Multi-Agent Systems to Implement Industry 4.0 Components

Lucas Sakurada* and Paulo Leitão*
* Research Centre in Digitalization and Intelligent Robotics (CeDRI), Instituto Politécnico de Bragança,
Campus de Santa Apolónia, 5300-253 Bragança, Portugal
Email: {lsakurada, pleitao}@ipb.pt

Abstract—**The fast-changing market conditions, the increased global competition and the rapid technological developments demand flexible, adaptable and reconfigurable manufacturing systems based on Cyber-Physical Systems (CPS). Aligned with CPS, the adoption of production system architectures is suitable to reduce complexity and achieve interoperability in the industrial applications. In this context, the Reference Architecture Model for Industry 4.0 (RAMI4.0) provides the guidelines to develop Industry 4.0 (I4.0) compliant solutions, considering the existing industrial standards. The so-called I4.0 components implement this model in practice, combining the physical asset with its digital representation, named Asset Administration Shell (AAS). This paper explores the use of Multi-Agent Systems (MAS) to implement the AAS functionalities, taking advantage of their inherits characteristics, e.g., autonomy, intelligence, decentralization and reconfigurability. In this context, the mapping between AAS functionalities and MAS characteristics is provided, as well as the challenges for this implementation. The applicability is illustrated by digitalizing an inspection cell comprising an UR3 robot and several console products by using MAS technology.**

Keywords: Asset Administration Shell, I4.0 component, Multi-Agent Systems, RAMI4.0.

I. INTRODUCTION

The fast-changing market conditions, the increased global competition and the rapid technological developments, drastically increase the complexity of production systems. In this context, the adoption of new production system architectures is a key enabler to reduce complexity and achieve interoperability in the industry [1]. The digital transformation, associated to Industry 4.0 (I4.0), promotes the transition of traditional manufacturing systems towards modern and smart factories, which are more flexible, adaptable and reconfigurable, bringing new opportunities and innovative solutions for modern manufacturing systems [2], [3].

This transformation is based on Cyber-Physical Systems (CPS) that act as backbone to implement Industry 4.0, integrating physical processes with computational resources in the same network [4]. CPS are combined with emergent technologies, namely the Internet of Things (IoT), Artificial Intelligence (AI), cloud computing, virtual reality and collaborative robotics, to develop large-scale systems, performing tasks in a decentralized and intelligent manner.

The Reference Architecture Model for I4.0 (RAMI4.0) [5] is a production system architecture that provides guidelines for implementing solutions based on the I4.0 considering existing standards. As a reference model, the RAMI4.0 specifies the I4.0 components that combine the assets with their digital representation, called Asset Administration Shell (AAS), which allow the access and control of the asset information and provides an interface communication with other I4.0 components based on the service-oriented architecture [2], [5].

The implementation of AAS can be performed by using some industrial technologies, such as Automation Markup Language (AutomationML) and Open Platform Communications Unified Architecture (OPC UA), which ensure its compliance according to the standards [6]. Although the use of these technologies allow the transparent communication and lossless data exchange between heterogeneous engineering tools, e.g., robots and Programmable Logic Controllers (PLCs) [7], they do not provide intelligent capabilities, which are not required capabilities defined for the AAS.

In this context, Multi-Agent System (MAS) [8] is a suitable approach to embedded intelligence in the system, providing decentralized solutions through interaction among autonomous, proactive and cooperative entities, namely software agents. Aligned with the I4.0 principles, industrial agent-based solutions are a promising approach to develop I4.0 components, using MAS to implement the key functionalities of AAS, such as providing a digital representation of the asset, collecting data and performing the communication with other I4.0 components [1]. MAS can also support the introduction of intelligence in AAS, exploring a new dimension in I4.0 components.

Having this in mind, this paper explores the use of MAS technology to implement the AAS functionalities, considering its inherent characteristics, and analysis the key challenges on how to digitalize assets based on MAS. The proposed approach is illustrated through the digitalization of an UR3 collaborative robot and console display products into I4.0 components, using MAS technology to implement the AAS functionalities related to the RAMI4.0 layers.

The rest of the paper is organized as follows. Section II overviews the RAMI4.0 architecture, the AAS concept and the guidelines to digitalize physical assets according to the RAMI4.0 layers. Section III discusses the challenges of implementing AAS and Section IV presents MAS as a suitable approach to implement the AAS functionalities, describing

978-1-7281-6389-5/20/$31.00 ©2020 IEEE

how these are mapped into the MAS characteristics. Section V describes two examples of digitalizing physical asset using MAS. Finally, Section VI rounds up the paper with the conclusions and points out the future work.

II. RAMI4.0 AND AAS

This section overviews the RAMI4.0 reference architecture and the realization of I4.0 components based on the AAS.

A. *RAMI4.0 Reference Architecture*

The RAMI4.0 is a three-dimensional reference model, illustrated in Figure 1, to design I4.0 compliant solutions, providing information and flexibility across the productive chain. RAMI4.0 focuses on integrating manufacturing systems, mapping and linking different aspects of I4.0 in a common model [5].

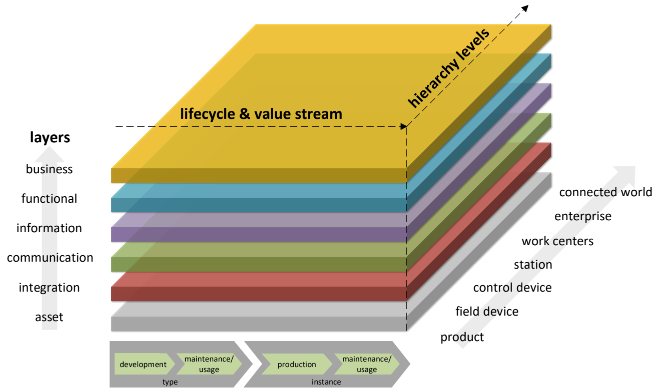

Figure 1. The Reference Architecture Model for I4.0 (adapted from [5]).

The RAMI4.0 layers focus on the product development and production scenarios, describing the configuration and function of each layer level. The *asset* level represents the physical and nonphysical objects, e.g., machines, services, orders or documents. The *integration* level represents the transition from the real to the digital world, through the conversion of data from assets, usually obtained by sensors, on compatible information to be processed in a computer. The *communication level* provides a standard communication platform between the *integration* level and the *information* level, which is responsible to structure data in a proper manner. The *functional* level describes the functionalities provided by the asset and is responsible for taking some actions based on the information obtained on the previous level. Lastly, the *business* level is responsible for the business process of the asset [5], [9].

In the I4.0 environment, products and machines present different behaviours during their lifecycle, i.e. beginning, middle and end of life, showing the distinction between data, functions and configurations. Thus, the *lifecycle* axis defines the lifecycle of a product, machine and factory based on the IEC 62890 standard. To complement the *lifecycle* axis, the *value stream* records information along the process, being possible to make corrections and improvements during the product lifecycle and in the next production of products [5], [9].

The hierarchy level follows the IEC 62264 and IEC 61512 standards, and describe the different functionalities in the automation plant, namely field device, control device, station, work center and enterprise. In order to extend these functionalities for the I4.0 context, other two levels were added, namely the product and the connected world, reflecting the importance of the product and not only the production process, and the connection with other systems through the use of IoT and Internet of Services (IoS) [5], [10].

B. *Asset Administration Shell*

In the Industry 4.0 context, the I4.0 component was developed to consolidate and apply the RAMI4.0 model in practice. As shown in Figure 2, this component combines the physical object with its digital representation, namely asset and AAS, respectively [2].

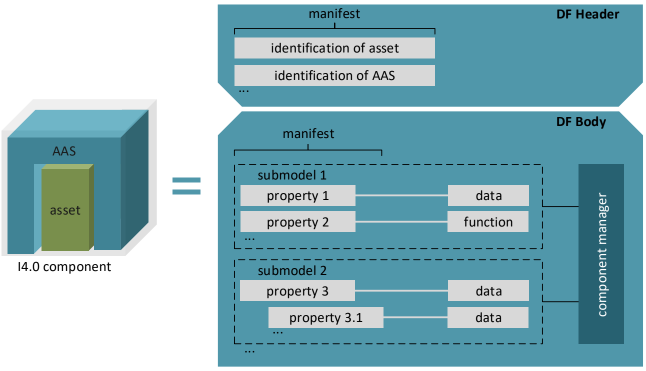

Figure 2. I4.0 component model (adapted from [11]).

The asset is a valuable industry component, not necessarily a low-level controller (e.g., PLC or robot), but also a nonphysical object, such as a service or an order [2], [3]. The AAS allows to convert a simple asset into an I4.0 component, accessing to the information, properties and data of the asset and encapsulating its functionality, facilitating the exchange of information and communication between all components of the productive chain. According to Figure 2, the AAS has a well-defined structure, namely the Digital Factory (DF) Header and the DF Body, basically composed by a manifest and a component manager. The manifest is a list of properties contained in DF Header and DF Body of the I4.0 component, e.g., the identification of the asset and the AAS. The component manager is responsible for managing the submodels contained in the DF body, which contain properties (i.e. the manifest) related to several data and functions of the asset [2], [12].

C. *Interpreting How Digitalizing Assets*

In the I4.0 context, everything can be digitalized. However, it does not mean that everything needs to be digitalized, being a mistake and a waste of time and resources if the asset does not bring any add value. The key challenge is to determine why, what and how digitalizing the physical assets. In this sense, the layers of the RAMI4.0 architecture can assist to get the answers for these questions. As illustrated in Figure 3, these layers can be divided into two groups according to the characteristics of each layer: the IoS related group that includes the business and functional layer, and the IoT related group that includes the other layers.

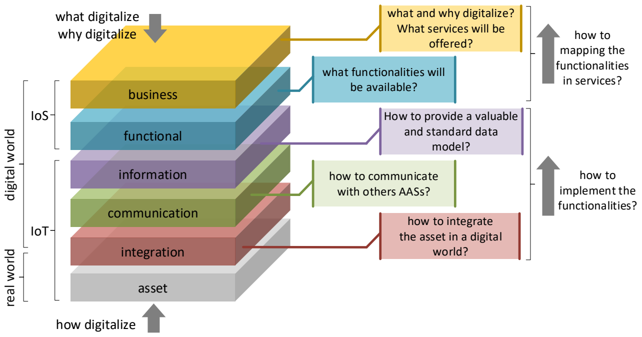

Figure 3. Procedure to digitalizing assets.

To answer the *why digitalizing* and *what to digitalize*, it is necessary to perform a top-down analysis focused in the IoS group, analysing the business, services and functionalities that will bring added value. The starting point is to verify the demands from customers to know what and why digitalizing, as well as the services that will be offered, considering the resources availability and the implementation costs.

At the moment the decision to digitalize is performed, it is necessary to study how the functionalities offered as services will be implemented. For this purpose, a bottom up analysis focusing the IoT group should be performed, starting by analyzing how to map the AAS functionalities into services. Then, it is essential to verify how to implement the AAS functionalities, considering the asset, the integration between the asset and the AAS, and how to deal with the information and communication issues. For this purpose, it is fundamental to integrate the asset in a digital world, collecting information through sensors and controllers. The access to this data should follow industrial standard communication protocols, such as OPC UA or MQTT. Lastly, for avoiding data inconsistency, it is essential to use standard data models, such as AutomationML and FIREWARE, to maintain the exchanged data in a structured and integrated manner.

III. CHALLENGES OF IMPLEMENTING AAS

This section identifies the key challenges related to implementing the AAS, namely how to support interoperability, the interface between the asset and its digital representation, how to embed and distribute intelligence and security aspects.

A. *How to Support Interoperability*

The industrial production system consists of heterogeneous engineering software applications and hardware devices, where each one has its own data model and a specific form of communication. In this case, the data and communication incompatibility causes inconsistency and inefficiency for exchange of information across the production chain. In this context, interoperability is a key challenge to be achieved, allowing the transparent communication and lossless data exchange between different assets. For this purpose, the adoption of technologies aligned with industrial standards may contribute to reach interoperability. As examples, AutomationML (standardized in IEC 62714), which provides a valuable data format for the exchange of information between the engineering tools [13], [14], and OPC UA protocol (standardized in IEC 62541) [14], defined as a platform-independent architecture for communication of devices in the industrial environment.

B. *How to Interconnect the Digital and Physical Counterparts*

The physical world is surrounded by unexpected, uncertain and inaccurate conditions, which demands concerns regarding how to design a suitable interface between the physical asset and its digital representation to ensure reliability, feasibility and stability. Moreover, real-time interaction is another crucial aspect, especially in interfaces that simulate the physical object through a digital object, e.g., the Digital Twin. Another obstacle is to develop interfaces according to the industrial requirements, that fit different kinds of assets and considers criteria for assessment.

Considering all of these factors, the design of the technological interface is not a straightforward task. In these cases, the knowledge about the application scenario requirements can assist in selecting the best interface. As an example, for applications based on MAS, particularly industrial agents, the IEEE P2660.1 working group [15] is being developing recommended practices to interconnect software agents with low-level automation functions, through a specific methodology [16] to attend the industrial scenario.

C. *How to Embed and Distribute Intelligence*

Industry takes significant steps to achieve interoperability inside and outside the factory floor, connecting with the world through IoT and IoS. RAMI4.0 is a starting point for this new reality, supported by AAS, which becomes possible to consolidate and implement this architectural model in practice. However, this model leaves a gap, since it does not refer to intelligent capabilities. In this context, embedding AI and Machine Learning (ML) algorithms in AAS is a crucial challenge, e.g., to support the analysis of collected data aiming monitoring, diagnosis, prediction and optimization.

Another issue is the distribution of intelligence among edge, fog and cloud computing layers. Each layer presents differences in terms of functional, data analysis, technological and implementation aspects, as pointed at [17]. The edge layer is embedded near the data sources, which makes possible the real-time monitoring and control, however, presents limitations for data storage and processing. The fog layer is close to data sources, working between the edge and the cloud computational layers. This layer allows reducing the size and increasing the quality of data sent to the cloud in a distributed manner. Lastly, the cloud layer provides high quality to analyze and process a significant amount of data, but presents a certain level of latency for being away from the data source [17], [18]. In this way, the selection of the proper AI algorithms and the suitable layer to execute these algorithms are a crucial issue to achieve high performance in I4.0 environments, considering where and how to deploy these computational resources.

D. Security

I4.0 is closely linked to the concept of CPS, which integrates ICT technologies in the same network. The use of IoT technologies to ensure the connectivity of industrial devices over the Internet, brings several benefits but also presents several problems. In fact, the data is not anymore restricted, becoming more vulnerable to cyber-attacks [19], [20]. According to [21], cybersecurity is one of the main challenges faced by companies around the world, showing an investment of more than 500 million dollars in cybersecurity and a loss of 400 billion dollars a year.

In this context, security aspects are indispensable to protect the data, ensuring privacy, integrity and availability. This requires the implementation of simple mechanisms, e.g., authentication and encryption, but also more complex ones based on the use of ML techniques, e.g., to detect possible threats. Besides the protection of data, another fundamental aspect is to ensure safety for possible physical damage involving humans or other machines [19].

IV. MAS AS AN ENABLER TO IMPLEMENT AAS

Derived from the area of distributed AI, MAS [8] is based on a society of intelligent, cooperative, proactive and autonomous entities, called agents, that represent physical or logical objects in the system. The agents are distributed over the environment, interacting with each other and the environment, in order to reach a particular or a common goal, exchanging information, making decisions and adapting their behaviour according to condition changes [8], [22].

In a preliminary comparative analysis between MAS and AAS architectures, Figure 4 shows that the MAS technology presents the fundamental requirements to implement AAS. Both present the ability to represent and interact with physical assets, as well as communicate, access and exchange information with other agents or AASs. Moreover, each agent may encapsulate the AAS structure, comprising the manifest and the component manager.

In a detailed perspective, some of the MAS characteristics, such as reusability, responsiveness, social ability, learning ability, reasoning, proactivity, cooperation and autonomy [8], [22], can be leveraged to implement the AAS functionalities associated with the RAMI4.0 layers. In this sense, Table I presents how the MAS characteristics are mapped to implement the AAS functionalities. MAS can represent the assets through its ability to represent physical objects by using autonomous and cooperative agents that use communication protocols to assist the integration with the asset. In these systems, the overall system emerges from the iteration among the agents (each one representing different assets) by using cooperation patterns following the standardized FIPA-ACL communication language. Moreover, MAS structure the collected data using proper data models and embed artificial intelligence algorithms to support monitoring, diagnosis, prediction and optimization. The functionalities of each agent are encapsulated and offered as services to other agents.

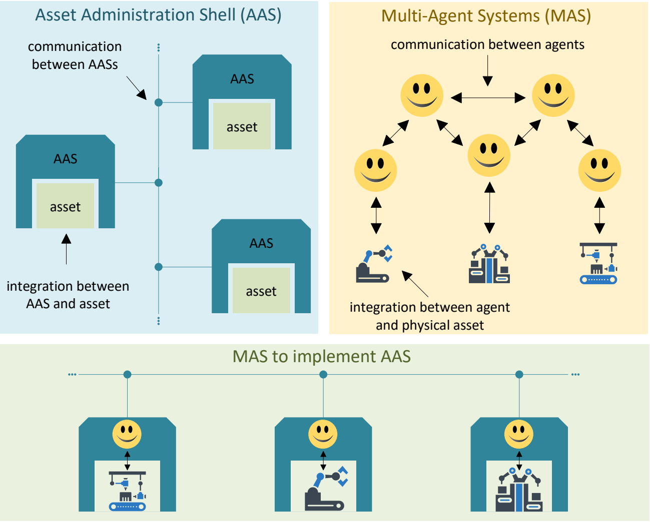

Figure 4. Multi-Agent Systems to implement AAS.

The MAS solutions have been gaining industrial relevance in the last years. As examples, several innovative European research projects, such as GRACE and GO0D MAN, have demonstrated the benefits of introducing agents to improve and solve problems in production systems. GRACE focuses on integrating the process and quality control taking advantage of flexibility, on-the-fly reconfigurability and self-adaptability that MAS provide [23]. GO0D MAN develops an approach based on MAS for zero-defect manufacturing in multi-stage production systems, through the distribution of intelligence by agents, allowing the early detection of defects and the implementation of self-adjustments [24]. Both projects showed consistent results in accordance to industrial requirements [23], [24]. Although these projects are not directly applied in the implementation of AAS, they show the feasibility and potential application of MAS in industrial environments.

V. EXAMPLES OF DIGITALIZING ASSETS USING MAS

This section presents two examples of digitalizing two distinct assets for an inspection station that checks console displays. Both examples are developed using the JADE framework and comprise the ground for implementing AAS using MAS, covering all RAMI4.0 layers. JADE is a proper platform for facilitating the development of MAS solutions, providing mechanisms for communication between agents, yellow pages services, and debug tools [22]. Figure 5 illustrates the architecture of the digitalized inspection station, where there is an agent responsible to manage the entire lifecycle of each asset.

Table I
MAPPING MULTI-AGENT SYSTEM IN AAS FUNCTIONALITIES

RAMI4.0 Layers	AAS Functionalities	MAS Characteristics	MAS Support Functionalities
Asset	Provide a digital representation of the asset.	Representative entity: the agent represents the physical or logical objects in the system. Reusability: the agent is a software component that can be reused, taking advantage of its structure and specific functionalities.	JADE framework to develop the MAS-based solution compliant with Foundation for Intelligent Physical Agents (FIPA).
Integration	Establish a connection between the asset and its digital representation.	Responsiveness: the agent can establish a bridge between the real and digital world, perceiving the real-world through sensors and respond by means of actuators.	Integration with asset through communication protocols: MQTT, OPC UA, Modbus and Ethernet/IP. Recommended practices by IEEE P2660.1.
Communication	Access the asset information and establish the communication with the other assets.	Social ability: the agent can interact with humans, assets or other agents to achieve their goals, providing or accessing information.	Send or receive FIPA-ACL messages between agents. Client-Server or Publish-Subscribe approach.
Information	Provide data in a structured and integrated manner.	Learning and social ability: the agent has the ability to learn and provide the knowledge learned to humans, assets or other agents.	Combine MAS with data models, such as FIREWARE and AutomationML, to support the exchange of data in a structured and integrated manner.
Functional	Provide functions based on the collected information.	Learning ability and reasoning: the agent has the ability to learn and adapt its behaviour to fit their objectives. Proactivity and cooperation: the agent may take the initiative to reach its objectives, cooperating or not with other agents.	Combine MAS with ML algorithms, IoT technologies, Big Data, computational processing layer (edge, fog and cloud) and Intelligent Products.
Business	Provide services based on the AAS functions.	Autonomy: the agent can receive information and formulate a strategy to achieve its goals without external intervention. Service-orientation: the functionalities of the agent can be encapsulated and offered as services.	Combine MAS with Service-Oriented Architecture (SOA). The Directory Facilitator (DF) from the agent platform provides yellow pages services to other agents (i.e. registration and discovery services).

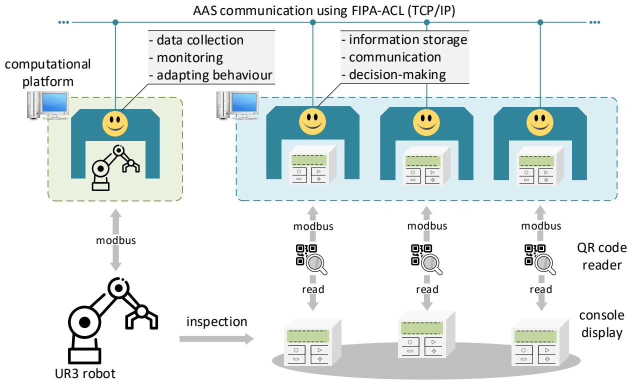

Figure 5. Architecture of the digitalized inspection station.

A. Smart Collaborative Robot

This example presents the digitalization of an UR3 collaborative robot, that is performing the inspection tasks of the console displays, bringing add value by offering services to improve the inspection station performance, e.g., monitoring the condition operation and adapting the inspection tasks.

The integration between the agent and the UR3 followed the recommended practices of the IEEE P2660.1. In this case, the agent is hosted on a remote computational platform, since the UR3 controller does not support embedding agents. As interaction mode, the client-server schema using the Modbus protocol was adopted, following the synchronous communication based on the request-response schema. For this purpose, it was used the Java Modbus Library (jamod) that allows accessing real-time UR3 data. The communication between the agents, e.g., the agent representing the UR3 robot and the product agents, is done over the Ethernet network using TCP/IP protocol, encoded for FIPA-ACL communication language. Moreover, the UR3 agent embed ML algorithms to assist in the inspection task, e.g., detecting non-compliant quality products and possible trends in the quality products, and also adapt the inspection conditions to the condition changes (namely different types of consoles or environment conditions).

B. Smart Product

This example considers the digitalization of a product, i.e., the console display being checked in the inspection station, being important to transform it into an Intelligent Product that can manage and optimize its behaviour along its lifecycle. The digitalization of the console display product aims to manage its information across the lifecycle, communicate and make decisions to solve internal problems or improve its behaviour.

For each product, one agent is responsible for its management, acting as an intelligent core, storing the product information (e.g., identification, production process and quality results), processing information and communicating with other agents using FIPA-ACL through the TCP/IP protocol, as well as offering the real-time monitoring and traceability capabilities across the lifecycle. As the console display does not provide computational capabilities the agent is running in a remote computational platform, communicating with the QR code reader to read the product information associated to the console using the Modbus protocol (note that each console is identified by a QR code).

VI. CONCLUSIONS

The digitalization of manufacturing assets taking into consideration the RAMI4.0 architecture is a suitable approach to reduce the complexity and achieve interoperability in modern manufacturing systems. This model is implemented through the I4.0 components, combining the asset with its digital representation, the AAS. MAS seems a suitable enabler to implement the AAS functionalities based on the descriptions of RAMI4.0, as well as to embed intelligence in the system.

This paper discusses the key challenges for implementing AAS and focuses on the analysis of how MAS technology can be used to implement the AAS, mapping its characteristics and supporting functionalities into AAS functionalities. The feasibility of using MAS to implement the AAS functionalities has been illustrated through two industrial application examples, namely the digitalization of the UR3 robot and console display products, both covering all layers of RAMI4.0. In both cases, the digitalization was performed by implementing the agents using the JADE framework, which is FIPA compliant, being the integration performed by using the Modbus protocol. The agents can communicate with each other by using TCP/IP protocol and following the FIPA-ACL communication language. These agents, besides the real-time collection and storage of data from their assets, also implement some ML algorithms to support the monitoring and adaptation of their behaviour according to condition changes.

Future work will be devoted to extend the mapping of AAS functionalities into the MAS technology, particularly addressing the identified challenges and using industrial standards, namely OPC UA and AutomationML.

ACKNOWLEDGMENTS

This work has been supported by FCT - Fundação para a Ciência e Tecnologia within the Project Scope: UIDB/05757/2020, and partially supported by the European Regional Development Fund (ERDF) through the Interreg Spain-Portugal V-A Program (POCTEP) under grant 0677_DISRUPTIVE_2_E (Intensifying the activity of Digital Innovation Hubs within the PocTep region to boost the development of disruptive and last generation ICTs through cross-border cooperation).

REFERENCES

[1] S. Karnouskos, L. Ribeiro, P. Leitão, A. Lüder, and B. Vogel-Heuser, "Key directions for industrial agent based cyber-physical production systems," in *Proceedings of the IEEE International Conference on Industrial Cyber Physical Systems (ICPS'19)*, 2019, pp. 17–22.

[2] 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.

[3] J. Fuchs, J. Schmidt, J. Franke, K. Rehman, M. Sauer, and S. Karnouskos, "I4.0-compliant integration of assets utilizing the asset administration shell," in *Proc. of the 24th IEEE Int'l Conference on Emerging Technologies and Factory Automation*, 2019, pp. 1243–1247.

[4] P. Leitão, A. W. Colombo, and S. Karnouskos, "Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges," *Computers in Industry*, vol. 81, pp. 11–25, 2016.

[5] ZVEI, "Reference architecture model industrie 4.0 (rami4.0)," 2015.

[6] ——, "Specification - details of the asset administration shell," 2019.

[7] R. Rosendahl, A. Calà, K. Kirchheim, A. Lüder, and N. D'Agostino, "Towards smart factory: Multi-agent integration on industrial standards for service-oriented communication and semantic data exchange," in *Proc. of the 19th Workshop "From Objects to Agents" (WOA'18)*, 2018.

[8] M. Wooldridge, *An Introduction to Multi-Agent Systems*. John Wiley and Sons, 2002.

[9] R. Sharpe, K. van Lopik, A. Neal, P. Goodall, P. P. Conway, and A. A. West, "An industrial evaluation of an industry 4.0 reference architecture demonstrating the need for the inclusion of security and human components," *Computers in Industry*, vol. 108, pp. 37 – 44, 2019.

[10] ZVEI, "Industrie 4.0: The reference architectural model industrie 4.0 (rami 4.0)," 2015.

[11] Platform Industrie 4.0 (I4.0), "Relationships between i4.0 components — composite components and smart production," 2017.

[12] L. A. Cruz Salazar, D. Ryashentseva, A. Lüder, and B. Vogel-Heuser, "Cyber-physical production systems architecture based on multi-agent's design pattern—comparison of selected approaches mapping four agent patterns," *The Int. Journal of Adv. Manufacturing Technology*, 2019.

[13] X. Ye and S. H. Hong, "An automationml/opc ua-based industry 4.0 solution for a manufacturing system," in *Proceedings of the IEEE 23th International Conference on Emerging Technologies and Factory Automation (ETFA'18)*, vol. 1, 2018, pp. 543–550.

[14] A. Lüder, M. Schleipen, N. Schmidt, J. Pfrommer, and R. Henßen, "One step towards an industry 4.0 component," in *Proceedings of the 13th IEEE Conference on Automation Science and Engineering (CASE'17)*, 2017, pp. 1268–1273.

[15] P2660.1, "Recommended practices on industrial agents: Integration of software agents and low level automation functions." [Online]. Available: https://standards.ieee.org/develop/project/2660.1.html

[16] C. Sharma, R. Sinha, and P. Leitao, "Iaselect: Finding best-fit agent practices in industrial cps using graph databases," in *Proceedings of the IEEE 17th International Conference on Industrial Informatics (INDIN'17)*, vol. 1, 2019, pp. 1558–1563.

[17] J. Queiroz, J. Barbosa, J. Dias, P. Leitão, and E. Oliveira, "Development of a smart electric motor testbed for internet of things and big data technologies," in *Proceedings of the 43th Annual Conference of the IEEE Industrial Electronics Society (IECON'17)*, 2017, pp. 3435–3440.

[18] J. Queiroz., P. Leitão., J. Barbosa., and E. Oliveira., "Distributing intelligence among cloud, fog and edge in industrial cyber-physical systems," in *Proceedings of the 16th International Conference on Informatics in Control, Automation and Robotics (ICINCO'19)*, 2019, pp. 447–454.

[19] H. Xu, W. Yu, D. Griffith, and N. Golmie, "A survey on industrial internet of things: A cyber-physical systems perspective," *IEEE Access*, vol. 6, pp. 78 238–78 259, 2018.

[20] A. Sadeghi, C. Wachsmann, and M. Waidner, "Security and privacy challenges in industrial internet of things," in *Proc. of the 52th ACM/EDAC/IEEE Design Automation Conf. (DAC'15)*, 2015, pp. 1–6.

[21] McKinsey & Company, "Digital and risk a new posture for cyberrisk in a networked world," 2018.

[22] F. Bellifemine, G. Caire, and D. Greenwood, *Developing Multi-Agent Systems with JADE*. Chichester, England: NJ: John Wiley, 2007.

[23] P. Leitão, N. Rodrigues, C. Turrin, and A. Pagani, "Multiagent system integrating process and quality control in a factory producing laundry washing machines," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 4, pp. 879–886, Aug. 2015.

[24] J. Barbosa, P. Leitão, A. Ferreira, J. Queiroz, C. A. S. Geraldes, and J. P. Coelho, "Implementation of a multi-agent system to support zdm strategies in multi-stage environments," in *Proc. of the IEEE 16th Int. Conf. on Industrial Informatics (INDIN'16)*, 2018, pp. 822–827.