Improving Industrial Interoperability and Scalability Through OPC-UA and Smart Object-Based Architectures

Guilherme Coelho[1], Liliana Antão[1][a], Beatriz Coutinho[1][b], Gil Gonçalves[1][c], António Augusto[2] and Miguel Moura[2]

[1] SYSTEC ARISE, Faculdade de Engenharia, Universidade do Porto, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal

[2] Idepa - Indústria de Passamanarias, Lda, Portugal

Keywords: OPC-UA, Industry 4.0, Smart Object, Data Model.

Abstract: This paper presents the development and implementation of a remote monitoring and control system for industrial machines, aligned with the principles of Industry 4.0. The proposed solution builds under the Advanced4i work package of the PRODUTECH R3 initiative, addressing key limitations in industrial digitalization scalability, responsiveness, and usability. A redesigned architecture is introduced, build upon a previous data model and architecture by Neto et al., featuring advanced communication protocols, a refactored LabVIEW-based interface, and a middleware layer to enhance data flow and synchronization. A structured data model and an optimized graphical user interface further enable real-time monitoring and remote configuration. The system was validated in a real-world deployment at IDEPA, a real manufacturing company a leader in the labels, tapes, and textile accessories markets , integrating over 30 sensors, and subsequently scaled to support up to 85 sensors with minimal data loss and high responsiveness under operational stress. The results demonstrate capability of maintaining monitoring performance and scalability, offering a practical roadmap for the deployment of interoperable and modular Industry 4.0 solutions in manufacturing environments.

1 INTRODUCTION

Industry 4.0 has transformed manufacturing through digitalization, real-time data exchange, and intelligent automation. A core requirement of this shift is the ability to remotely monitor and control machinery to optimize processes, reduce costs, and improve efficiency. However, many current solutions struggle with scalability and usability in complex, heterogeneous environments (Pisching et al., 2018; Boyes et al., 2018; Danys et al., 2022).

To address these challenges, the Reference Architecture Model for Industry 4.0 (RAMI 4.0) provides a structured framework for designing and deploying Industry 4.0 systems. By combining product lifecycles, hierarchy levels, data types, and technology layers into a multidimensional Service-Oriented Architecture(SOA), RAMI 4.0 enables consistent modeling and analysis of smart manufacturing systems (Pisching et al., 2018; Morgan et al., 2021).

Furthermore, the inclusion of the Asset Administration Shell (AAS) within the RAMI 4.0 framework significantly enhances its practical relevance. The AAS serves as the "digital twin" of physical assets, enabling standardized data management and interoperability across diverse industrial platforms (Pisching et al., 2018).

This paper presents a scalable and modular system architecture aligned with RAMI 4.0 principles, designed to enhance remote monitoring, configurability, and real-time diagnostics in industrial automation. The proposed solution emphasizes fault isolation, dynamic reconfiguration, and integration with existing OPC Unified Architecture (OPC UA) infrastructures, addressing key limitations of current monitoring systems. Furthermore, it contrasts the proposed design with the architecture previously implemented in (Neto et al., 2019), highlighting key differences such as the introduction of a middleware layer and modifications to the LabView environment.

[a] https://orcid.org/0000-0002-2903-0989
[b] https://orcid.org/0009-0000-9769-6726
[c] https://orcid.org/0000-0001-7757-7308

Coelho, G., Antão, L., Coutinho, B., Gonçalves, G., Augusto, A. and Moura, M.
Improving Industrial Interoperability and Scalability Through OPC-UA and Smart Object-Based Architectures.
DOI: 10.5220/0013802900003982

In *Proceedings of the 22nd International Conference on Informatics in Control, Automation and Robotics (ICINCO 2025) - Volume 2*, pages 457-468
ISBN: 978-989-758-770-2; ISSN: 2184-2809

1.1 Advanced 4I PRODUTECH-R3 Project

This work was carried out within the framework of an existing work package known as Advanced4i, which is a part of a larger project called PRODUTECH R3 – Recovery-Resilience-Reindustrialization, a Mobilizing Agenda of Production Technologies for Reindustrialization. Advance4i aims to strengthen the national industry's ability to compete in an increasingly demanding market in terms of sustainability while reducing energy and operating costs. Given the European Union's sustainability targets, it is necessary to integrate solutions based on IoT (Internet of Things), predictive maintenance, and energy efficiency technologies. Energy efficiency, which is often neglected in poorly digitized industrial environments, suffers from a lack of visibility of the associated costs. Advance4i aims to reverse this reality by providing transparency on energy consumption at different levels, from production lines to specific equipment and processes. Specifically, this paper will focus on adapting the basis of the digitalization architecture defined.

1.2 Smart Box

This work builds upon a foundational Industry 4.0 aligned project focused on digitalizing factory operations to improve monitoring, control, and efficiency. The initial implementation at IDEPA a leading manufacturer of labels, tapes, and textile accessories involved the integration of approximately 30 sensors. While successful, the current LabVIEW based solution faces challenges related to scalability, reliability under load, and user accessibility. These limitations, including data loss and system instability, highlight the need for a more robust and user-friendly architecture. This Section reviews the existing implementation, identifies its shortcomings, and outlines potential improvements.

The Smart Box and its related components, shown in Fig. 1, integrates an OPC-UA server, a LabVIEW application, and a Backoffice system, deployed on a CompactRIO 9040 industrial computer. This architecture aimed to enable real-time data exchange and control within a factory environment through *Smart Object Self Descriptor* (SOSD) based configuration.

1.3 Smart Object

The *Smart Object*, also referred to as the *Smart Object Self Descriptor* (SOSD), is a comprehensive data model designed to encapsulate the complete set of system-level information relevant to a given industrial component. As a digital abstraction, the SOSD embodies the essential characteristics of the *Component 4.0 Paradigm*, a foundational concept within the Reference Architectural Model for Industry 4.0 (RAMI 4.0) framework. This alignment ensures robust compliance with key principles of Industry 4.0, particularly those related to *interoperability*, *modularity*, and *decentralized control* (Neto et al., 2020).

The Smart Object model is not merely a static descriptor, but an active, semantically rich representation of both the internal configuration of a component and its dynamic relationships with other entities within the system. By explicitly modeling structural, functional, and behavioral aspects, the SOSD enables seamless integration into Industry 4.0-compliant environments.

In the context of this work, the SOSD serves as the foundational construct for the proposed data model. It facilitates a unified description of system components, allowing for enhanced system-level orchestration and intelligent decision-making. The adoption of this model is instrumental in achieving higher degrees of automation, resilience, and scalability in industrial production systems.

2 RELATED WORK

To address the identified problems, research was conducted to evaluate existing industrial solutions and explore potential technologies that could complement or replace the current system. The investigation began with an analysis of the Industry 4.0 paradigm to understand its relevance and requirements. This was followed by a study of alternative communication protocols and a deeper examination of the advantages and limitations of the OPC-UA protocol that is currently under use.

The digital transformation of industrial sectors has been an increasingly prominent topic of academic and industrial discussion. Several works have explored the intersection between industrialization and digitalization, highlighting the pervasive impact of emerging technologies across different sectors.

In particular, the work in (Neto et al., 2019) employs the same data Modeling approach as this study, this being the Smart Object Self Descriptor (SOSD). Their research validates the relevance and applicability of the chosen methods in analyzing digital transformation trends. The reuse of these approaches, along with their consistent findings, demonstrates that this topic has already received considerable attention in the literature.

Moreover, other studies such as (Neto et al., 2020)

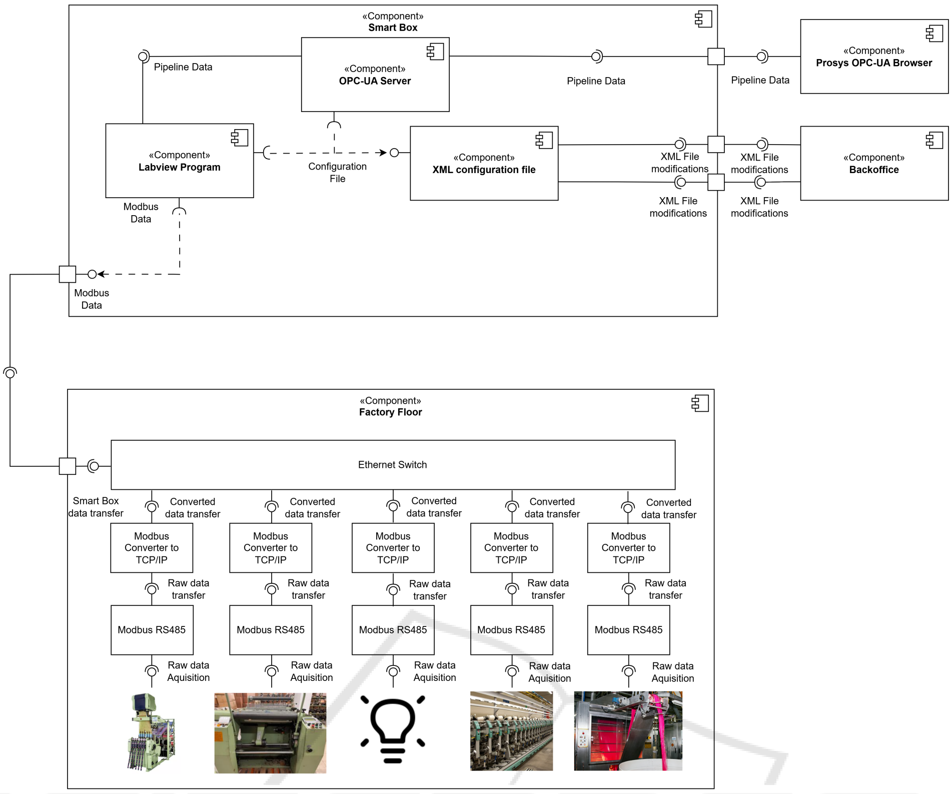

Figure 1: Smart Box and related components.

provide a broader perspective on the digitalization and industrialization of various industries. These works focus on the structural and operational shifts driven by digital technologies, particularly in traditionally non-digital sectors.

For instance, one paper specifically addresses the food industry and its ongoing industrial transformation (Demartini et al., 2018). It highlights how digital technologies are not confined to sectors such as textiles but are also being rapidly adopted in the food sector. The paper also analyzes the number of publications over time, underscoring the growing academic interest and the emergence of this topic in diverse sectors of industry.

Another relevant contribution is the study focused on industrialization within the broader economic sector (Zhang and Li, 2023). Similar to the aforementioned works, it discusses the rising imperative for digitalization across all industrial domains. This work reinforces the notion that digital transformation is not an isolated phenomenon, but rather a global necessity affecting all branches of the industrial economy.

A particularly comprehensive study is offered in (Bauer et al., 2018), which evaluates over 385 Industry 4.0 use cases across German industrial sectors. This large-scale review categorizes applications based on their technological maturity—namely, information, interaction, and intelligence—and sheds light on real-world deployment of digital solutions in domains such as production, logistics, engineering, and services. The findings show that while the majority of applications are still in the "information" phase (data availability and transparency), there is a significant trend toward integrating interaction-level and intelligence-level systems that support real-time responsiveness and autonomous decision-making. The study also emphasizes human-centric approaches, underlining the importance of Human-Technology Interaction and the evolving role of workers in digitalized environments. These insights are crucial in framing digital transformation not only as a technical upgrade but also as a sociotechnical evolution that involves new models of cooperation, substitution, and collaboration between humans and machines.

Together, these studies provide a comprehensive overview of ongoing efforts to understand and support digital transformation in industrial contexts. They collectively emphasize the strategic importance of integrating digital technologies to enhance productivity, efficiency, and innovation across various industrial sectors.

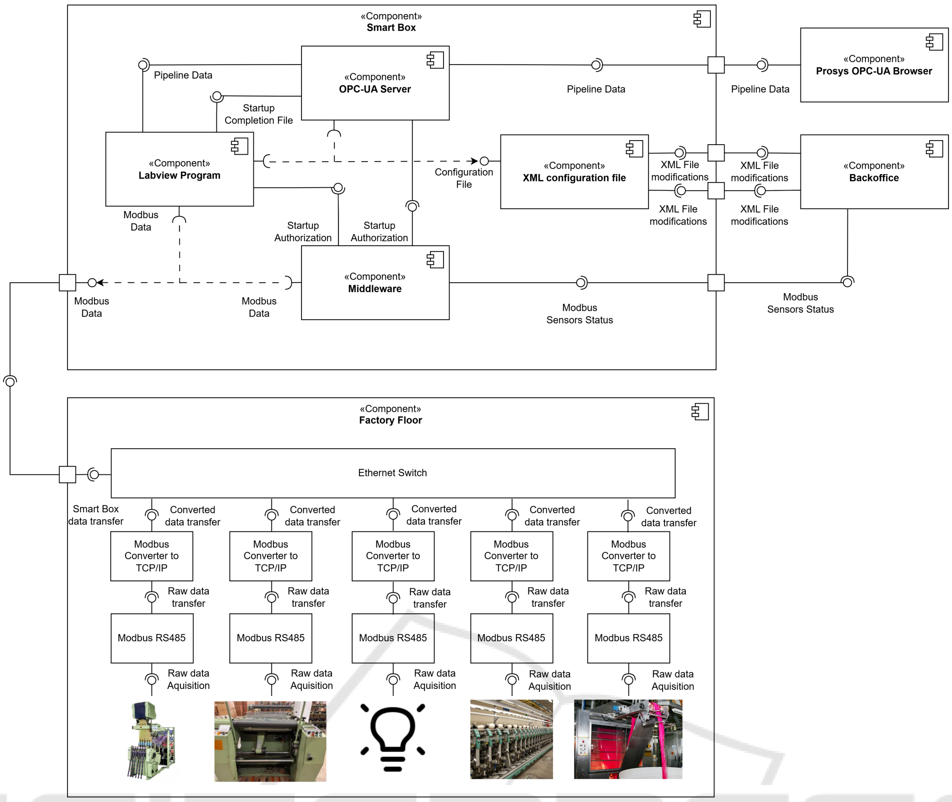

Figure 2: Proposed System architecture.

3 IMPLEMENTATION

In the proposed solution all the modules of the system were subject to change, retaining core functionality while improving system robustness and scalability. The updated design is shown in Figure 2.

A major architectural improvement is the middleware layer, which decouples system components and mitigates previous instability issues. This facilitates modularity, scalability, and easier maintenance. Consequently, a custom middleware was implemented to ensure efficient synchronization and information flow, maintaining system coherence and reliability. The middleware's design and functional benefits are detailed in the following sections.

A refined control and data acquisition system was developed for the IDEPA factory floor using a CompactRIO 9040, serving as a local industrial controller interfacing directly with factory-floor machines and sensors. This unit enables real-time monitoring and control, hosts the LabVIEW-based logic, and integrates an OPC-UA server that exposes sensor data in accordance with the Industry 4.0 (I4.0) component model for standardized communication.

3.1 Data Model

The SOSD model facilitates the transformation of physical assets such as machines, sensors, and actuators into digital entities that are accessible via the OPC-UA network. As a digital representation of Component 4.0, SOSD enables the modeling of complex industrial environments by providing a flexible and standardized framework to define attributes, behaviors, and interrelationships among various manufacturing components (Neto et al., 2020)(Torres et al., 2022).

The SOSD Architecture can be seen in the following Figure 3, where we can see all the components and relationships that compose the model.

The device Set shown in the Figure 3 is composed by equipments and Sensors, which represent 2 different types of data. The Equipments have a description and a collection of Sensors equipments. The Sensor type of data represents a digitalization of the factory floor Sensor, with all its variables and subscribed equipment. This data model is written in modeled Extensible Markup Language (XML).

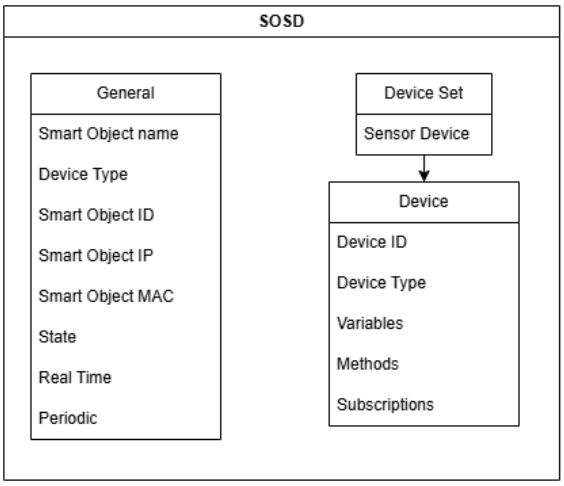

Figure 3: SOSD Architecture.

3.2 OPC-UA Server

The OPC-UA server exposes factory data through a structured *Smart Object Self Descriptor* (SOSD) represented in XML format. This serves as the basis for dynamically generating data pipelines and mapping them to a dedicated namespace. The exposed data is also accessible through standard tools such as the Prosys OPC-UA Viewer, facilitating monitoring and validation of system behavior.

Building upon the existing OPC-UA server infrastructure, several enhancements were introduced to extend its functionality and improve operational robustness. The pipeline creation process was systematically restructured to streamline operations, reducing initialization latency and improving overall startup efficiency. Additionally, a graceful shutdown mechanism was implemented to enable autonomous and reliable termination of the server process without requiring manual intervention, thereby supporting better integration in automated environments.

To further enhance system interoperability, a file-based handshake protocol was introduced to synchronize the OPC-UA server with the LabVIEW interface during startup. This coordination mechanism effectively mitigates race conditions and ensures temporal alignment between components

3.3 Data Acquisition Program

The LabVIEW program operates as the core data acquisition and preprocessing service within the system architecture. It interfaces with the Modbus communication bus to retrieve real-time sensor and equipment data, processes the acquired values, and subsequently transmits them through the data pipelines provisioned by the OPC-UA server. During initialization, the LabVIEW application parses the SOSD XML file to identify and configure the relevant Modbus registers and associated IP addresses. This file is transformed into a structured representation of devices, enabling systematic access to the underlying industrial assets.

To enhance the system's modularity, scalability, and performance, several architectural improvements were introduced to the LabVIEW implementation. The XML parsing logic was refactored to decouple equipment and sensor definitions into two distinct data structures. This abstraction promotes cleaner code organization and facilitates future extensibility. To ensure reliable coordination with the OPC-UA server, a file-based synchronization protocol was implemented. This mechanism enforces a startup dependency, ensuring that LabVIEW initiates communication only after the OPC-UA server is fully operational, thereby eliminating race conditions.

The most significant advancement lies in the redesign of the data acquisition loop to support parallel execution. Each equipment unit is managed within its own dedicated thread, and sensor polling is handled independently, allowing for concurrent Modbus interactions. This multithreaded approach significantly improves the system's responsiveness and fault tolerance—ensuring that communication failures with individual sensors do not propagate through the acquisition cycle. While the transition to parallelism introduces higher CPU load, it also opens opportunities for hardware-specific optimizations to balance performance and resource utilization.

3.4 Middleware Component

A dedicated middleware was developed to coordinate LabVIEW and the OPC-UA server. Beyond managing synchronization, it validates Modbus configurations and logs channel availability. These diagnostics are exposed to the Backoffice interface, providing operators with real-time feedback on sensor status. The middleware also handles server lifecycle management, including shutdown commands, ensuring that system components remain decoupled and independently manageable.

3.5 Industrial Backoffice

Finally, an integral component of the system architecture is the *Backoffice*, a locally hosted web application that provides a graphical interface for real-time system interaction and oversight. Designed as a lightweight yet extensible control layer, the Backoffice facilitates secure, remote access to the SOSD

XML configuration file through a browser-based interface. Authorized personnel can inspect, modify, and manage system configurations without interrupting ongoing data acquisition, enabling continuous system operation and reducing maintenance downtime.

To extend its utility and align with the overall goals of system modularity and observability, the Backoffice interface underwent a series of enhancements. The graphical user interface (GUI) was augmented to support full visibility into system components, including detailed information on Modbus configurations, sensor to equipment mappings, and OPC-UA pipeline statuses. This extended view provides maintainers with granular insights necessary for system diagnostics and performance tuning.

Built on a modern web stack, the updated Backoffice allows interactive manipulation of the XML-based SOSD model. Equipment and sensor entities can be added, removed, or edited through the interface without requiring direct modification of the LabVIEW application or server-side code. These updates are retained persistently and applied during the next system restart, during which the OPC-UA server re-registers all updated elements based on the revised configuration. Furthermore, in-place XML editing, element reordering, and backup creation capabilities have been added to improve usability and configuration traceability.

To support runtime transparency, middleware-generated diagnostics—such as sensor connectivity status, polling errors, and synchronization reports—are now integrated into the Backoffice dashboard. This provides operators with real-time feedback on system health and aids in rapid identification of faults.

These advancements reflect a broader effort to elevate the system's architectural maturity. Improvements to the Backoffice were developed in conjunction with enhancements to other subsystems, including the modularization of the LabVIEW interface, restructuring of the XML schema, increased OPC-UA reliability, and the introduction of a middleware layer to orchestrate inter-component synchronization. Collectively, these improvements foster greater scalability, resilience, and user-centric control within the Industry 4.0-compliant system.

4 TESTING AND RESULTS

This section presents the evaluation outcomes of the proposed industrial monitoring system, with emphasis on scalability, stability, efficiency, and practical deployment, particularly in environments like the IDEPA factory floor.

To assess the scalability and effectiveness of the proposed architecture, a comprehensive test suite was designed to replicate a range of industrial scenarios typical of the IDEPA factory floor. Each test was executed on both the previous and proposed system architectures to allow direct performance comparison.

Five representative operational scenarios were defined to emulate varying levels of system complexity:

- **Low Complexity:** 2 equipments with 2 sensors each.
- **Medium Complexity (Baseline):** 3 equipments with 5 sensors each.
- **High Complexity:** 3 equipments with 10 sensors each.
- **Stress Test:** Maximum load using 6 equipments with 10 sensors and high acquisition frequency.
- **Centralized Deployment:** All sensors connected to a single equipment, compared against the distributed configuration.

Each scenario was run under multiple sampling rates (2s, 1s, 0.5s) to measure responsiveness and reliability under dynamic conditions.

Tests were conducted using a CompactRIO 9040 with NI Linux Real-Time OS, replicating the deployed environment at IDEPA. Sensors aggregate into logical equipments, sending data via an Ethernet switch to the Smart Box. Varying key system parameters helped simulate realistic operating conditions:

- **Equipments:** 2 to 6
- **Sensors/Equipment:** 2 to 10
- **Acquisition Rate:** 2s to 0.5s

The summary of all test configurations is shown in Table 1.

Table 1: Summary of Test Cases and Configuration Parameters.

Test ID	Complexity Level	Equip.	Sensors	Rate (s)
1	Medium	3	5	2
1.1	Medium - centralized	3	5	2
2	Medium	3	5	1
3	Medium	3	5	0.5
4	Low	2	2	2
4.1	Low - centralized	2	2	2
5	Low	2	2	1
6	Low	2	2	0.5
7	High	3	10	2
7.1	High - centralized	3	10	2
8	High	3	10	1
9	High	3	10	0.5
10	Extreme	6	10	2

To objectively assess system performance and ensure comparability between the two architectures, a set of quantitative metrics was defined:

- **OPC UA Server Setup Time:** Time to initialize and expose the full server namespace, measured via server stdout timestamps.
- **CPU Utilization:** Average and peak processor load during runtime, indicating computational efficiency.
- **RAM Usage:** Memory footprint during operation, collected via a Python script for later analysis.
- **System Stability:** Frequency of runtime errors, data loss, or crashes, assessed from test logs.
- **Jitter:** Variability between sensor acquisition and OPC UA node updates, measured and analyzed by data collection done with Python.

4.1 Baseline Evaluation: Tests 1–3

To establish a reliable foundation for subsequent evaluations, a baseline test was executed. This comprises three distinct tests using a setup consisting of three equipment units, each with five sensors, for a total of fifteen sensor data streams. Each test scenario evaluated both the sequential and the proposed (multi-threaded) data acquisition architectures under increasing levels of data acquisition frequency. These tests were designed to provide insight into system behavior in terms of CPU and RAM utilization, data acquisition latency (measured as jitter), and operational responsiveness (startup time).

All test scenarios were run for a duration of approximately 10 minutes, during which relevant performance data was captured for comparative analysis.

Test 1: Low-Frequency Acquisition (2 Seconds)

Test 1 evaluated system performance at a 2-second acquisition interval, representing a low-stress baseline. Table 2 summarizes CPU, RAM, and jitter metrics for both architectures.

Table 2: Comparison of Test 1: 2s Acquisition Interval.

Metric	Sequential	New
CPU Usage Max (%)	99.00	100.00
CPU Usage Avg (%)	8.90	8.16
RAM Usage Avg (%)	33.64	36.20
Jitter Avg. (s)	0.1845	0.0759

The new architecture matches the sequential system's CPU and RAM usage despite its multithreaded design, while achieving significantly lower jitter, indicating improved timing consistency. Figures 4 and 5 illustrate resource usage over time, highlighting reduced initialization overhead and smoother operation in the new solution.

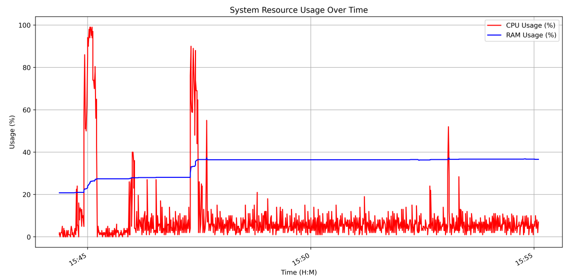

Figure 4: CPU and RAM Usage Over Time — Sequential Architecture (Test 1).

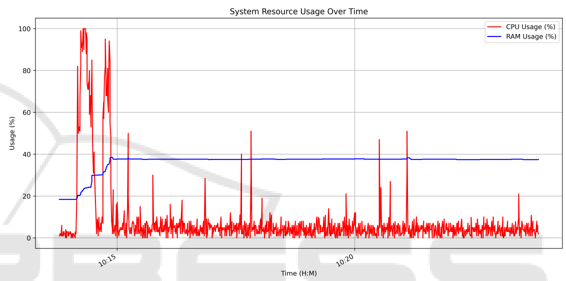

Figure 5: CPU and RAM Usage Over Time — New Architecture (Test 1).

Conclusions:

- Both architectures show similar resource usage.
- The new architecture significantly reduces jitter.
- System responsiveness improves with the new design.

Test 2: Medium-Frequency Acquisition (1 Second)

At a 1-second interval, both systems exhibited slightly increased CPU and RAM usage (Table 3), with the new architecture maintaining lower jitter.

Table 3: Performance Metrics for Test 2: 1s Acquisition Interval.

Metric	Sequential	New
CPU Usage Avg (%)	10.76	9.94
RAM Usage Avg (%)	37.48	37.80
Jitter Avg. (s)	0.1663	0.0505

Resource usage trends are consistent with Test 1. The new architecture sustains better timing precision, with over 50% jitter reduction. The behavior similarity is evident in Figures 6 and 7.

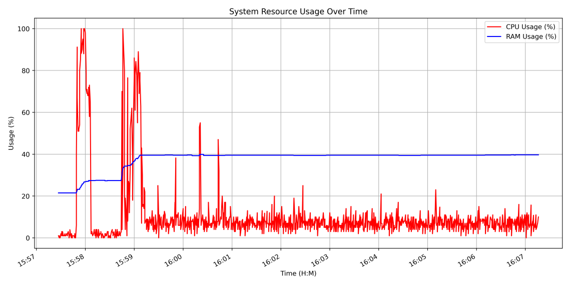

Figure 6: CPU and RAM Usage Over Time — sequential Architecture (Test 2).

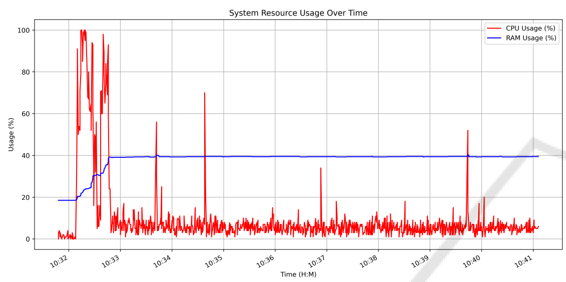

Figure 7: CPU and RAM Usage Over Time — New Architecture (Test 2).

Test 3: High-Frequency Acquisition (0.5 Seconds)

At 0.5 seconds per sample, CPU usage nearly doubled (averaging 16.04% sequential, 14.75% new), with RAM usage stable. The new architecture maintained superior jitter performance, demonstrating scalability under high load (full results omitted for brevity).

Summary for Tests 1–3:

- The new architecture consistently reduces jitter, enhancing timing accuracy.
- Multithreading did not increase resource consumption significantly.
- Faster initialization and responsiveness favor the new design.

4.2 Evaluation of Low Complexity Scenarios: Tests 4–6

This subsection presents the results and analysis of Tests 4, 5, and 6, which evaluate system behavior under low-complexity configurations involving two pieces of equipment, two sensors, and increasing data acquisition frequencies. The tests were conducted, as mentioned before, on a CompactRIO (cRIO) device. Both the sequential and the newly proposed architectures were assessed for comparison.

Performance Results for Test 4

Table 4: Comparison of Test 4: 2 Equipment, 2 Sensors, 2s Acquisition Interval.

Metric	Sequential Solution	Proposed Solution
CPU Usage Max (%)	98.50	99.00
CPU Usage Avg (%)	12.74	10.48
CPU Usage Min (%)	1.50	2.00
RAM Usage Max (%)	32.30	32.90
RAM Usage Avg (%)	30.59	29.87
RAM Usage Min (%)	18.80	17.80
Jitter Equipment 1 (s)	0.2021	0.1892
Jitter Equipment 2 (s)	0.1530	0.1476

The results indicate that both architectures deliver similar jitter performance, with the proposed solution slightly outperforming the sequential system, as per the baseline scenarios. This improvement is likely due to the parallelization employed in the new architecture, contrasting with the serialized approach of the sequential implementation. RAM usage remains comparable, and small differences are attributed to background system processes and runtime variability.

Results for Tests 5 and 6

These tests confirm that both solutions scale effectively with increased data acquisition frequencies in low-complexity scenarios. CPU usage increases proportionally, as expected, while RAM usage remains stable across both implementations. The jitter remains within acceptable limits, with the proposed architecture again achieving slightly better performance due to its parallel execution model.

Conclusions About Tests 4-6

Two primary conclusions can be drawn from these tests:

- Both the sequential and proposed architectures demonstrate similar performance in terms of average CPU and RAM usage, with consistent peak activity.
- The proposed architecture offers a significant reduction in startup time for data acquisition, approximately 45 seconds faster, achieving improved responsiveness without compromising resource efficiency.

These results validate the capability of both systems to handle low-complexity data acquisition scenarios effectively. However, the proposed architecture presents clear advantages in startup latency and jitter optimization, thereby laying the groundwork for superior scalability in more complex environments.

Test 7: Baseline High Complexity Configuration

In this test, both the previous and proposed architectures were evaluated under the same configuration. The proposed architecture demonstrated reliable performance, successfully collecting and displaying data from all 30 sensors with minimal network latency. In contrast, the sequential solution encountered significant limitations. Due to the absence of a handshake mechanism between the OPC-UA server and the LabVIEW interface, the system lacked the ability to optimize server setup. This resulted in system failure during initialization, a previously mentioned limitation where the solution could not handle more than 20 sensors.

Resource usage data for the proposed solution during Test 7 is summarized below:

Table 5: Performance Metrics for Test 7.

Metric	Value
CPU Usage Max (%)	100.00
CPU Usage Avg (%)	12.79
CPU Usage Min (%)	0.00
RAM Usage Max (%)	45.30
RAM Usage Avg (%)	42.87
RAM Usage Min (%)	20.70

Jitter analysis was also performed, with average jitter values recorded as follows:

- Equipment 1: 0.071164 seconds
- Equipment 2: 0.094439 seconds
- Equipment 3: 0.082088 seconds

These results show that, although the resource consumption was higher than in prior tests, the system operated within acceptable thresholds. Jitter remained stable, suggesting the network was not overloaded despite the increased sensor count.

Tests 8 and 9: Increased Acquisition Frequency

Tests 8 and 9 involved gradually increasing the data acquisition frequency. At a 1-second interval, the system remained stable and efficient, showing only marginal increases in CPU, RAM usage, and decrease in jitter. However, when the acquisition frequency was increased to 0.5 seconds, the system encountered a critical issue after approximately 7 minutes. Data loss occurred due to Modbus server disconnections, triggered by the excessive number of requests. The LabVIEW logs confirmed that the Modbus program could not handle the volume of concurrent threads, which points to a potential bottleneck in the LabVIEW Modbus layer rather than the overall system architecture.

Conclusions About Test 7-9

After the conclusion of these tests the following conclusions were made:

- The previous solution is fundamentally limited to handling up to 20 sensors.
- The proposed solution scales effectively to 30 sensors with minor trade-offs in RAM and CPU usage.
- At acquisition frequencies below 1 second, the Modbus Labview implementation becomes a limiting factor, requiring optimization.

4.3 Stress Testing: Test 10

Test 10 was designed to push the architecture to its operational limits. The test utilized a configuration file specifying 60 sensors across 6 different equipment modules, with a 2-second acquisition rate, doubling the complexity of the previously mentioned tests.

Testbed Configuration and Results in the Raspberry Pi 5

Since the IDEPA factory floor didn't have 60 sensors sending data to the Modbus Bus, the system was emulated using three Raspberry Pi 4 units and one Raspberry Pi 5. These devices hosted the Modbus servers and clients, and a custom XML configuration was deployed. The first deployment was carried out on a Raspberry Pi 5, a more modern computing platform than the CompactRIO, albeit not optimized for industrial control. Despite this, the solution worked flawlessly, with no interruption in data collection. CPU and RAM usage were comparable to the CompactRIO in absolute terms, but slightly lower in percentage due to the more capable hardware. Attempts to scale beyond 60 sensors failed due to thread allocation limitations, which was expected based on earlier architectural analysis.

Test Results on CompactRIO

The second phase of Test 10 involved re-deploying the architecture to the CompactRIO testbed present in FEUP. This environment closely simulates the actual IDEPA factory floor. The results on the compact

Rio were more successful, the system was capable of capturing 85 sensors without any issues, in contrast with the previously mentioned 60 on the Raspberry Pi 5. This is the case because the CompactRio is built for industrial setups and data processing, which the Raspberry is not. Attempts to move to the 86th sensor proved unsuccessful.

Conclusion About Test 10

The stress test confirmed the resilience and adaptability of the proposed architecture. The main conclusions are:

- The new system can manage 60 sensors across multiple devices under moderate acquisition frequency.
- It can operate reliably on both industrial-grade and consumer-grade hardware.
- Thread allocation and Modbus handling are the primary bottlenecks for further scalability, especially under high-frequency conditions.

4.4 Conclusion on Tests 1.1, 4.1 and 7.1

The results obtained from Test 4.1 confirmed the expected behavior: the system, configured with 4 sensors and a single equipment unit, performed equivalently to its distributed architecture counterpart. A similar outcome was observed in Test 7.1, further validating the solution's consistency across different configurations.

These results reinforce the conclusion that the proposed architecture is both scalable and robust. It demonstrates that the system can reliably perform its intended functions without requiring changes to the existing factory layout, thus supporting seamless deployment in varied industrial setups.

5 CRITICAL ANALYSIS

This section presents and interprets the key findings obtained through the comprehensive testing and evaluation of the proposed monitoring system. The results are analyzed from an industrial deployment perspective, with a particular focus on scalability, operational stability, system efficiency, and user experience. Emphasis is placed on the system's practical readiness for deployment in real-world manufacturing environments, such as the IDEPA factory floor.

5.1 Performance Evaluation

Testing confirmed that the proposed system offers significant enhancements over the baseline implementation, particularly in terms of scalability, fault isolation, and user operability, key requirements for modern industrial control systems.

A core architectural change was the transition from a sequential to a multithreaded design. Contrary to initial expectations, the multithreaded implementation did not incur excessive computational overhead. The sequential system, operating sequentially, polled sensors linearly, resulting in a single point of failure—where one unresponsive sensor could stall the entire acquisition process. The newly implemented parallel architecture isolates such failures, ensuring uninterrupted data flow from remaining sensors, thus improving system reliability and robustness.

Tests, as detailed in the previous section, were conducted under varying loads, acquisition rates, and sensor configurations using the CompactRIO 9040 hardware—the same controller deployed on the IDEPA factory floor. These scenarios simulated conditions commonly found in medium-sized industrial environments. Performance metrics revealed that both the sequential and new architectures exhibited comparable CPU and memory footprints under most operating conditions. However, the new architecture significantly reduced acquisition jitter, thereby enhancing temporal precision and consistency of data streams.

These results indicate that the proposed solution is well-suited for deployment across industrial sites similar in scope to IDEPA, especially where distributed equipment layouts and real-time reliability are critical.

5.2 Scalability and Limitations

The system demonstrated reliable scalability in controlled test environments, consistently supporting up to 85 sensors during stress tests. This capability exceeds the sensor handling capacity of the previous solution and positions the architecture for broader industrial applicability.

However, limitations emerged when scaling beyond this threshold. During system initialization with over 85 sensors configured via the XML interface, startup failures were observed. These issues were traced to the excessive number of threads created, which exceeded system resource limits. This behavior highlights a practical ceiling on vertical scalability when using the current threading model.

Additionally, during high-frequency acquisition scenarios (0.5-second intervals) with 30 sensors, data loss occurred after sustained operation (approximately seven minutes). This suggests a throughput ceiling beyond which the system cannot guarantee consistent data integrity, signaling the need for further optimization in high-load environments.

5.3 System Evaluation: Strengths and Weaknesses

The proposed system architecture exhibits several notable strengths. One of its key advantages lies in its fault isolation capabilities. By employing a multithreaded design, the system ensures that individual sensor failures are contained within their respective execution threads, thereby preventing fault propagation and maintaining operational continuity—an essential requirement in real-time industrial monitoring environments.

Another significant strength is the robust initialization mechanism achieved through a dedicated handshake protocol between LabVIEW and the OPC UA server. This protocol guarantees an orderly and deterministic startup process, minimizing the likelihood of boot-time errors and enhancing system reliability.

Furthermore, the integration of a centralized configuration management solution via the Backoffice module contributes to streamlined deployment and maintenance. Configuration data stored in XML format enables consistent system setup across multiple factory installations, supporting scalability and ease of maintenance.

The system also supports real-time monitoring by interfacing with the OPC UA Prosys client. This integration provides immediate visualization and live status updates, which are vital for plant-floor diagnostics, control, and decision-making processes.

Despite these advantages, the system also presents certain limitations. The reliance on legacy hardware and software components introduces concerns regarding long-term maintainability and may pose challenges to future upgrades or performance optimization. Additionally, while the multithreaded architecture improves fault tolerance, it also increases computational overhead. This added processing load, although acceptable under current operating conditions, may limit the system's deployability on resource-constrained hardware platforms.

Overall, the system demonstrates a balanced design that favors reliability, configurability, and real-time monitoring, albeit with trade-offs in terms of hardware compatibility and computational efficiency.

6 CONCLUSIONS AND FUTURE WORK

This work demonstrates the value of reengineering legacy industrial systems to meet the demands of Industry 4.0. By modernizing middleware, communication layers, and user interfaces, the proposed solution improves scalability, robustness, and operational transparency. The integration of key principles discussed in this paper is critical for realizing Industry 4.0 objectives, enabling improved operational visibility, reduced downtime, and data-driven decision-making.

The system modernized at IDEPA enhances scalability, usability, and synchronization through several key improvements. Notably, the introduction of a middleware layer resolves startup race conditions and improves component synchronization. In parallel, a modular Modbus acquisition approach enables parallel polling and fault tolerance. While OPC-UA remains the core protocol for standardized data modeling, its limitations under high sensor loads led to the exploration of a complementary Message Queuing Telemetry Transport (MQTT) based architecture. Although full MQTT integration was beyond the project's scope, its advantages in decoupled, scalable messaging were validated and recognized for future use.

The redesigned GUI significantly enhances user interaction by improving error messaging, operational transparency, and control intuitiveness, thereby empowering operators during maintenance and diagnostics.

The system's modular design ensures long-term adaptability and positions it for future integration with open standards and modern protocols. This platform contributes directly to IDEPA's digital transformation goals and offers a scalable reference model for similar industrial deployments. Continued evolution in this direction aligns with the RAMI 4.0 architecture and the broader Industry 4.0 vision.

Future work will focus on modernizing the system architecture to enhance performance, flexibility, and maintainability. A key objective is the migration from the current CompactRIO 9040 platform to a contemporary industrial controller that offers superior processing capabilities and broader hardware openness. In parallel, the communication infrastructure will be restructured around the MQTT protocol to facilitate improved scalability and reduce the overall architectural complexity. Furthermore, the existing LabVIEW-based data acquisition layer is intended to be replaced with an open-source framework—such as Python, Java, or C++ to eliminate licensing dependencies and increase development agility. Finally, the software stack will undergo significant refactoring to eliminate legacy XML-based logic, thereby streamlining data management and better supporting containerized deployment approaches aligned with modern development practices.

ACKNOWLEDGMENT

This work has been supported by the European Union under the Next Generation EU, through a grant of the Portuguese Republic's Recovery and Resilience Plan (RRP) Partnership Agreement, within the scope of the project PRODUTECH R3 – "Agenda Mobilizadora da Fileira das Tecnologias de Produção para a Reindustrialização", Total project investment: 166.988.013,71 Euros; Total Grant: 97.111.730,27 Euros. The authors acknowledge Fundação para a Ciência e a Tecnologia (FCT) for its financial support via the project UIDB/50022/2020 (LAETA Base Funding)

REFERENCES

Bauer, W., Schlund, S., Hornung, T., and Schuler, S. (2018). Digitalization of industrial value chains-a review and evaluation of existing use cases of industry 4.0 in germany logforum digitalization of industrial value chains-a review and evaluation of existing use cases of industry 4.0 in germany. *LogForum*, 14:331–340.

Boyes, H., Hallaq, B., Cunningham, J., and Watson, T. (2018). The industrial internet of things (iiot): An analysis framework. *Computers in Industry*, 101:1–12.

Danys, L., Zolotova, I., Romero, D., Papcun, P., Kajati, E., Jaros, R., Koudelka, P., Koziorek, J., and Martinek, R. (2022). Visible light communication and localization: A study on tracking solutions for industry 4.0 and the operator 4.0. *Journal of Manufacturing Systems*, 64:535–545.

Demartini, M., Pinna, C., Tonelli, F., Terzi, S., Sansone, C., and Testa, C. (2018). Food industry digitalization: from challenges and trends to opportunities and solutions. volume 51, pages 1371–1378. Elsevier B.V.

Morgan, J., Halton, M., Qiao, Y., and Breslin, J. G. (2021). Industry 4.0 smart reconfigurable manufacturing machines.

Neto, L., Gonçalves, G., Torres, P., and Dionísio, R. (2020). On the development of a component model for the realization of industry 4.0. In *2020 IEEE Conference on Industrial Cyberphysical Systems (ICPS)*, volume 1, pages 481–486.

Neto, L., Torres, P., Dionisio, R., Malhão, S., and Gonçalves, G. (2019). An industry 4.0 self description information model for software components contained in the administration shell.

Pisching, M. A., Pessoa, M. A., Junqueira, F., dos Santos Filho, D. J., and Miyagi, P. E. (2018). An architecture based on rami 4.0 to discover equipment to process operations required by products. *Computers and Industrial Engineering*, 125:574–591.

Torres, P., Dionísio, R., Malhão, S., Neto, L., and Gonçalves, G. (2022). Machinery retrofiting for industry 4.0. *Lecture Notes in Mechanical Engineering*, pages 213–220.

Zhang, T. and Li, N. (2023). Measuring digital economy: From the perspective of digital industrialization and industry digitalization. In *Procedia Computer Science*, volume 221, pages 1006–1012. Elsevier B.V.