Proceedings of the 6th South American Conference on Industrial Engineering and Operations Management
Sorocaba, Sao Paulo, Brazil, May 12-15, 2025

Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/SA6.20250094
Published: May 12, 2025

Edge-based Microservice Orchestration using Node-RED for Process Control in Industry 4.0

Matheus River de Souza, Eduardo A. Viana, Fábio A. Nogueira, Eduardo Paciência Godoy, and Ricardo Pasquati Pontarolli
São Paulo State University (Unesp)
Sorocaba, Sao Paulo, Brazil
matheus.river@unesp.br, eduardo.viana@unesp.br, fa.nogueira@unesp.br,
eduardo.godoy@unesp.br, pasquati.pontarolli@unesp.br

Abstract

The transition to Industry 4.0 is driving the adoption of advanced automation and control systems, integrating microservices and Edge Computing to enhance flexibility, scalability, and efficiency. However, the lack of standardized tools for developing industrial automation applications at the edge poses challenges. This study explores the use of Node-RED as an orchestration tool for Industrial Automation Systems, leveraging a custom block library that emulates IEC function blocks to facilitate microservice-based application development. A practical implementation was conducted in a process control pilot plant, where microservices for data acquisition and control were deployed and orchestrated using Node-RED. The results demonstrated the feasibility of this approach, ensuring real-time control and interoperability in industrial applications while overcoming challenges related to microservice coordination and standardization. Future work will focus on extending the methodology by incorporating IEC 61499 standards and improving microservice monitoring strategies.

Keywords
Edge Computing, Microservices, Industrial Automation Systems, Node-RED.

1. Introduction
The transition to Industry 4.0 (I4.0) is fundamentally reshaping manufacturing by fostering economic, environmental, and social sustainability through digital transformation (GHOBAKHLOO, 2020). The first steps towards a more flexible and fully interconnected industrial automation and control environment have already been taken with the development and adoption of new technologies such as the Industrial Internet of Things (IIoT), Cloud Computing, and Service-Oriented Architectures (SOA) or microservices. These technologies have enabled connectivity, interoperability, and vertical integration of industrial processes, allowing software component reuse and promoting communication between equipment and systems across different hierarchical levels (CARLSSON et al., 2018).

However, the adoption of Cloud Computing in the industry has led to challenges in its implementation, related to latency, dependence on internet connectivity, and data transfer with strict time requirements, which could be obstacles depending on the application (DELSING, 2017). In response to these challenges, Edge Computing has emerged as a solution, decentralizing data processing and storage by bringing these operations closer to the devices and equipment where data is generated. Instead of relying on cloud resources over the Internet, Edge Computing utilizes local communication and processing networks (computers, gateways, or embedded systems) to perform these tasks (CAO et al., 2020).

Edge Computing, when combined with a microservice architecture, offers numerous advantages for I4.0, such as reduced latency, increased network efficiency, resilience, scalability, and cost reduction. By decentralizing information processing, Edge Computing enables real-time data access and improves data transfer efficiency. The microservices approach involves dividing a system or application into smaller, independent components microservices that perform specific functions. These microservices are self-contained and can be developed, deployed, and scaled separately, thus enhancing flexibility, modularity, and resilience in industrial applications (PONTAROLLI, 2023).

Despite these benefits, the widespread adoption of microservices and Edge Computing applications in I4.0 still faces significant challenges. The lack of standardized tools for developing microservice applications at the edge hinders uniform development (HOSSAIN et al., 2023). Additionally, the IEC 61131 standard, which governs the design and programming of industrial automation controllers, has major limitations in terms of connectivity and compatibility with emerging I4.0 technologies while IEC 61131 has been widely accepted and used in industrial automation, it is inherently designed for monolithic and centralized control systems, making it unsuitable for distributed, service-oriented, and highly flexible architectures required in I4.0 (BABAYIGIT and ABUBAKER, 2024).

Meanwhile, IT tools such as Node-RED provide better support for microservices and Edge Computing but lack standardization, often leading to proprietary and difficult-to-maintain industrial applications (LARRINAGA et al., 2022). Given these challenges, it is crucial to investigate how to enhance the support for microservices and Edge Computing in industrial control and automation scenarios by bridging the gap between traditional OT tools and modern IT advancements.

Therefore, this study explored the use of Node-RED as a tool for developing and deploying automation and control applications in I4.0 using microservices and Edge Computing. Node-RED is a tool that simplifies the creation, integration, and coordination of microservices, enabling efficient communication between edge devices. Its capability to integrate with various devices, communication protocols, and services enhances interoperability and communication. Additionally, Node-RED can run on different types of hardware, expanding the possibilities for developing Edge Computing applications. This approach was implemented and experimentally tested in a pilot plant.

1.1 Objectives

This research aims to develop automation and process control applications using microservices within the industry 4.0 context. To achieve this, Node-RED is utilized as both a microservice orchestration tool and an Edge Computing development platform. The proposed applications are implemented and tested in a pilot plant for industrial processes, ensuring practical validation of the approach.

2. Literature Review

Industry 4.0 has significantly advanced digitalization in the industrial sector, emphasizing the importance of cloud computing and service-oriented architectures. These technologies have enhanced interaction among industrial systems and optimized information exchange, leading to greater efficiency and connectivity in smart infrastructures (JAVED et al., 2023) and (CARLSSON et al., 2018). In Industrial Automation Systems (IAS), they have transformed cross-layer integration and improved data processing and manipulation, even for traditional automation equipment with limited computing power. However, IAS often requires real-time analysis and decision-making, which cloud computing alone cannot fully support. To address these challenges, Edge Computing has gained prominence (ARGUNG, 2023).

Edge Computing in Industry 4.0 follows a computational model that reduces latency by positioning edge devices close to end users. Moreover, integrating Edge Computing with microservices has enabled Industry 4.0 applications to function as services, enhancing versatility, IT system integration, and support for IIoT technologies such as data analytics and Artificial Intelligence (AI) (BABAYIGIT and ABUBAKER, 2023). These applications primarily operate within the upper layers of the ISA-95 reference architecture, including SCADA, MES, and ERP. Such solutions are predominantly software-based, benefiting from IT advancements that facilitate integration with web technologies, service orientation, and virtualization. As a result, web-based SCADA and MES systems, along with SCADA tools such as process historians, data aggregation, and manipulation, are commonly deployed at the edge, while ERP systems typically reside in the cloud (DASS et al., 2023).

Despite these advancements, enhancing IAS support at lower levels remains essential to fully leverage Edge Computing and service-oriented approaches (LANGMANN and STILLER, 2019). Traditionally, data acquisition, control, and automation have relied on hardware-dedicated, single-task, and standardized solutions with limited IT integration (CAO et al., 2021). Conventional IAS equipment, including Programmable Logic Controllers (PLC), Distributed Control Systems (DCS), and development software based on IEC 61131 and 61499, was not originally designed for edge applications, such as shared processing, and does not inherently support microservices.

Microservice integration is typically achieved through frameworks that use Ethernet TCP/IP or web service communication rather than traditional industrial networks (FRANCESCO et al., 2018). Although IAS development tools (IEC 61131 and 61499) support these communication methods, integrating microservices remains complex due to commercial tool limitations that restrict custom solution development. This integration often requires specific function blocks, which may not be available in commercial tools, or the development of communication drivers to support microservices.

Recent approaches leveraging the IEC 61499 standard have demonstrated improved flexibility in developing distributed automation solutions, aligning industrial applications with microservice-based architectures and enabling better orchestration of edge computing resources (DAI et al., 2023). Consequently, IAS applications utilizing microservices and running at the edge demand flexible hardware and are primarily developed using open automation tools such as Codesys, OpenPLC, and 4DIAC, enabling customization (MANDIĆ et al., 2022).

Edge PLCs have recently emerged as a viable solution for developing IAS applications in Industry 4.0. These devices, often based on single-board computers, perform the same operational technology (OT) tasks as PLCs while incorporating IT functionalities necessary for microservice-based IAS applications at the edge. According to MANDIĆ et al. (2022), key features of edge devices include a Linux-based operating system, support for multiple programming languages (Python, JavaScript, C), compatibility with development frameworks (e.g., Node-RED), containerization by Docker for virtualization, and database support for data storage. Several studies have explored the development of IAS applications using microservices and edge devices within the Industry 4.0 framework.

Remote I/O is responsible for acquiring sensor and actuator data in IAS. Distributed I/O represents an evolution of Remote I/O, incorporating processing capabilities and network communication support. CHAWDA and NISHIYAMA (2022) proposed an extended Remote I/O solution that combines edge device operation with visual programming in Node-RED. Similarly, PONTAROLLI et al. (2022) developed a distributed I/O system as a microservice, utilizing an edge device with I/O expansion modules and a microservice framework for service-oriented data acquisition in IAS. While these approaches have improved connectivity in Industry 4.0, they lack standardization and often result in proprietary or black-box industrial applications, complicating maintenance and updates.

PONTAROLLI et al. (2023) introduced a microservice-based architecture designed to support IAS applications in Industry 4.0. Built on the Moleculer framework, this architecture enables IAS functionalities such as data acquisition, process control, and SCADA to operate as microservices. In these applications, microservices must be orchestrated in a predefined execution sequence. However, the orchestration tool used in this implementation is neither compatible with IAS standards nor optimized for edge applications. This research aims to enhance the architecture by ensuring that the orchestration application runs at the edge and is developed using an IT tool that aligns with industrial block programming principles.

To achieve this, the Node-RED tool has been integrated into the framework. Node-RED is a widely used IT development tool for IAS and IIoT connectivity (EMBONG et al., 2024) that effectively utilizes Edge Computing. This approach builds on the concept of using Node-RED for microservice orchestration (LARRINAGA et al., 2022) while extending its capabilities by creating programming blocks for microservices within Node-RED. These blocks are designed to resemble or replicate IEC function blocks used in industrial tools, facilitating their adoption in IAS applications.

3. Methods

This section presents the methodology used in this study, detailing the approach for developing and integrating microservices in industrial automation. The research focuses on implementing a control and automation system using an orchestrator, transporter, and industrial microservices deployed in an Edge Computing environment.

3.1 Methodology

This research investigates software tools for the development of automation and control applications using microservices and edge computing in I4.0. The main challenge to do this is the lack of tools and standardization to create microservice IAS applications at the edge. Typically, IAS tools are standardized but do not properly support microservice and edge functionalities. Also, IT tools provide better support for microservices and edge computing, but at the cost of creating proprietary and non-standardized applications. Therefore, the research problem is how to properly integrate the elements of figure 1: IAS Orchestrator, transporter, pilot plant.

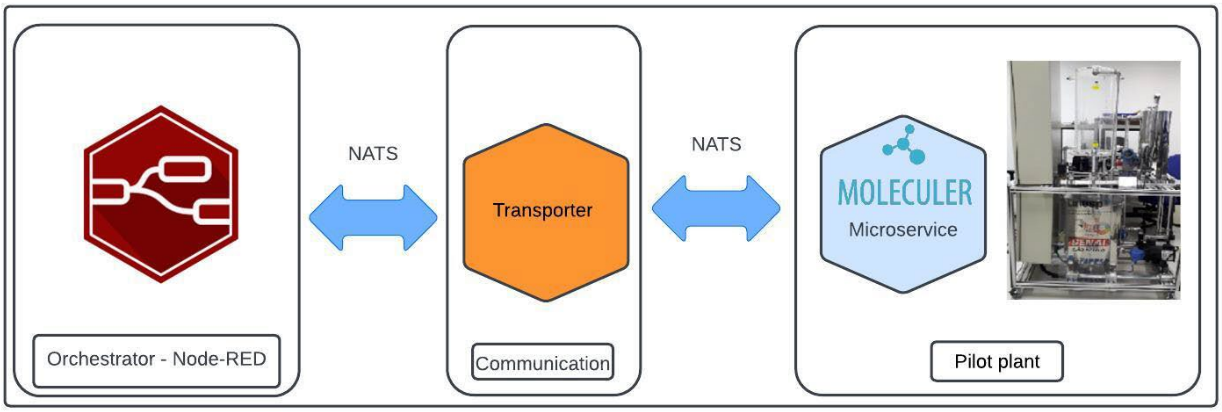

Figure 1. Block diagram for proposed scenario

The industrial process is any process that can be controlled or automated. Industrial microservices are functionalities provided as services using a service framework. The orchestrator is the software tool for creating and deploying the control and automation application. The communication is the necessary connection between the orchestrator application and the microservices.

The research focused on investigating, developing, and testing the use of Node-RED as an orchestrator for IAS applications. A block library was developed to standardize microservice programming and orchestration, simplifying the creation of control applications by abstracting the complexity of microservice interactions. Through a process called service composition, multiple microservices are orchestrated and executed in a specific sequence. Additionally, a communication was implemented to enable communication between Node-RED and the microservices. All microservices operate on single-board computers within a local network, with applications deployed in an Edge Computing structure to enhance integration and efficiency.

The microservices for IAS were previously developed using the Moleculer framework, and the industrial process is a process control pilot plant with IAS equipment adapted to be used with the microservices. The edge computing architecture for microservice and orchestrator deployment is based on several single-board computers (Raspberry Pi) on the network infrastructure. The development and testing of this scenario are detailed in the next sections of the paper.

3.2 Pilot plant 4.0

The Pilot Plant 4.0 is a process control plant adapted for use with microservices (PONTAROLLI et al., 2020). It includes edge devices for microservices and orchestrators, along with all necessary IAS equipment such as sensors (reservoir pressure, tank level, tank temperature, pipe pressure, and pipe flow) and actuators (water pumps with frequency inverters and an electronic proportional valve).

The Pilot Plant 4.0 features multiple single-board computers (Raspberry Pi 3B+ and 4B) operating within a network where microservices and orchestration applications are developed and deployed across different edge devices. Among the available microservices, this study utilized the data acquisition microservice (DAQ) and the PID4.0 microservice. The DAQ microservice is responsible for reading sensor inputs and updating actuator outputs, functioning as a distributed I/O system (PONTAROLLI et al., 2022). To ensure compatibility with industrial peripherals (e.g., analog and digital signals), it incorporates an industrial I/O module alongside the edge device hardware.

The PID4.0 microservice implements a modified PIDPlus control algorithm for networked control, designed for closed-loop control applications (PONTAROLLI et al., 2023). Figure 2 provides an overview of the Pilot Plant 4.0.

Figure 2. Pilot Plant 4.0 overview: (a) Internal View; (b) External View; (c) Side View

3.3 Software Tools

The software tools used in this study include the Moleculer framework, which enables the development of microservices, and Node-RED, which serves as an orchestrator for industrial automation applications. These tools facilitate system integration, communication, and process control within an Industry 4.0 environment.

3.3.1 Moleculer framework

Moleculer (MOLECULER, 2023) is an open-source development platform based on microservices that uses the JavaScript programming language. This framework enables the creation of efficient and scalable microservices in Node.js, allowing them to access peripheral devices where they are installed and making them available within a network structure.

The Figure 3 illustrates the MOA architecture of the Moleculer framework, where services run on individual servers (nodes) that communicate through a messaging microservice (transporter), also known as a message broker, represented by the orange hexagon. The communication service is a crucial component of the architecture, functioning as the mechanism that coordinates message transmission and reception within a queue-based system (PONTAROLLI, 2020).

Each microservice within the architecture is represented by a blue hexagon. Microservices can provide and execute different tasks, referred to as actions. The green hexagon represents the gateway microservice (API Gateway), which serves as the connection interface between internal services (blue hexagons) and external applications through network or cloud calls using the REST protocol.

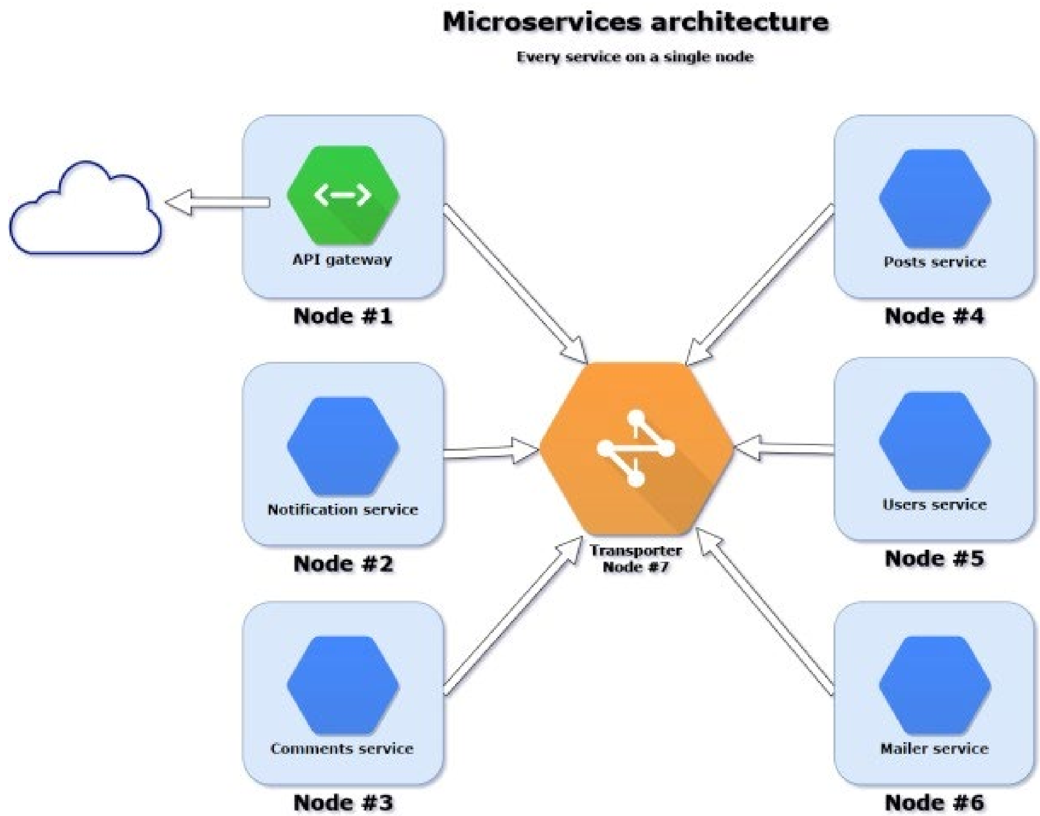

Figure 3. Framework moleculer architecture

The main features of the Moleculer framework are (MOLECULER, 2023): request-response concept, event-driven architecture, middleware support for Node.js, variable caching support, multiple communication options (Transporters), various serializer options such as JavaScript Object Notation (JSON), MsgPack, and Protocol Buffer, support for running multiple services on the same node, native support for service registration, automatic service discovery, and an API for interfacing with external applications. This framework was used as the foundation for developing a microservices architecture for automation and control applications in Industry 4.0 (PONTAROLLI, 2023). This architecture provides various microservices that will be utilized in this project.

3.3.2 Node-RED

Node-RED is an open-source visual development platform that has been widely used in Industry 4.0 due to its ability to integrate systems and automate processes. It offers an intuitive flow-based interface, allowing users to create applications and workflows by connecting predefined blocks, known as "nodes," through a graphical interface. Node-RED has become one of the leading tools for implementing Edge Computing solutions and has been incorporated into most industrial IIoT and Industry 4.0 platforms.

In this project, Node-RED will be used alongside a block library compatible with the Moleculer framework. This will enable the creation of control applications for the loops available in the plant. Additionally, Node-RED includes block libraries that facilitate the development of dashboards, which will provide an intuitive visual interface for users. The required applications will be developed by creating programs (flows) that coordinate the plant’s microservices according to the desired functionalities (LARRINAGA et al., 2022).

4. Development

This section presents the development of the block library for Node-RED and its application in process control. The created blocks follow the IEC standard structure but internally make requests to microservices using the Moleculer framework. For example, the analog output voltage write block sends commands to the DAQ microservice to set an analog value (0 to 10 VDC) on a specific output of the I/O module. Table 1 provides a complete list of the developed blocks and their functions.

To validate this approach, applications were created for the control and monitoring of four loops in the pilot plant: tank level, tank pressure, line pressure, and flow rate. The implementation followed two main stages. First, a block library was developed to facilitate the use of microservices and simplify the creation of new applications. Then, Node-RED flows were structured to control the loops and generate interactive dashboards for system performance monitoring.

Table 1. Node-RED Moleculer framework Microservice blocks

Items	Function Block description	Microservice name in Node-RED
1	Microservice DAQ Analog output voltage write	DAQ_AO_U_WT
2	Microservice DAQ Analog output voltage read	DAQ_AO_U_RD
3	Microservice DAQ Analog input voltage read	DAQ_AI_U
4	Microservice DAQ Analog output current write	DAQ_AO_I_WT
5	Microservice DAQ Analog output current read	DAQ_AO_I_RD
6	Microservice DAQ Analog input current read	DAQ_AI_I
7	Microservice DAQ Output write Open Drain	DAQ_AO_OD_WT
8	Microservice DAQ Output read Open Drain	DAQ_AO_OD_RD
9	Microservice DAQ Digital in read Opto	DAQ_DI_OPTO
10	Microservice PID4.0	PID4.0

4.1 Node-RED Block Library

The Node-RED allows the development of customized block libraries using the JavaScript language. After being programmed, these custom blocks can be generated and added to the Node-RED tool. These blocks allow user parameterization in addition to configuration through the traditional connection of wires between the inputs and outputs of the blocks. Although it is a flexible and user-friendly interface, the development of applications does not follow a specific standard, which is important in the industrial case, especially in terms of management and maintenance.

As a result, this article proposed the development of a Node-RED block library that mimics the IEC standard function blocks to improve its industrial usage. This block library was developed and added to the Node-RED pallet for IAS programming.
According to Node-RED (2023), to create a block in Node-RED, three files are required:

- A JavaScript file that defines what the block does;
- An HTML file that defines the block properties, checkboxes, help texts, etc.;
- A JSON file that combines everything into a package as a npm module.

Thus, was created a library (Table 1), using the Moleculer framework as a base. For each available microservice in the plant, a block was developed that uses the call function of Moleculer to invoke the microservice, with the response being returned at the output of the block in Node-RED.

It is important to highlight that both the Node-RED custom blocks and the Moleculer framework for microservices are based on JavaScript programming. As a result, this enabled the joint development of the Node-RED block library and the integration proposed in this article in the same code/block. For the integration part, a generic Node-RED library provided by Moleculer was used as a basis. This library was modified to create a personalized version for the IAS microservices.

In the Node-RED block library, the blocks were created and can be configured with the same inputs and outputs as the IEC blocks. With the Node-RED block library, the programmer can use these blocks to compose automation and control logic in a way similar to the IEC standard, also abstracting the microservice part. This will allow the user to focus on the IAS application rather than dealing with low-level coding, which could cause confusion or misuse of the software and even jeopardize the industrial process.

These blocks can be freely used to compose applications that need to communicate with microservices. As a result, Node-RED acquires the concept of an orchestrator, coordinating the sequence of microservice executions as required by the application. The required applications will be created through the development of programs (flows) that orchestrate the plant's microservices according to the desired functionality (LARRINAGA et al., 2022).

4.2 Loop control

The implementation of loop control requires three microservices. The PID4.0 acts as the controller in a closed-loop system, calculating the actuator's output based on the error between the setpoint and the measured value. The acquisition of sensor data is handled by DAQ-AI-I, while DAQ-AO-U-WT controls the water pumps, adjusting them according to the controller's command. In addition to control, the developed flows include features to facilitate practical use and enable the creation of a dashboard for loop monitoring. The developed flow can be seen in Figure 4.

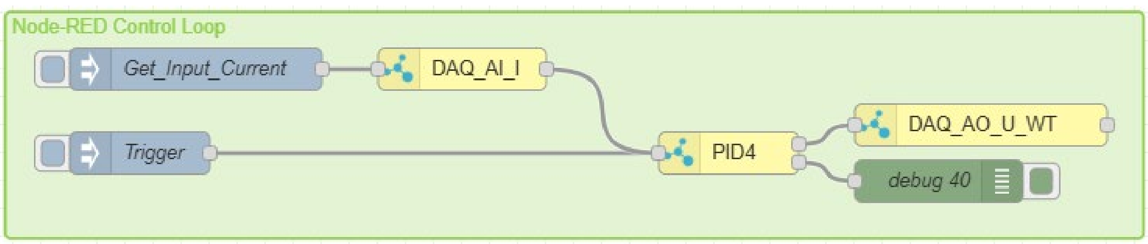

Figure 4. Developed flow for Tank T01 level control loop

The dashboard Figure 5, generated by the Node-RED flow, provides an interactive interface for real-time monitoring of loop control. Its main elements include:

- Tank T01 Level Indicator: Displays the current tank level as a percentage with color-coded status;
- Setpoint Selection: Allows choosing the desired setpoint type (e.g., sinusoidal wave);
- PID Parameter Selection: Enables adjustment of the PID controller parameters (Kp, Td, Ti);
- Control Button: Activates or deactivates loop control;
- "Setpoint vs. Process Variable" Graph: Compares the setpoint and process variable over time;
- "Manipulated Variable vs. Error" Graph: Shows the manipulated variable and control error.

This dashboard facilitates real-time monitoring and adjustments of the tank level control and other process variables.

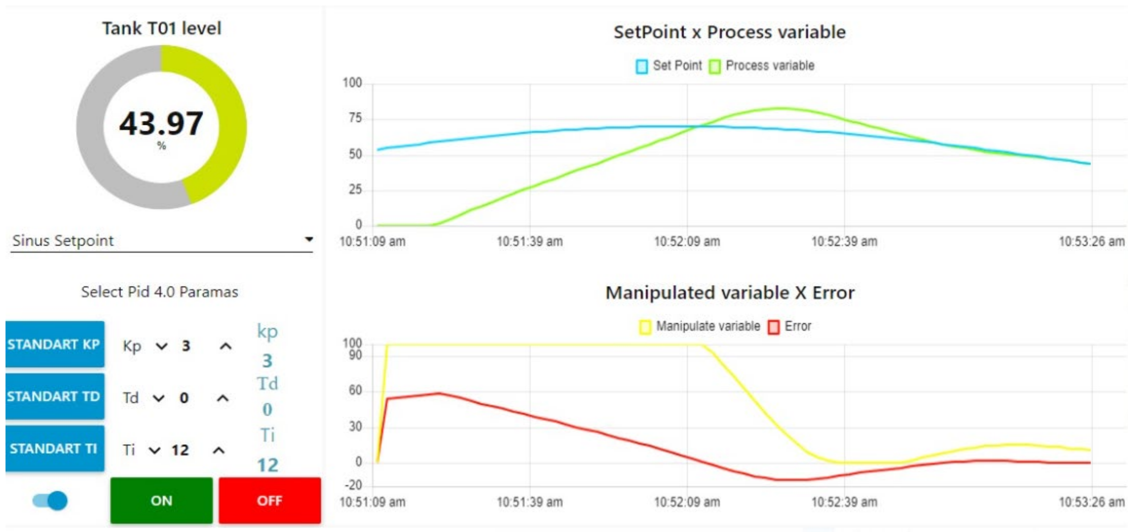

Figure 5. Tank T01 Level control loop dashboard

5. Results and Discussion

The practical experiment was carried out according to the block diagram shown in Figure 1. The orchestrator is the Node-RED platform, which uses the block library to communicate with the Moleculer microservices of the Pilot Plant 4.0.

The closed-loop control orchestration developed in Node-RED is shown in Figure 5. The block library created in this article allowed the creation of a program (flow) like a traditional PLC program, using the Function Block Diagram (FBD) language from IEC 61131-3. This is a contribution of this article, as it can help in the adoption and use of Node-RED in IAS applications.

The responses of the tracking control using the microservice orchestration in Node-RED are shown in Figure 6. For tank level control, a setpoint profile with a cosine waveform was used. The control graph can be seen in Figure 6 graphic A, demonstrating the overlap of the graphs between the tank level setpoint and the process variable LIT 125. Upon analyzing Figure 6 graphic B, similar behavior is observed for the tank pressure control, with a triangular setpoint profile. The control graph shows the overlap of curves between the process variable PIT 129 and the tank pressure setpoint, indicating that the process is adequately controlled.

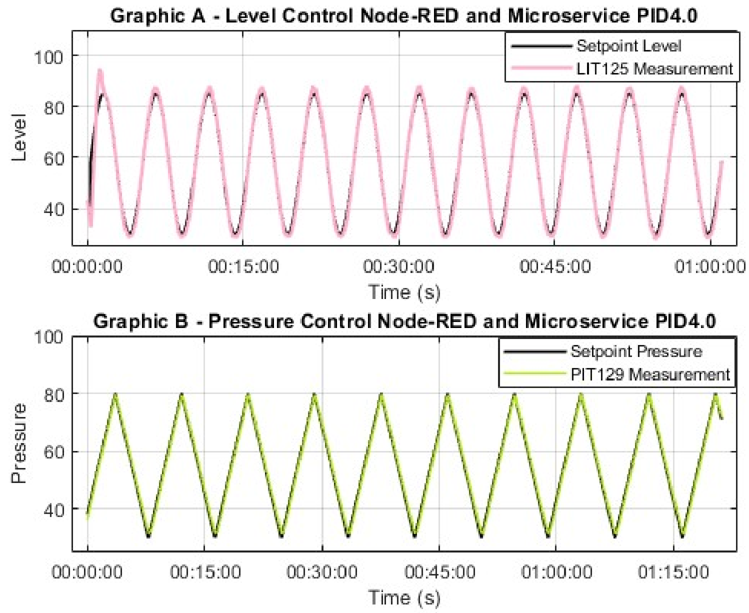

Figure 6. Level and pressure charts controlled by PID4.0

Finally, it is worth highlighting that, since the orchestrator is running at the network edge, it is possible to execute multiple simultaneous control loop executions. In the experiments of this article, the Node-RED orchestrator ran simultaneously in different instances, demonstrating that control and automation applications with microservices for non-reactive processes can handle the latency and jitter at the network edge.

However, it is important to note some challenges related to the use of microservices in this context. Due to their distributed nature, microservices are deployed across multiple devices, making it difficult to manage their availability and status. Monitoring whether a microservice is online or offline, as well as handling message exchanges between different instances, becomes more complex compared to centralized architectures.

To evaluate the performance of the microservices, their response times to requests were analysed. The average response times and their respective standard deviations were measured within Node-RED. Table 2 presents the response time results for each individual microservice, as well as for the complete control loop with all microservices operating together.

Table 2. Request-Response Time Performance of Microservices

Microservice	Average Response Time (ms)	Standard Deviation (ms)
DAQ Analog Input	38.41	16.48
DAQ Analog Output	37.38	16.52
PID 4.0	38.42	23.86
Control Loop (integrated)	90.13	33.07

6. Conclusion

The results of this study demonstrated the feasibility of using Node-RED for developing control and automation applications based on microservices and edge computing. The creation of a block library enabled standardized programming, simplifying the adoption of Node-RED in Industrial Automation Systems (IAS) and improving integration with industrial applications. However, the distributed nature of microservices presents challenges in the management of those microservices.

The experiments conducted in the Pilot Plant 4.0 confirmed that using Node-RED as an orchestrator allows for efficient execution of microservices in an industrial environment. This ensures flexibility and modularity in application development while maintaining scalability through container-based implementation (Docker).

Despite these challenges, this study contributes to the adoption of emerging technologies in Industry 4.0, promoting the integration of microservices in industrial automation. Future work may explore the use of the IEC 61499 standard and tools such as 4DIAC to evaluate new orchestration approaches in IAS, while also refining strategies to improve microservice coordination and monitoring.

Acknowledgement

Research supported by grant 2024/21954-7, São Paulo Research Foundation (FAPESP).

References

Argungu, M. Malami idina, U. Chalawa, U. Ummar, S. Buhari bello, I. Arzika, B. Mala, A survey of edge computing approaches in smart factory, International Journal of Advanced Computer Research 12 , 17–30. 2023. doi:10.17148/IJARCCE.2023.12903.

Babayigit, M. Abubaker, Industrial internet of things: A review of improvements over traditional scada systems for industrial automation, *IEEE Systems Journal* 18 , 120–133. 2024.doi:10.1109/JSYST.2023. 3270620.

Cao, S. Hu, Y. Shi, A. W. Colombo, S. Karnouskos, X. Li, A survey on edge and edge-cloud computing assisted cyber-physical sys tems, *IEEE Transactions on Industrial Informatics* 17, 7806–7819, 2021. doi:10.1109/TII.2021.3073066.

Cao, Y. Liu, G. Meng, Q. Sun, An overview on edge computing research, *IEEE Access 8* , 85714–85728. 2020.doi:10.1109/ACCESS. 2991734.

Carlsson, J. Delsing, F. Arrigucci, A. W. Colombo, T. Bangemann, P. Nappey, Migration of industrial process control systems to serviceoriented architectures, *International Journal of Computer Integrated Manufacturing* 31 , 156–165, 2018. URL: https://doi.org/10.1080/ 0951192X.2017.1392615.

Darshan Chawda, Rapid factory control system development using node-red to expand industrial automation, *Conference Paper's Information* 122 , URL: https://ken.ieice.org/ken/ 23, 2022. paper/20220520UC9F/eng/. doi:https://ken.ieice.org/ken/paper/ 20220520UC9F/eng/.

Dass, M. Nayak, S. Ranjan, A Panigrahi, Cloud Computing in Industrial Automation Systems and its Future, *Research and Applications: Embedded System*, vol. 6, nº 3, p. 8–18,2023, doi: 10.5281/zenodo.10208415.

Delsing J., Local cloud internet of things automation: Technology and business model features of distributed internet of things automation solutions. *IEEE Industrial Electronics Magazine, IEEE*, v. 11, n. 4, p. 8–21, 2017. URL: https://ieeexplore.ieee.org/document/8241149, dOI: 10.1109/MIE.2017.2759342.

Embong, A>; L. Asbollah; S. A. Hamid. Empowering industrial automation labs with iot: A case study on real-time monitoring and control of induction motors using siemens plc and node-red. *Journal of Mechanical Engineering and Sciences*, p. 10004–10016, 2024. URL: https://journal.ump.edu.my/jmes/article/view/9600. doi:10.15282/jmes.18.2.2024.3.0790.

Francesco, P. Lago, I. Malavolta, Migrating towards Microservice Architectures: an Industrial Survey, *IEEE International Conference on Software Architecture*, 2018. doi:10.1109/ICSA.2018.00012.

Ghobakhloo, M. Industry 4.0, digitization, and opportunities for sustainability. *Journal of Cleaner Production*, v. 252, p. 119869, 2020. ISSN 0959-6526. URL: https://sciencedirect.com/science/article/pii/S0959652619347390.

Hossain, M. D. et al. The role of microservice approach in edge computing: Opportunities, challenges, and research directions. *ICT Express*, v. 9, n. 6, p. 1162–1182, 2023. ISSN 2405-9595. URL: https://sciencedirect.com/science/article/pii/S2405959523000760.

Javed, A. Tripathy, J. van Deventer, H. Mokayed, C. Paniagua, & J. Delsing , An approach towards demand response optimization at the edge in smart energy systems using local clouds. *Smart Energy*, *12*, 100123.

Langmann, M. Stiller, The plc as a smart service in industry 4.0 production systems, *Applied Sciences 9* , 2019. URL: https://www. mdpi.com/2076-3417/9/18/3815. doi:10.3390/app9183815.

Larrinaga, F., W. Ochoa, A. Perez, J. Cuenca, J. Legaristi, M. Illarramendi, Node-red workflow manager for edge service orchestration, in: *NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium*, pp. 1–6. 2022, doi:10.1109/NOMS54207.2022.9789940.

Mandić, S. Stankovski, G. Ostojić, B. Popović, Potential of edge computing plcs in industrial automation, in: *2022 21st International Symposium INFOTEH-JAHORINA (INFOTEH)*, pp. 1–5. 2022, doi:10.1109/INFOTEH53737.2022.9751324.

Moleculer. Progressive microservices framework for node.js, URL: https://moleculer.services/, 2023.

Node-RED. Creating your first Node-RED node, URL: https://nodered.org/docs/creating-nodes/first-node, 2023.

Pontarolli, J. A. Bigheti, F. O. Domingues, L. B. de Sá, E. P. Godoy, Distributed i/o as a service: A data acquisition solution to industry 4.0, *HardwareX 12*, e00355. 2022. URL: www. sciencedirect.com/science/article/pii/S2468067222001006. doi:https://doi.org/10.1016/j.ohx.2022.e00355.

Pontarolli, J. A. Bigheti, L. B. de Sá, E. P. Godoy, Microservice-Oriented Architecture for Industry 4.0, Eng 4 1179–1197, 2023. URL: https://www.mdpi.com/2673-4117/4/2/69. doi:10.3390/ eng4020069.

Pontarolli, J. A. Bigheti, S. L. Risso, F. O. Domingues, M. M. Fernandes, E. P. Godoy, Pilot Plant 4.0: A Service-Oriented Approach to Process Automation and Control, *Brazilian Congress in Automatics - CBA 2* , 2020. URL: https://doi.org/10.48011/asba.v2i1.1741.

Biographies

Matheus River de Souza is an undergraduate student in Control and Automation Engineering at the São Paulo State University "Júlio de Mesquita Filho" (UNESP). Technical degree in Mechatronics from ETEC Rubens de Faria e Souza (2019), with research interests in IoT and Industry 4.0. https://lattes.cnpq.br/1172582340983050

Eduardo Paciencia Godoy received the B.Eng. degree in Control and Automation Engineering at Itajubá Federal University (MG-Brazil) in 2003 and the M.Sc. and Ph.D. degrees in Mechanical Engineering at University of São Paulo (SP-Brazil) in 2007 and 2011. Currently he is an Associate Professor of Unesp (SP-Brazil) with research interests in IoT and Industry 4.0. http://lattes.cnpq.br/0072632067545698.

Ricardo Pasquati Pontarolli Holds a degree in Electronics Technology – Industrial Automation from the Faculty of Technology of Tatuí (SP) (2011) and a degree in Computer Engineering from the Virtual University of the State of São Paulo (UNIVESP) (2024), as well as a degree in the Special Teacher Training Program for Basic Education from the Federal Institute of São Paulo (2018), and a master's degree in Electrical Engineering from the São Paulo State University "Júlio de Mesquita Filho" (2020). Was a visiting researcher at the University of Brescia (Italy) in 2024 (sandwich PhD). Currently works as an Industrial Automation Laboratory Technician at the Federal Institute of São Paulo – Boituva Campus. Has experience in the fields of Computing, Electrical Engineering, and Automation. http://lattes.cnpq.br/6917700616010402.

Fábio A. Nogueira is a technologist in Electronics – Industrial Automation from the Faculty of Technology of Tatui (SP-Brazil) in 2008. He is currently studying for a master's degree. Specialist in Automation Engineering and Industrial Electronics from Unyleya and professor at the Faculty of Technology (FATEC) in Tatui (SP-Brazil). His research interests include development and programming of embedded systems aimed at the Internet of Things (IoT) and the Internet of Industrial Things (IIoT), with a focus on Industry 4.0, digital twins and related technologies. http://lattes.cnpq.br/3288692774567000.

Eduardo A. Viana Holds a degree in Electrical Engineering from the São Paulo State University "Júlio de Mesquita Filho" (2001). Completed a postgraduate program in Project Management at FGV–Rio de Janeiro (2011). Holds a master's degree in Electrical Engineering from UNESP in Sorocaba, SP (2024). He is a co-owner of Global Force Engenharia, where he works as Engineering Manager. http://lattes.cnpq.br/4455400060874674.