systems using the NGSI protocol. In *Proceedings of the 12th International Conference on the Internet of Things* (pp. 33-40). https://doi.org/10.1145/3567445.3567460

[19] Cirillo, F., Solmaz, G., Berz, E. L., Bauer, M., Cheng, B., & Kovacs, E. (2020). A standard-based open source IoT platform: FIWARE. *IEEE Internet of Things Magazine*, *2*(3), 12-18. https://doi.org/10.1109/iotm.0001.1800022

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The authors equally contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

Report potential sources of funding if there is any

Conflict of Interest

Please declare anything related to this study.

The authors have no conflicts of interest to declare that are relevant to the content of this article.

Appendix

Table 1. Comparison of EV charging system components Source: created by the authors

Criteria / Elements	**Comparison of EV charging system components**	
	FIWARE-based components	***Standard platform's components***
Compliance with standards	NGSI, OCPP, OCPI, NGSI-LD	Usually only OCPP, OCPI (partially), proprietary API
Openness / Interoperability	Fully open	Closed code, limited API access
Scalability	Containerized, cloud-ready	Cloud scalability (SaaS)
Context Awareness (Context Mgmt)	Yes - Orion Context Broker	No (rarely context-entity-based model)
Analytics / Big Data support	QuantumLeap, Cygnus, Grafana	Limited - closed BI (Business Intelligence) modules
Security and IAM (OAuth, roles)	Keyrock, AuthZForce	Built-in (usually centralized IAM system)
Integration with external systems	REST, NGSI, MQTT, Kafka, eMobility API	Limited or only through a paid API
Sustainability and reliability	HA, Kubernetes-ready	Supported by the cloud provider
Visualization	WireCloud, Grafana	Built-in panels (usually with personalization limits)
Licensing/price	Open Source	Paid access, monthly or annual subscription
Local control / offline mode	Supported via Edge IoT Agents	No - depends on a constant internet connection
AI/ML and predictive charging support	Yes - via built-in AI services	Partially - proprietary algorithms, not transparent

Table 2. Comparison of different EVMS according to general characteristics. Source: created by the authors

EVMS	Semantic Interop.	Interop. Layer	Comm. Protocols	Event-driven Architecture	Integration Flexibility	Vendor Lock-in Risk	Security Capabilities	Extensibility via Plugins	Microservice Scalability
AMPECO	Partial	API-based	OCPP, OCPI, OICP (V2G/VPP via API)	Partial events	Cloud, Hybrid	Medium	Standard security stack	High API, market place	High
Driivz	Partial	Limited middleware	OCPP, OCPI	Partial	Cloud-centric	Medium	Enterprise-grade security	Low extensibility	High but closed
GreenFlux	Proprietary	Limited	OCPP, OCPI, ISO 15118	Limited	Cloud	Medium	Standard security controls	Low	Medium
EV Connect	Partial	Good API layer	OCPP, OCPI, ISO 15118-ready	Partial	Cloud	Medium	Enhanced security stack	Medium API layers	Medium
Charge Lab	Non-semantic	API middleware	OCPP, OCPI, REST	Partial	Edge, Cloud	Medium	OAuth2-based framework	Medium extensibility	Medium
Ampcontrol	Partial	AI middleware	OCPP, API	Event-driven smart charging	Cloud	Medium	Robust API-level security	Medium	Medium
Monta	Partial	Roaming/API layer	OCPP, OCPI	Partial	Cloud	Medium	OAuth2-based framework	Medium	Medium
Virta	Partial	Roaming layer	OCPP, OCPI	Partial	Cloud-first	Medium	Certified security management	Medium	High
Wallbox	Proprietary	Limited	OCPP, REST	Low	Cloud	High	Hardware-assisted security	Low	Medium
Shell Recharge Solutions	Partial	Enterprise middleware	OCPP, OCPI	Partial	Cloud	Medium	Enterprise-grade security	Low	High
FIWARE-based Stack	Full semantic with NGSI-LD	Full interoperability with Orion-LD	OCPP, OCPI, MQTT, OPC-UA, Modbus, LwM2M, REST	Full event-driven architecture	Cloud, Edge, Hybrid	Low Open Source	Modular open-standard security (Keyrock, Wilma, AuthZForce, OAuth2)	High FIWARE Catalogue	Very high

Table 3. Summary of core technical characteristics of EV charging station models. Source: created by the authors

Technical Specification	CS-1	CS-2	CS-3	CS-4	CS-5	CS-6	CS-7	CS-8	CS-9	CS-10	CS-11
Manufacturer	Victron Energy	Alfen	Alfen	CTEK	DEFA	EVNet	Huawei	Huawei	Schneider Electric	SUNNIC	ABB
Model / family	EV Charging Station	Eve Single S-line	Eve Single Pro-line	Chargestorm Connected 2	Solid EV CS	COMPACT 3 3SLC & 3TLC	Smart Charger SCharger-7KS-S0	Smart Charger SCharger-22KT-S0	EVlink Home	SKBAC Series AC EV CS B/C	Terra AC
Max AC charging power (kW)	22 kW	7.4 kW (1-ph) 22 kW (3-ph)	7.4 kW (1-ph) 22 kW (3-ph)	Up to 22 kW (3-phase, 32A)	≈4.6 kW (230 V, 20A, 1-phase)	7.4 kW max (1-ph) / 22 kW max (3-ph)	7.4 kW	22 kW (Mode A) / 11 kW	3.7 kW / 7.4 kW (1-ph) / 11 kW (3-ph)	3.5 kW 7 kW (1-ph)	3.7 / 7.4 / 11 / 22 kW
Number of phases	3-phase	1-phase 3-phase	1-phase 3-phase	1-phase 3-phase	1-phase	1-phase 3-phase	1-phase	3-phase	1-phase 3-phase	1-phase 3-phase	1-phase 3-phase
Input voltage range	170–265VAC 50 Hz	230 V 1-ph / 400 V 3-ph, 50 Hz	230 / 400 V AC, 50 Hz	230/400 V AC, 50 Hz (TN-S)	230 VAC, 1-phase, 50–60 Hz	230 VAC 50 Hz (1-ph)	Single-phase 230 V AC (±20%)	Three-phase 400/230 V AC (±20%)	230 V ±10% (1-ph)	230 V 50/60 Hz (1-ph)	230 V (1-ph)
Rated current (max per phase, A)	32 A / phase	32 A per phase	32 A per phase	Up to 64 A 32 A per outlet	6–20 A	1×32 A 3×32 A	32 A	32 A (Mode A) 16 A (Mode B)	16–32 A	16 or 32 A	Up to 32 A
Output / charging current range (A)	6–32 A	Up to 32 A	Up to 32 A	Up to 32 A per outlet	6–20 A	6–32 A	Up to 32 A	Up to 32 A	Adjustable 6–32 A	Up to 32 A	Up to 32 A
Charging mode (IEC 61851)	Mode 3	Mode 3	Mode 3	Mode 3	Mode 3	Mode 3	Mode 3	Mode 3	Mode 3	Mode 3	Mode 3
Connector type	IEC 62196 Type 2	Type 2 socket or fixed cable	Type 2 socket or fixed cable	1 or 2 Type 2 outlets fixed cable	Type 2	IEC 62196 Type II (socket/cable)	Type 2	Type 2	Type 2 socket with or without fixed cable	IEC 62196-2 Type 2 socket / plug with cable	Cable Type 1 or Type 2

Supported earthing grid types	Not specified	TN system	TN system	TN-S and IT networks	TN and IT power grid	TT, TN, TN-C-S	Single-phase, three-wire	Three-phase five-wire	TT, TN-S, TN-C-S	Not specified	TN/TT systems (typical
Built-in RCD type	External RCD required	External RCD required	External RCD per EVZ/E	30 mA AC 6 mA DC	30 mA AC & 6 mA DC	RCD A, 6 mA DC	RCP internal	RCP internal	30 mA AC (A-SI) with 6 mA DC filter	RCDM Type A 30 mA + DC 6 mA built-in	Not specified
DC leakage detection	Not specified	Max. 6 mA DC detection	Max. 6 mA DC detection	Yes, 6 mA DC	Yes, 6 mA DC	Yes, 6 mA DC	Yes, according to IEC 62955	Yes, per IEC 62955	Yes, 6 mA DC filter	Yes, 6 mA DC	Yes, 6 mA DC sensor
Overcurrent protection	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Overtemperature protection	Not specified	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Ingress protection (IP rating)	IP44	IP55	IP55	IP54	IP44	IP55	IP54	IP54	IP54 - cable IP55 - socket	IP65	IP54
Mechanical impact protection (IK rating)	Not specified	IK10	IK10	IK10	IK08	Not specified	Not specified	Not specified	IK10	Not specified	IK10
Operating temperature °C	-25 to +50	-25 to +40	-25 to +40	-30 to +50	-40 to +50	Not specified	-35 to +45	-35 to +40/50	-30 to +50	-30 to +55	-35 to +50
Storage temperature °C	-40 to +80	Not specified	Not specified	-30 to +50	-50 to +85	Not specified	-40 to +70	-40 to +70	-40 to +80	-40 to +75	Not specified
Max relative humidity	95%, non-condensing	5–95%	5–95%	Up to 100% at +25 °C	Not specified	standard outdoor	5–95% RH, non-condensing	5–95% RH, non-condensing	Up to 95% at 40 °C	<=95% RH, no condensation	5–95% RH
Communication - WiFi	802.11 2.4 GHz	No (Ethernet primary)	No (Ethernet primary)	No (Ethernet/4G)	Not specified	2.4 GHz WLAN	WiFi 2.4 GHz	WiFi 2.4 GHz	No (home version)	smartphone app (Ethernet/4G)	Optional WiFi
Communication - Bluetooth	Yes	No	No	No	No	Yes	Yes	Yes	No	Not specified	No
Communication - Ethernet / LAN	No, Modbus TCP over WiFi	Yes, RJ45 Ethernet / LAN	Yes, RJ45 Ethernet / LAN	Yes, RJ45 Ethernet	No	RS-485 Modbus RTU	FE 100 Mbit/s Ethernet	FE 100 Mbit/s Ethernet	No	Yes, RJ45 Ethernet	Ethernet
Communication - Cellular	No	Optional CPR	Optional CPR	4G (LTE Cat-1)	No	GSM 2G/4G	Not specified	No	No	Not specified	Optional 4G

(GSM/3G/4G)		S 3-phase	S 3-phase	3G/2G fallback							
OCPP support	Not specified	OCPP 1.5 / 1.6	OCPP 1.5 / 1.6	OCPP 1.5 / 1.6	No	OCPP 1.6	OCPP 1.6	OCPP 1.6	No	OCPP 1.6J	OCPP 1.6
RFID support	No	Yes	Yes	Yes	No	Yes	Yes	Yes	No	Yes	Yes
Energy meter	Not specified	MID certified	MID certified	Internal or MID energy meter	Not specified	Internal	Internal	Not specified	No MID meter basic	Internal	MID certified
UI / display	Yes	Yes	Yes	Yes	Not specified	Yes	Yes	Yes	Yes	Yes	Yes
Dimensions (mm)	377 × 307 × 148	370 × 240 × 130	370 × 240 × 130	449 × 282 × 160	312 × 85 × 70	200 × 129 × 350 200 × 108 × 350	335 × 180 × 145	335 × 180 × 145	282 × 409 × 148	410 × 260 × 140	Not specified
Approx. weight	3 kg	≈4 kg	≈4 kg	Up to 24 kg	2.3 kg	≈3.1–3.6 kg	≈3.0 kg	≈3.1 kg	≈3.2–5.6 kg	≤10–12 kg	Not specified
Main standards / certifications	IEC 61851-1 IEC 61851-22	IEC 61851-1	IEC 61851-1	IEC 61851-1	CE	CE	EN 61851-1:2019	EN 61851-1:2019	IEC 61851-1	IEC 61851-1	IEC/ EN 61851

Table 4. Electric Vehicle Smart Data Model (CHARGESTORM CONNECTED 2 example) Source: created by the authors

Attribute	**AttributeType**	**Value**
id	Core	urn:ngsi-ld:EVChargingStation:001
type	Core	EVChargingStation
modelName	Property	Chargestorm Connected 2, 910-17050
stationName	Property	Parking spot 001 – Chargestorm Connected 2
manufacturer	Property	CTEK
modelFamily	Property	Chargestorm Connected
location	GeoProperty	{"type": "Point", "coordinates": [42.676191, 23.368973]}
address	Property	{"streetAddress": "", "addressLocality": "", "postalCode": "", "addressCountry": "BG"}
description	Property	AC EV charging station (CTEK Chargestorm Connected 2) with Type 2 outlet, OCPP connectivity, and MID energy metering.
status	Property	available
capacity	Property	1
socketNumber	Property	1
allowedVehicleType	Property	["car"]
socketType	Property	["Type2"]
chargingPower	Property	{"min": 1.4, "max": 11, "unit": "kW"}

maxACChargingPower	Property	{"max": 22, "phases": ["threePhase"], "maxCurrentPerPhase": 32, "unit": "kW"}
numberOfPhases	Property	["singlePhase", "threePhase"]
inputVoltageRange	Property	{"min": 230, "max": 400, "frequency": "50 Hz", "system": "TN-S", "unit": "V"}
amperage	Property	{"totalMax": 64, "perOutletMax": 32, "unit": "A"}
outputCurrentRange	Property	{"min": 6, "max": 32, "unit": "A"}
chargingMode	Property	{"iec61851Mode": "Mode 3", "currentType": "AC"}
connectorType	Property	Type 2
numberOfOutlets	Property	1
supportedEarthingGridTypes	Property	["TN-S", "IT"]
builtInRCD	Property	{"type": "RCD", "ac": 30, "dc": 6, "unit": "mA"}
groundFaultDetection	Property	{"AC": 30, "DC": 6, "unit": "mA"}
dcLeakageDetection	Property	{"present": true, "threshold": 6, "unit": "mA"}
overcurrentProtection	Property	TRUE
overtemperatureProtection	Property	TRUE
protection	Property	{"ingress": "IP54", "impact": "IK10"}
operatingTemperature	Property	{"min": -30, "max": 50, "unit": "°C"}
storageTemperature	Property	{"min": -30, "max": 50, "unit": "°C"}
maxRelativeHumidity	Property	{"max": 100, "referenceTemperature": 25, "unit": "%"}
communicationProtocol	Property	OCPP 1.5 / 1.6
connectivity	Property	["Ethernet", "4G", "3G", "2G"]
communicationWifi	Property	FALSE
communicationBluetooth	Property	FALSE
communicationEthernet	Property	TRUE
communicationCellular	Property	["4G", "3G", "2G"]
ocppSupport	Property	["OCPP 1.5", "OCPP 1.6"]
rfidSupport	Property	TRUE
energyMeter	Property	MID
energyMeterDetail	Property	{"type": ["internal", "MID"], "midCertified": true}
uiDisplay	Property	TRUE
dimensions	Property	{"width": 282, "height": 449, "depth": 160, "unit": "mm"}
weight	Property	{"max": 24, "unit": "kg"}
standardsCertifications	Property	["IEC 61851-1"]
chargeType	Property	["other"]
acceptedPaymentMethod	Property	[]
openingHours	Property	24/7
network	Property	
operator	Property	
vehicleType	Property	car
powerConsumption	Property	
command	Property	["start", "stop", "status"]

dataValues	Property	{}
dataDescriptor	Relationship	urn:ngsi-ld:Person:001
relatedParkingSpot	Relationship	urn:ngsi-ld:ParkingSpot:001
dateCreated	Property	2025-12-10T00:00:00Z
dateModified	Property	2025-12-10T00:00:00Z
dataProvider	Property	YourOrganizationName
source	Property	EV charging station specifications and Smart Data Models EVChargingStation schema
owner	Property	[]
seeAlso	Property	[]
@context	Context	["http://context:80/context.jsonld", "https://smart-data-models.github.io/dataModel. Transportation/EVChargingStation/context.jsonld", "https://uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context.jsonld"]

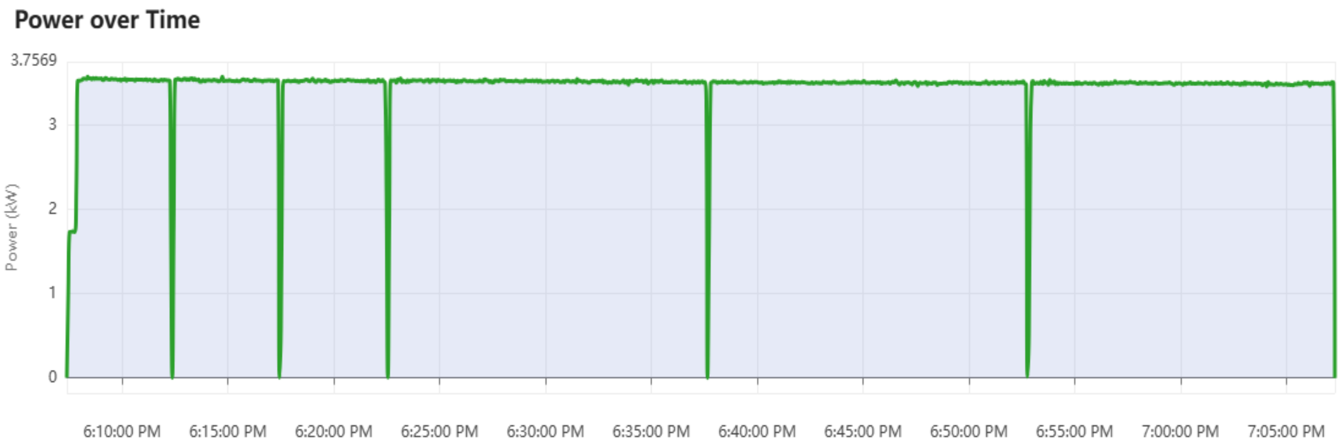

Fig. 4. Power over time during the charging session Source: created by the authors

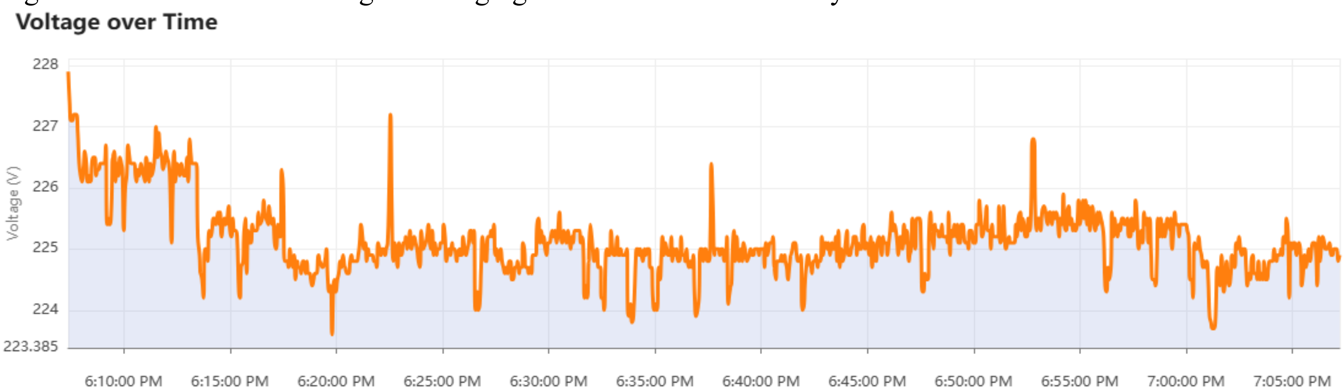

Fig. 5. Voltage over time during the charging session Source: created by the authors

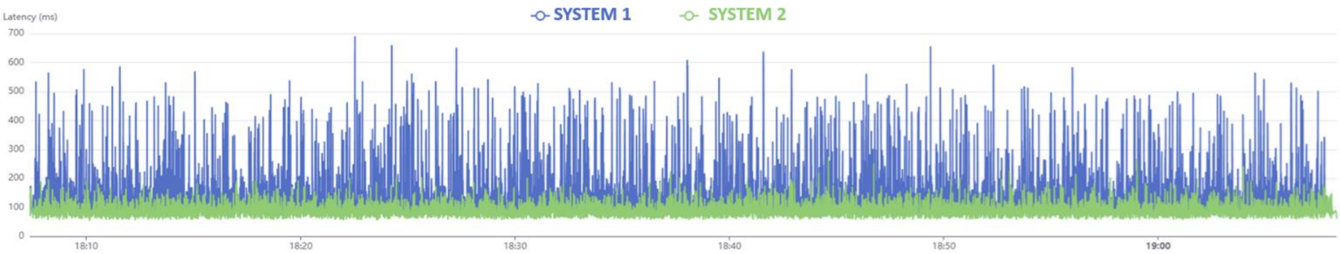

Fig. 6. Latency comparison Source: created by the authors

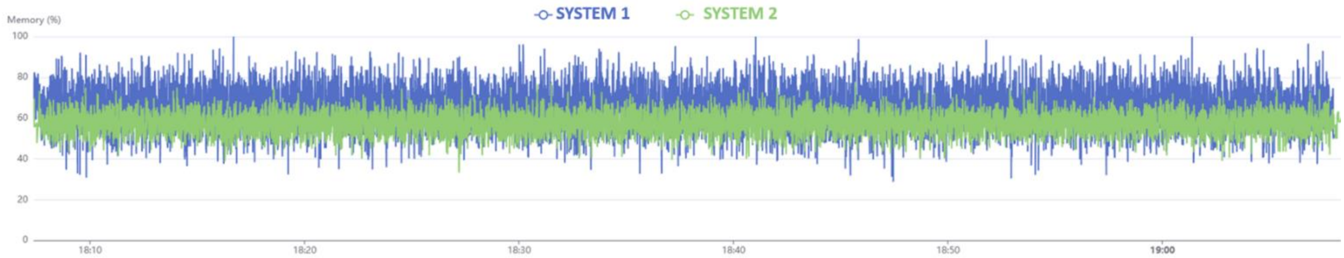

Fig. 7. Memory load comparison Source: created by the authors

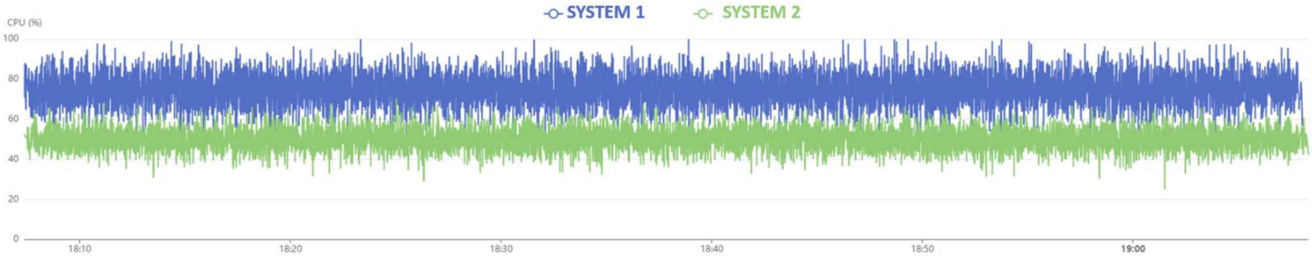

Fig. 8. CPU load comparison Source: created by the authors