Seamless Integration of Assets into Industry 4.0 Intelligent Systems: Data and Model Interoperability Analysis

Vitor Furlan de Oliveira
Faculdade Impacta de Tecnologia
São Paulo, Brazil
Center for Engineering, Modeling and Applied Social Sciences
Universidade Federal do ABC
Santo André, Brazil
vitor.furlan@ufabc.edu.br

Ogobuchi Daniel Okey
Center for Engineering, Modeling and Applied Social Sciences
Universidade Federal do ABC
Santo André, Brazil
ORCID: 0000-0003-0686-2763

Gabriel Venturini
Faculdade Impacta de Tecnologia
São Paulo, Brazil
ORCID: 0009-0004-7560-3401

Gabriel de Oliveira Pereira
Faculdade Impacta de Tecnologia
São Paulo, Brazil
ORCID: 0009-0007-2065-0165

Fabrício Junqueira
Escola Politécnica
Universidade de São Paulo
São Paulo, Brazil
ORCID: 0000-0002-4963-2688

Claudio José Bordin Júnior
Center for Engineering, Modeling and Applied Social Sciences
Universidade Federal do ABC
Santo André, Brazil
ORCID: 0000-0002-7016-5922

Rômulo Gonçalves Lins
Center for Engineering, Modeling and Applied Social Sciences
Universidade Federal do ABC
Santo André, Brazil
ORCID: 0000-0001-9878-0081

***Abstract*—The evolution of Industry 4.0 demands seamless integration of intelligent assets capable of interoperating across various systems and contexts. This paper proposes an interoperability evaluation in intelligent systems structured according to the Industry 4.0 principles, namely the Reference Architecture Model Industrie 4.0 and the Asset Administration Shell as a digital representation standard. Two intelligent system architectures were developed and compared: a centralized approach emphasizing data interoperability and a federated learning setup demonstrating model interoperability. Both scenarios were validated using a realistic predictive maintenance dataset, and interoperability was assessed through metrics related to model performance, computational cost, and data quality. Semantic interoperability was also qualitatively analyzed, highlighting the importance of standardized data dictionaries. The results show that while the centralized model achieved higher predictive performance, the federated approach provided computational efficiency and data privacy without loss of interoperability. The study also highlights the impact of technology-related constraints, such as limitations of the machine learning framework, on the outcomes of interoperability. These findings demonstrate the viability of AAS-based architectures for developing intelligent, interoperable Industry 4.0 systems.**

***Index Terms*—Interoperability, Industry 4.0, Intelligent Systems, Federated Learning**

I. Introduction

Industry 4.0 (I4.0) represents a transformation driven by the integration of information and communication technologies into industrial systems [1], responsible for providing new functionalities to assets, in addition to their native functionalities. It aims to create Smart Factories, with intelligent processes that adapt to market demands and enable effective interconnection among system entities [2].

The intelligence of Smart Factories is primarily artificial, often enabled by artificial intelligence (AI) techniques, particularly machine learning (ML) and deep learning (DL), which allow assets to learn from data and adapt to dynamic environments [3], [4]. These techniques support flexibility, production robustness, energy efficiency, and continuous learning, making data a central resource for intelligent behavior in I4.0 systems. Equally important is interoperability, which ensures that assets can be effectively integrated and interact within industrial processes. Several initiatives have highlighted interoperability as a foundational requirement for the digital transformation envisioned by I4.0 [5].

Although prior studies offer valuable frameworks for interoperability in Industry 4.0 [6]–[9], they often lack empirical validation, particularly in distributed or multi-agent contexts, and typically focus on intra-asset integration. This paper presents a practical approach based on the Asset Administration Shell (AAS) for integrating assets into intelligent systems. Each asset is structured as an Industry 4.0 Component (I4.0C), comprising two digital parts specified according to AAS standards. These components interact—either directly or via their virtual counterparts—to form intelligent cyber-physical systems (CPS). The virtual part serves to configure assets and streamline their integration. The proposed approach was evaluated in two scenarios: a centralized system and a distributed system based on federated learning (FL). The main contribution of this work is to empirically demonstrate how I4.0 standards, particularly RAMI 4.0 and AAS, can be effectively applied to support interoperability in intelligent systems.

The remainder of the work is organized as follows: Section II presents a technological context of the interoperability-related topics necessary for this paper. Section III presents a literature analysis of related works. Section IV describes the approach adopted in this work for the development of interoperable intelligent systems from both centralized and federated perspectives. Section V presents the results obtained from experiments conducted with these systems in simulated environments. Finally, Section VI describes the conclusions derived from this paper.

II. Technological Contextualization

A. Interoperability in Intelligent Systems

Interoperability is the ability of devices, applications, and systems to interact, exchange data, and use this information efficiently [10]. This capability is essential for ML and DL models to operate in conjunction with different systems and technologies, eliminating the need for manual adaptations. In this context, interoperability in ML and DL emerges as a crucial aspect to enable different technologies and systems to communicate and operate in different scenarios without compromising efficiency, information integrity or model quality [10].It is not just a technical capability but a central element to effectively integrate diverse systems and technologies, meeting the growing demands for collaboration and efficient information exchange without compromising model performance and quality.

Interoperability is a multidimensional concept encompassing organizational, technical and semantic levels [11]. This work focuses on the semantic aspect, more specifically on the exchange of data and models in cyber-physical intelligent systems. Ensuring that information is consistently interpreted and usable across platforms pertains to data exchange, while the ability to transfer and execute models in heterogeneous environments relates to model integration. Data-related aspects are typically evaluated through metrics such as data quality (conversion success, error rates, availability), similarity (semantic and structural alignment), and effectiveness (completeness and missing values). The model-related aspects are assessed based on computational cost, model performance, and preservation of original characteristics. Together, these dimensions provide a comprehensive basis for analyzing interoperability in intelligent systems.

B. Asset Administration Shell

The AAS is the standardized digital representation of an asset in the context of Industry 4.0. It provides a minimal, unique, and sufficient asset description from different perspectives relevant to each use case [12]. The AAS represents an asset's digital and standardized form in the context of I4.0, which in turn ensures interoperability between components [13]. An asset, physical or logical, is described through a set of standardized elements organized in a rigid, hierarchical structure. The combination of an asset and its AAS gives rise to the I4.0C [14], which consists of the combination of the real and digital worlds. Combining these two elements enables services and functionalities to be offered on the standardized network for interaction between I4.0Cs.

The AAS elements are divided into two main classes: Identifiable and Referable [13]. Identifiable elements have a globally unique identifier and may include proprietary domain-specific identifiers. Key subclasses include the AAS itself, the Asset it describes, and the Submodel, which allows an asset to be represented from different functional or domain-specific perspectives. Submodels are composed of Submodel Elements, which form a broad superclass encompassing other classes such as Data Elements and Property. The "Property" subclass plays a central role in this work, as it defines asset characteristics (e.g., model parameters, hyperparameters, data fields) using standardized formats such as those from the IEC 61360 Common Data Dictionary (CDD) and eCl@ss [12], [15], [16].

This structure ensures that both data and models can be represented consistently across assets, enabling semantic interoperability. By anchoring descriptions in standard repositories, the AAS minimizes ambiguity and facilitates reliable automation and analytics across diverse industrial contexts.

III. Related Works

Interoperability in I4.0 intelligent systems has been widely explored. Prior works have addressed integration at various levels, often focusing on data and model exchange. Among these, some proposals stand out for their alignment with the principles and goals of this work.

Bousdekis and Gregoris (2021) [17] propose an architectural framework for big data-driven processes in Industry 4.0, grounded in the principles of RAMI 4.0 and the AAS. The framework addresses interoperability across technical, semantic, and organizational layers, with an emphasis on integrating predictive analytics within the scope of a single asset. Semantic interoperability is supported through structured Submodels and standardized data representations that promote consistency across layers. Although based on similar architectural principles, the approach presented in the current study differs in both scope and application. The focus here is on coordinating multiple interoperable I4.0Cs within intelligent systems environments. In contrast, the framework in [17] concentrates on intra-asset interoperability without evaluating interaction among distinct I4.0Cs. The contribution of the present study lies in demonstrating cross-asset interoperability.

Sun et al. (2020) [18] propose an architecture that extends the RAMI 4.0 by integrating Distributed Ledger Technology and semantic modeling to address challenges related to data transparency, privacy, and interoperability. While it presents relevant contributions, particularly in securing data ownership and promoting information sharing among stakeholders, it overlooks key developments already present in the I4.0 ecosystem. Specifically, the concerns raised about RAMI 4.0's ability to support semantic interoperability do not take into account the role of the AAS, which is designed to standardize the semantic representation of industrial assets through structured submodels and established data dictionaries. It is important to recognize that RAMI 4.0, as a reference architecture, is intentionally abstract and extensible; it does not prescribe implementation constraints but instead provides a structured foundation upon which interoperable solutions can be built. Its ongoing evolution naturally leaves room for enhancements, but many of the identified challenges may stem not from limitations of the model itself, but from incomplete or inconsistent application of its principles.

Relevant contributions are presented by [19] and [20] , which offer a comprehensive analysis of existing approaches to semantic interoperability CPS within the context of I4.0. The authors emphasize the central role of ontologies in establishing a shared understanding of data and enabling communication and integration across heterogeneous systems. These papers also propose that combining ML techniques with semantic models can enhance interoperability in CPS architectures (even those based on RAMI 4.0 principles). While these perspectives are valuable, a noteworthy limitation lies in the fact that the intelligent system itself is expected to act as an enabler of interoperability while remaining vulnerable to the very interoperability challenges it aims to mitigate. This dual role raises important concerns about the robustness and generalization of such solutions in complex, distributed industrial environments.

Finally, [5] explore the use of the AAS to model and implement production scheduling agents within multi-agent systems for decentralized optimization. The proposed architecture leverages the collaborative capabilities of agents to solve complex problems beyond their individual scope, with a particular emphasis on the integration of intelligent agents powered by machine ML and DL techniques. While the conceptual framework presents a promising approach to distributed intelligence in I4.0 environments, it lacks an empirical evaluation of interoperability-related metrics. The actual effectiveness of the proposed architecture in supporting interoperability across heterogeneous systems remains unverified.

In light of the reviewed work, this paper aims to address important gaps that remain open in the current literature. Although previous studies offer valuable conceptual frameworks and emphasize semantic and organizational aspects of interoperability, they often lack empirical validation in multi-agent or distributed contexts. This work builds on the foundations of I4.0 to propose and evaluate a concrete architecture that enables data and model interoperability across multiple interoperable I4.0Cs.

IV. Materials and Methods

The proposed methodology aims to evaluate interoperability in I4.0 environments by implementing and comparing two machine learning scenarios: one focused on model interoperability through FL, and the other on data interoperability using a centralized learning approach. Both scenarios are grounded in the principles and architectural layers of RAMI 4.0 and make use of standardized communication protocols and AAS to ensure strict compliance with the I4.0 requirements.

To evaluate the proposed methodology, a realistic yet synthetic dataset for predictive maintenance was employed. The dataset, extracted from a publicly available research work [21], was specifically designed to reflect real-world industrial conditions while remaining accessible to the research community. It comprises 10,000 samples, each containing six features: product ID, air temperature, process temperature, rotational speed, torque, and tool wear. In addition, each data point includes a binary "machine failure" label indicating whether a failure occurred due to one or more of five independent failure modes: tool wear failure, heat dissipation failure, power failure, overstrain failure, or random failure. These failure modes are defined based on realistic operational thresholds and interactions among the recorded variables, though the specific cause of failure is not disclosed in the label, aligning with real-world conditions.

In both the centralized and federated scenarios, logistic regression was selected as the classification technique. The model was configured with a set of hyperparameters to optimize performance. The primary objective of the logistic regression model was to predict whether a machine failure would occur, without distinguishing between the different types of failure modes. This binary classification approach simplifies the prediction task to detecting failure presence or absence.

A. Federated Approach

To demonstrate model interoperability, a federated cyber-physical learning system was developed. This system comprises two main I4.0Cs: multiple federated clients and a central server. In each training round, the client is responsible for developing and evaluating a local model, then sharing both the model weights and evaluation metrics with the server. The server, in turn, initializes a global model and, in subsequent rounds, aggregates the model weights received from all clients, calculates global model evaluation metrics, and distributes the updated global model back to the clients. This process enables iterative training and aggregation across the system.

Both the client and the server are virtually and consistently represented by their respective AAS which are shown in Figures 1 and 2, respectively. In the simulated scenario, the clients correspond to industrial equipment enhanced with functionalities related to new communication and information technologies. In addition to technical specifications related to operational functions, the client's AAS contains local model information such as variables and hyperparameters, and server connection details. The AAS of the server includes specifications for global model variables and hyperparameters, aggregation and client selection strategies, and connection details for interacting with clients.

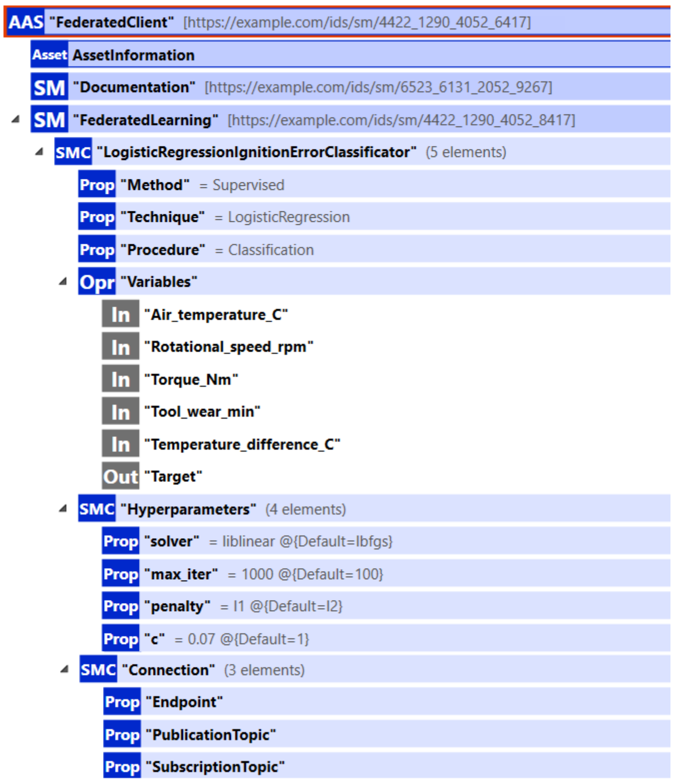

Fig. 1. Federated client AAS.

To align with the I4.0 requirements, specifications defined in the RAMI 4.0 axis layers were adopted. Interaction between clients and the server was implemented using MQTT, while the integration between each asset and its AAS utilized HTTP. These protocols correspond to the Communication and Functional Layers of RAMI 4.0, respectively, ensuring strict adherence to I4.0 standards. It is important to note that this work does not detail the internal structure of the I4.0Cs according to the layers and elements defined by RAMI 4.0. Instead, the system's dynamics are described at a higher level of abstraction as shown 3 and 6, focusing on the processes and interactions among components rather than their internal architectural implementation .

To simulate a realistic scenario, the client and server assets were implemented in different programming languages and without the use of a FL framework. The client asset was developed in Kotlin with ”Smile” as the ML library, while the federated server was implemented in Python using the ”scikit-learn” framework. Model interoperability between clients and server was evaluated based on classification performance metrics and computational cost.

Fig. 2. Federated server AAS.

The system operates dynamically based on a standardized initialization and communication workflow, represented in Figure 3. Upon instantiation, both clients and the server retrieve their operational parameters by sending an HTTP request to the API provided by the RAMI 4.0 Functional Layer. The parameters received from their respective AAS are then used to configure and initialize the assets. Once initialized, clients and server are ready to communicate using the MQTT protocol. Their interaction follows an iterative process consisting of local model training, evaluation, and sharing by the clients, followed by global model aggregation, evaluation, and redistribution by the server. This cycle continues until a predefined stopping criterion is met.

B. Centralized Approach

In this centralized learning scenario, the focus shifts from model interoperability to data interoperability. Instead of federated clients and a server, the system comprises industrial equipment enhanced with information and communication-related functionalities and a data consumer module (DCM), both represented as I4.0Cs. The industrial equipment is respon-

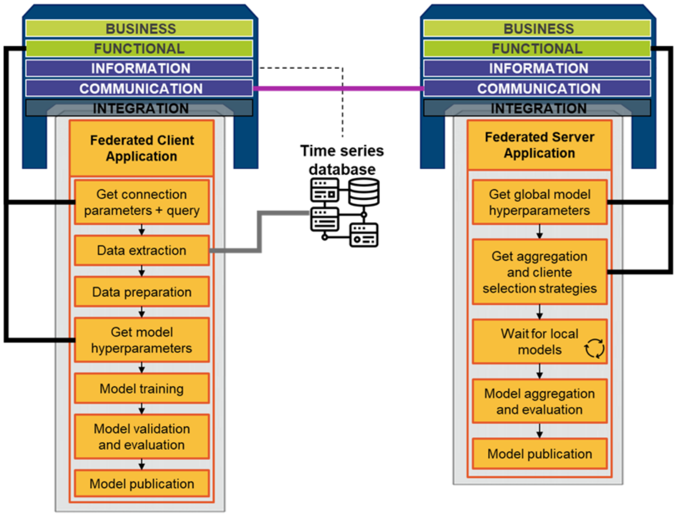

Fig. 3. Federated Cyber-Physical Learning System's Dynamics.

sible for generating operational data, while the DCM handles data extraction, preparation, and the development of a ML classification model. Once trained, the model is made available for prediction through an API.

Each AAS of the industrial equipment encapsulates not only its operational and technical specifications but also includes a submodel for time series data (TimeSeries Standardized Submodel) that defines how the equipment's data is structured and accessed. This specific Submodel is represented in Figure 4. Meanwhile, the DCM's AAS shown in Figure 5 contains detailed information about the machine learning model's parameters and variables, facilitating model configuration and management within the centralized learning framework.

Fig. 4. Industrial equipment AAS.

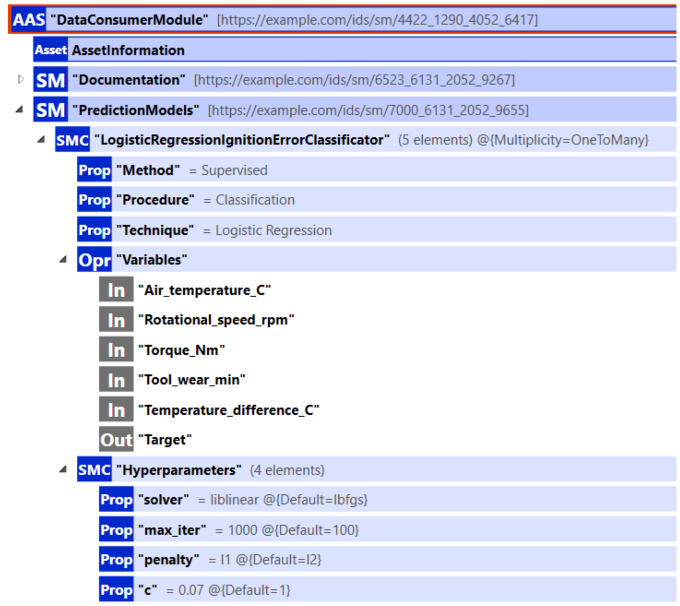

Fig. 5. Data Consumer Module AAS.

Both components were implemented in Python, ensuring a uniform technology stack. The system follows a centralized ML architecture in which time series data are extracted by a DCM, processed, and used to train an ML model. The overall process is structured as follows:

Upon instantiation, the DCM accesses its AAS through the Functional Layer and retrieves the model configuration parameters, including the relevant variables and hyperparameters. These parameters, defined in the AAS, enable the creation and proper configuration of the ML model. Subsequently, the DCM accesses the AAS of the asset to which the time series data is associated, again through the Functional Layer, and obtains the data access parameters (e.g., endpoint and query). With this information, the DCM extracts the data, performs the necessary preprocessing steps, and trains the ML model. Once trained, the model is made available via an API, allowing it to be used for inference. For example, the model can be employed in real time to make predictions based on new operational parameter readings from the asset that generates the data. This dynamics is schematically represented in Figure 6.

The evaluation process employed in the centralized learning scenario was aligned with the federated setup to enable meaningful comparison. Standard classification performance metrics were used to assess the ML model. Additionally, to evaluate data interoperability, specific metrics were introduced, including data quality and data transmission effectiveness. These metrics aimed to assess the consistency, compatibility, and utility of the data extracted the source. Together, these metrics provided a comprehensive view of the system's ability to support interoperable data-driven applications in line with I4.0 principles.

V. Results and Discussion

This section presents the results obtained from implementing the proposed methodology and provides a discussion on

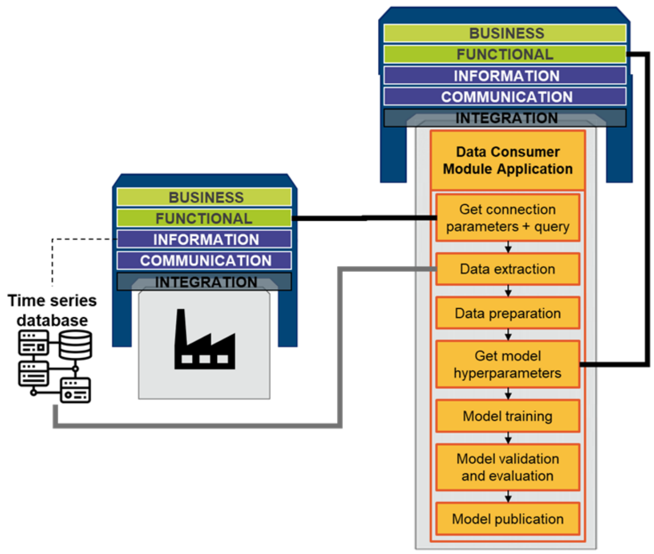

Fig. 6. Centralized Cyber-Physical Learning System's Dynamics.

their implications.

To analyze interoperability in the exchange of models between client and server, three criteria were used: in terms of preservation of original characteristics of the models, the hyperparameters that can be configured in each framework were considered. Regarding model performance, accuracy and recall were used. Finally, regarding computational cost, the time, memory usage, and CPU steps in the training and aggregation stages of the models were evaluated.

The frameworks used in the client and federated server, Smile (Kotlin) and Scikit-learn (Python) respectively, offer interfaces for creating logistic regression models with important differences in the available hyperparameters, reflecting differences in design and flexibility. Table I presents a comparison of the frameworks in terms of hyperparameters:

TABLE I
HYPERPARAMETER COMPARISON BETWEEN SMILE AND SCIKIT-LEARN FOR LOGISTIC REGRESSION

Hyperparameter	Smile	Scikit-learn
Regularization	`lambda`	`penalty`, `C`, `l1_ratio`
Convergence tolerance	`tol`	`tol`
Maximum iterations	`maxIter`	`max_iter`
Intercept	Implicit	`fit_intercept`, `intercept_scaling`
Solver	Not available	`solver`
Class weight	Not available	`class_weight`
Randomness	Not available	`random_state`
Incremental training	Not available	`warm_start`

As shown in Figures 1 and 5, in the logistic regression model, the penalty is set to 'l1' to allow feature selection, while C = 0.07 applies strong regularization to avoid overfitting. The solver adopted was liblinear because of its compatibility with L1 regularization, which promotes sparsity by reducing some coefficients to zero. The maximum number of iterations is increased to 1000 to ensure convergence, given the complexity added by the L1 penalty.

Although model configurations are defined in the AAS, differences between the ML frameworks hinder full model interoperability, as stated by [22], [23] in previous works. In particular, the Smile framework lacks several key hyperparameters, limiting model flexibility and performance. The absence of penalty options restricts the use of sparsity-promoting strategies, important for feature selection and overfitting control in high-dimensional data. The inability to choose a solver forces reliance on a fixed optimization method, which may not suit all data characteristics, affecting fit quality and convergence. Additionally, Smile does not support class weighting, which impairs performance on imbalanced datasets, nor does it offer parameters for result reproducibility. The lack of a warm start mechanism further limits incremental training—an essential feature in federated learning to reduce computational cost when models are updated frequently.

To evaluate interoperability in data exchange between the time series data source and the DCM, alongside model performance and computational cost, data quality was assessed using both univariate and bivariate statistical analyses. Specifically, the mean, standard deviation, and Pearson correlation coefficient with the class variable were calculated for each attribute, both at the data source and after extraction by the DCM. Tables II and III summarize these results for the attributes effectively used in model training.

TABLE II
STATISTICAL SUMMARY OF ATTRIBUTES FROM DCM AND TIME SERIES DATA SOURCES

Attribute	DCM		Time Series Data Source	
	Mean	Std. Dev.	Mean	Std. Dev.
Air temp. (°C)	27.8549	2.0003	27.8549	2.0002
Process Temp. (°C)	37,8555	3,7894	37,8555	3,7893
Rotational speed (RPM)	1538.7761	179.2841	1538.7761	179.2751
Torque (Nm)	39.9869	9.9689	39.9869	9.9684
Tool wear (min)	107.9510	63.6541	107.9510	63.6510

TABLE III
PEARSON CORRELATION COEFFICIENT WITH THE CLASS VARIABLE

Attribute	DCM	Time Series Data Source
Air Temperature (°C)	0.08256	0.08260
Process Temperature (°C)	0.03589	0.03590
Rotational Speed (RPM)	-0.04419	-0.04420
Torque (Nm)	0.19132	0.19130
Tool Wear (min)	0.10545	0.10540

The results show no significant differences between the statistical metrics calculated directly at the time series data source and those obtained after data extraction by the DCM. The observed discrepancies are minimal, limited to the fourth decimal place, and can be attributed to differences in the computational kernels used during the calculations SQL-based scripts were employed at the source, while Python-based computations were applied at the destination. These negligible variations support the conclusion that data quality is preserved during the exchange process, reinforcing the interoperability of the proposed solution.

An important dimension of data interoperability in I4.0 is semantic interoperability, which refers to the ability of systems not only to exchange data but also to interpret it consistently and meaningfully, a facet that was qualitatively analyzed in this study. This is particularly critical when integrating assets from different vendors or domains. In this context, standardized data dictionaries such as the IEC CDD and eCl@ss play a pivotal role. These repositories provide unified definitions and identifiers for properties and elements used within AAS, ensuring that all components refer to data attributes with shared semantics. Anchoring data descriptors in such standardized vocabularies enhances cross-system understanding, reduces ambiguity in data interpretation, and supports reliable automation and analytics across diverse industrial assets.

Table IV presents a comparative analysis between the centralized and federated approaches based on several performance and resource-related metrics. Given the nature of the problem, fault prediction in industrial assets is the most critical requirement, and it is necessary to correctly identify actual failure events. In this context, false negatives (i.e., real failures that go undetected) are significantly more detrimental than false positives. Therefore, recall is the most appropriate metric for evaluation. In addition to model performance, training time, CPU usage time, and memory consumption were considered to evaluate computational efficiency. For the FL scenario, all metrics were collected only from the final training round out of a total of 10 communication rounds. Additionally, metrics associated with the federated clients represent average values computed across all participating clients.

TABLE IV
PERFORMANCE COMPARISON: CENTRALIZED VS. FEDERATED LEARNING

Metric	**Centralized**	**Federated Client (Avg.)**	**Federated Server**
Accuracy (%)	97.45	81.78	85.92
Recall (%)	97.08	82.54	84.28
Training Time (s)	7.5837	0.5734	2.17
CPU Usage Time (s)	0.5001	0.4375	0.86
Memory Consumption (MiB)	9.3125	0.0039	1.66

The centralized model achieved the best performance in terms of evaluation metrics, demonstrating that high-quality models can be obtained while preserving data interoperability. In the federated setting, although model performance was inferior to that of the centralized approach, both clients and the federated server maintained models with similar metric values. This is consistent with the characteristics of FL, where the global model is evaluated through a weighted average based on client contributions rather than a simple arithmetic mean. These results indicate that, despite differences in implementation due to framework constraints, a reasonable level of model interoperability can still be achieved in federated environments.

From a computational cost perspective, the centralized model requires significantly more resources as it processes a larger volume of data during training. Federated clients, on the other hand, operate on local subsets of the data and therefore demand fewer computational resources. The federated server also incurs lower computational cost in comparison to the centralized approach, as its role is limited to aggregating local models rather than training them directly.

A brief comparison between the results of this study and those reported in the literature is presented. [24] provide an extensive review of studies that used the same dataset for binary classification, employing a variety of machine learning techniques. In addition to their own results, evaluation metrics from 16 other publications are available for comparison. Among these, 13 achieved accuracy levels above 95% using different classification methods. However, only five of them also maintained a high recall (greater than 80%). Notably, the review in [24] highlights that most prior works prioritized precision over recall, an approach that may be unsuitable for failure prognosis, where capturing all relevant failures (high recall) is critical. This brief analysis positions the present work among the top-performing studies in terms of both accuracy and recall.

VI. CONCLUSIONS

This work proposed and evaluated interoperability aspects of intelligent systems in the context of in I4.0. By leveraging the AAS and adhering to RAMI 4.0 principles, the proposed scenarios enables seamless integration of assets and ensures interoperability in both data and model dimensions.

The centralized approach demonstrated superior predictive performance, benefiting from complete data access and uniform implementation. However, the federated scenario, despite limitations in the ML frameworks used, achieved high-quality results and highlighted the feasibility of model interoperability even when using heterogeneous tools and environments with the advantage of preserving data privacy. In particular, federated clients operated with significantly lower computational costs, making this approach advantageous for scenarios where data locality or privacy is critical. Importantly, this study also revealed that technology-related factors, such as the capabilities and limitations of frameworks, can impose constraints on interoperability. Nevertheless, both scenarios could still preserve key interoperability metrics.

The results confirm that AAS can serve as a key enabler for interoperability in intelligent systems, supporting both model exchange and data standardization. Moreover, this study shows that even simple ML models, like logistic regression, can benefit from a well-structured interoperability architecture when properly integrated into I4.0 components.

One of the key challenges in evaluating interoperability in intelligent systems is isolating its specific influence from other confounding factors that may affect system behavior or performance metrics. Given the complexity and interdependence of components in such systems, it remains difficult to determine the extent to which observed outcomes can be directly attributed to interoperability itself. Addressing this challenge represents an important direction for future research, particularly in the development of evaluation frameworks capable of distinguishing interoperability's impact from that of other system variables.

Acknowledgment

The authors thank the FapUnifesp - Fundação de Apoio à Universidade Federal de São Paulo, the Brazilian governmental agencies, the scientific initiation program of the Faculdade Impacta de Tecnologia, the National Council for Scientific and Technological Development (CNPq), the Coordination for the Improvement of Higher Education Personnel (CAPES), and São Paulo Research Foundation (FAPESP) – grant nº 2020/09850-0, project title: "Center for Applied Research in Artificial Intelligence for the evolution of Industries to Standard 4.0", IAsmin Platform.

References

[1] R. S. Nakayama, M. de Mesquita Spínola, and J. R. Silva, "Towards i4. 0: A comprehensive analysis of evolution from i3. 0," *Computers & industrial engineering*, vol. 144, p. 106453, 2020.

[2] H. Kagermann, W. Wahlster, and J. Helbig, "Securing the future of german manufacturing industry: Recommendations for implementing the strategic initiative industrie 4.0," *Final report of the Industrie*, vol. 4, no. 0, 2013.

[3] H. Demirkan, S. Earley, and R. R. Harmon, "Cognitive computing," *IT professional*, vol. 19, no. 4, pp. 16–20, 2017.

[4] H. Kagermann and W. Wahlster, "Ten years of industrie 4.0," *Sci*, vol. 4, no. 3, p. 26, 2022.

[5] V. Siatras, E. Bakopoulos, P. Mavrothalassitis, N. Nikolakis, and K. Alexopoulos, "On the use of asset administration shell for modeling and deploying production scheduling agents within a multi-agent system," *Applied Sciences*, vol. 13, no. 17, p. 9540, 2023.

[6] Z. Gao, T. Wanyama, I. Singh, A. Gadhrri, and R. Schmidt, "From industry 4.0 to robotics 4.0-a conceptual framework for collaborative and intelligent robotic systems," *Procedia manufacturing*, vol. 46, pp. 591–599, 2020.

[7] H. Cañas, J. Mula, F. Campuzano-Bolarín, and R. Poler, "A conceptual framework for smart production planning and control in industry 4.0," *Computers & Industrial Engineering*, vol. 173, p. 108659, 2022.

[8] C. Salkin, M. Oner, A. Ustundag, and E. Cevikcan, "A conceptual framework for industry 4.0," in *Industry 4.0: managing the digital transformation*, pp. 3–23, Springer, 2017.

[9] S. Agrawal, A. Sahu, and G. Kumar, "A conceptual framework for the implementation of industry 4.0 in legal informatics," *Sustainable Computing: Informatics and Systems*, vol. 33, p. 100650, 2022.

[10] S. Purushothaman, G. S. Shanmugam, and S. Nagarajan, "Achieving seamless semantic interoperability and enhancing text embedding in healthcare iot: A deep learning approach with survey," *SN Computer Science*, vol. 5, no. 1, p. 99, 2023.

[11] I. Framework, "European interoperability framework for pan-european egovernment services," *2004*, 2004.

[12] H. Bedenbender, M. Billmann, U. Epple, T. Hadlich, M. Hankel, R. Heidel, O. Hillermeier, M. Hoffmeister, H. Huhle, M. Jochem, *et al.*, "Examples of the asset administration shell for industrie 4.0 components–basic part," *ZVEI white paper*, pp. 4–9, 2017.

[13] S. Bader, E. Barnstedt, H. Bedenbender, B. Berres, M. Billmann, and M. Ristin, "Details of the asset administration shell-part 1: the exchange of information between partners in the value chain of industrie 4.0 (version 3.0 rc02)," 2022.

[14] H. Bedenbender, A. Bentkus, U. Epple, T. Hadlich, M. Hankel, R. Heidel, and M. Woolschlaeger, "Relationships between i4. 0 components–composite components and smart production," *Federal Ministry for Economic Affairs and Energy (BMWi), Berlin*, 2017.

[15] P. Adolphs, H. Bedenbender, D. Dirzus, M. Ehlich, U. Epple, M. Hankel, R. Heidel, M. Hoffmeister, H. Huhle, B. Kärcher, *et al.*, "Reference architecture model industrie 4.0 (rami4. 0)," *ZVEI and VDI, Status report*, 2015.

[16] X. Ye and S. H. Hong, "Toward industry 4.0 components: Insights into and implementation of asset administration shells," *IEEE Industrial Electronics Magazine*, vol. 13, no. 1, pp. 13–25, 2019.

[17] A. Bousdekis and G. Mentzas, "Enterprise integration and interoperability for big data-driven processes in the frame of industry 4.0," *Frontiers in big Data*, vol. 4, p. 644651, 2021.

[18] S. Sun, X. Zheng, J. Villalba-Díez, and J. Ordieres-Meré, "Data handling in industry 4.0: Interoperability based on distributed ledger technology," *Sensors*, vol. 20, no. 11, p. 3046, 2020.

[19] M. M. Hafidi, M. Djezzar, M. Hemam, F. Z. Amara, and M. Maimour, "Semantic web and machine learning techniques addressing semantic interoperability in industry 4.0," *International Journal of Web Information Systems*, vol. 19, no. 3/4, pp. 157–172, 2023.

[20] J. Nilsson, S. Javed, K. Albertsson, J. Delsing, M. Liwicki, and F. Sandin, "Ai concepts for system of systems dynamic interoperability," *Sensors*, vol. 24, no. 9, p. 2921, 2024.

[21] S. Matzka, "Explainable artificial intelligence for predictive maintenance applications," in *2020 third international conference on artificial intelligence for industries (ai4i)*, pp. 69–74, IEEE, 2020.

[22] Y. Liu, C. Chen, R. Zhang, T. Qin, X. Ji, H. Lin, and M. Yang, "Enhancing the interoperability between deep learning frameworks by model conversion," in *Proceedings of the 28th ACM joint meeting on European software engineering conference and symposium on the foundations of software engineering*, pp. 1320–1330, 2020.

[23] Y. K. Lee, S. H. Park, M. Y. Lim, S.-H. Lee, and J. Jeong, "Towards ensuring software interoperability between deep learning frameworks," *Journal of Artificial Intelligence and Soft Computing Research*, vol. 13, no. 4, pp. 215–228, 2023.

[24] B. Ghasemkhani, O. Aktas, and D. Birant, "Balanced k-star: An explainable machine learning method for internet-of-things-enabled predictive maintenance in manufacturing," *Machines*, vol. 11, no. 3, p. 322, 2023.