Chapter 7
Integration of Cutting-Edge Interoperability Approaches in Cyber-Physical Production Systems and Industry 4.0

Luis Alberto Estrada-Jimenez
CTS UNINOVA, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Portugal

Terrin Pulikottil
CTS UNINOVA, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Portugal

Nguyen Ngoc Hien
https://orcid.org/0000-0003-4618-9462
Mondragon Unibertsitatea, Spain

Agajan Torayev
Institute for Advanced Manufacturing, University of Nottingham, UK

Hamood Ur Rehman
University of Nottingham, UK & TQC Ltd., UK

Fan Mo
Institute for Advanced Manufacturing, University of Nottingham, UK

Sanaz Nikghadam Hojjati
https://orcid.org/0000-0002-0839-9250
CTS UNINOVA, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Portugal

José Barata
CTS UNINOVA, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Portugal

ABSTRACT

Interoperability in smart manufacturing refers to how interconnected cyber-physical components exchange information and interact. This is still an exploratory topic, and despite the increasing number of applications, many challenges remain open. This chapter presents an integrative framework to understand common practices,

DOI: 10.4018/978-1-7998-6721-0.ch007

concepts, and technologies used in trending research to achieve interoperability in production systems. The chapter starts with the question of what interoperability is and provides an alternative answer based on influential works in the field, followed by the presentation of important reference models and their relation to smart manufacturing. It continues by discussing different types of interoperability, data formats, and common ontologies necessary for the integration of heterogeneous systems and the contribution of emerging technologies in achieving interoperability. This chapter ends with a discussion of a recent use case and final remarks.

INTRODUCTION

In recent years, manufacturing has experienced several changes because of the intensive development and research in sciences and technology and the provision of necessary equipment and systems to optimize industrial processes. The fourth industrial revolution precisely describes a new manufacturing paradigm engaging emerging technologies like machine learning, big data, internet of things, etc., and offering benefits like increased efficiency, fault tolerance, cognition and autonomy. In this regard, Cyber-Physical Production Systems (CPPS) emerge as one of the main enablers of Industry 4.0. Components of CPPS are smart and autonomous, connected in all levels in the production life cycle and provide fundamentally "intelligence, connectedness and responsiveness" (Monostori et al., 2016). Usually, a smart manufacturing environment is composed of various CPPS, which are continuously exchanging information and interacting. Currently, this is a focus of continuous research considering the high degree of heterogeneity in production systems (Y. Lu, 2017) also known as manufacturing interoperability. In this work, we refer to interoperability as a set of methodologies, tools and strategies needed to achieve information exchange. This also includes strategies and technologies utilized for the digitalization of machines, products, and internet platforms for data storage and data analysis.

The development of interoperability among CPPS has been tackled by industries as a standardization issue. Many authors consider that the creation of standardized interfaces and protocols may decrease the skepticism for the introduction of CPPS in industry (Leitão, Colombo, & Karnouskos, 2016). On the other hand, various researchers have implemented approaches using emerging technologies to show the benefits of principles of CPPS (Monostori et al., 2016a; Chaplin et al., 2015; Colombo, AW; Karnouskos, S; Mendes, 2015). Some examples are agent technologies, service based frameworks and cloud platforms. These technologies, standards and protocols are showing promising results but also new challenges that need to be overcome to reach a seamless integration. This chapter presents an integrative framework to explore common definitions, concepts, architectures, standards, technologies and a real case scenario considering interoperability approaches in smart manufacturing. The main objective of this study is to be a supportive conceptual text for researchers and practitioners in future implementations.

BACKGROUND

The fourth industrial revolution and the increasing research in information and communication technologies (ICT) results in a continuous evolution in the level of industrialization and technological development of factories. This new level of interaction and heterogeneity has brought the need of standardization and modelling of production systems as reference architectures. Those are used to describe high-level models and their internal relation.

Standard development organizations of countries such as USA, Germany and China have developed roadmaps and standardized solutions for smart manufacturing to integrate emerging ICT into the manufacturing domain. For instance, the Reference Architectural Model Industrie 4.0 (RAMI 4.0) provides a tridimensional description of the production life cycle, hierarchies and different layers of a smart production system. Similar characteristics are shared by the Smart manufacturing ecosystem (SME) developed by the National Institute of Standards and Technology (NIST - USA) and by the Intelligent Manufacturing System Architecture (IMSA) developed by the Ministry of Industry and Information technology of China.

Certainly, those standardization efforts show the high level of commitment of governments and the high interest of industrial stakeholders for the integration of cyber-physical components creating a smart and highly interconnected environment. This level of integration goes from an inter-organizational (horizontal integration) to a local or intra-organizational point of view (vertical integration), being the latest essential for a seamless collaboration in an enterprise hierarchy (Alcácer & Cruz-Machado, 2019).

A CPPS is a set of computational systems with high interconnection with physical resources, which precisely describes the necessity of interconnection and interoperability in industry 4.0. CPPS represent very heterogeneous units with the capacity of abstraction physical resources, products, legacy systems and even people's behaviors. Therefore, the communication and intercommunication of these entities are considered a challenging effort because of its heterogeneous nature. This high level of integration and collaboration requires very high levels of interoperability that is the main topic of discussion in this chapter.

To understand this issue we should understand what the definition of interoperability is. The IEEE standard computer dictionary defines interoperability as “the ability of two or more systems or components to exchange information and to use the information that has been exchanged”(Geraci, 1991). This emphasizes the capacity of communication between different systems despite their technological nature. This definition can be, however, abstract when referring to manufacturing systems. In this regard, a collection of specific definitions from selected works of the literature is presented and discussed below.

- In their work (Rojas & Rauch, 2019), describe interoperability as the continuous data and information accessibility with the production elements. Additionally, this work mentions that the challenge of interoperability is based on the data formalization, networking and connectivity. Finally, it is concluded that high levels of integration are strongly linked with high levels of interoperability.
- For (Zeid et al., 2019), the process of interoperability is highly related to the interaction of machines and the way how they are controlled. In this sense, real time interaction and communication is imperative to prevent failures and to ensure a high availability of the information. This can improve not just the safety but also the efficiency of the production process. Interoperability requires a high availability and collaboration from services inside and outside the shop floor and the utilization of cloud base technologies. For this purpose, a common understanding and data representation is required.
- For (Napoleone et al., 2020), a high degree of interoperability refers to a high degree of standardization. A well-structured representation as well as a proper integration of legacy systems is essential to implement CPPS and to ensure an easier integration of its components.
- In (Van Der Veer & Wiles, 2008), the concept of interoperability is referred as “the ability of equipment from different manufacturers (or different systems) to communicate together on the same infrastructure (same system), or on another while roaming”.

Indeed, the challenge of interoperability is born with the need of a seamless and high integration and cooperation of all levels in a factory: people, machines, business, organizational aspects, etc., and in this in turn with the value chain. Previous definitions suggest that the challenge of interoperability in smart manufacturing is not a single issue and several aspects should be consider. For example, compatibility of data types, abstraction levels, proper technological enablers, etc. Additionally, in smart manufacturing, interoperability should be addressed in a robust manner. Aspects like real time communication and high availability of services and resources are imperative to achieve the expectations of industry 4.0 optimizing processes and making them autonomous with little or non-human intervention. The common understanding, standardization and continuous evolution of technologies are paving the way to fulfill current expectations and even though there are many challenges that need to be overcome; there is currently a strong baseline of concepts, research and applications.

The following sections of this document are dedicated to address common approaches, emerging technologies and applications with regard to interoperability in smart manufacturing.

Types and Levels of Interoperability in the Integration of Cyber-Physical Production Systems

The IEEE Guide to the Enterprise Information Technology Body of Knowledge (EITBOK) has categorized the interoperability approaches mainly into two types: Syntactic and Semantic (Mosley, M., 2009), but in recent years we have seen the emergence of different categories or perspectives of interoperability like technical, organizational, device, networking, platform interoperability, etc. This section explains along with the main two types also device, factory (vertical integration) and cloud manufacturing (horizontal integration) which are very relevant to the smart manufacturing applications.

Device Interoperability

The term device is mainly used in the Internet of things to refer to "smart objects" with capabilities of integration and communication. Devices are highly heterogeneous and can be exemplified as sensors, actuators, parts to be assembled, and several low-level control hardware. The literature classifies the different types of devices in low level devices like Radio Frequency Identification (RFID) tags or barcodes and high-level devices like Programmable Logic Controller (PLCs) and computational boards (e.g. Raspberry) considering their embedded computational power and communication capabilities. Additionally, for a seamless communication these devices (low level and high level) need to manage necessary standards and protocols. Thus, it can be referred to device interoperability as the way how these heterogeneous devices are integrated, including standards, protocols and different technologies.

The literature in device interoperability in smart manufacturing is by far extensive. In (Chaplin et al., 2015), the utilization of a Raspberry pi allows the integration of agent technology, which in turn allows the communication among all entities in the shop floor. This development board is later interfaced with a PLC, which acts as main controller of the process. Additionally, this work shows the utilization of RFID technology as a method to integrate and identify products in different production stages. In (Leitao & Barbosa, 2019), with the purpose of demonstrating self-organization in a modularized conveyor system, agents are implemented through the utilization of a raspberry pi which also receives signals from different sensors and communicates with other boards via WIFI technology. In (Garcia et al., 2016), a CPPS system has been developed based on the virtualization of several stations via two platforms: Arduino and Raspberry pi. Those are in charge of receiving and handling input/output signals from the connected stations and of interfacing them in the network using MODBUS TCP and OPC Unified Architecture (OPC-UA) technologies.

Syntactic Interoperability

In general, if various systems are capable of exchanging information, they have syntactic interoperability. While exchanging information or service from one system to another, the content of the message needs to be serialized. The sender encodes the data in the message and the receiver decodes the received message. The sender and receiver use rules specified in some grammar to encrypt or decrypt the messages. The need for syntactic interoperability arises when these rules are incompatible with the receiver's decoding rules, which leads to mismatching message parse trees. The European Telecommunications Standards Institute (ETSI) defines syntactic interoperability as follows (Van Der Veer & Wiles, 2008): "Syntactical Interoperability is usually associated with data formats. Certainly, the messages transferred by communication protocols need to have a well-defined syntax and encoding, even if it is only in the form of bit-tables. However, many protocols carry data or content, and this can be represented using high-level transfer syntaxes such as Hypertext Markup Language (HTML), Extensible Markup Language (XML) or Abstract Syntax Notation One (ASN.1)".

Syntactic interoperability is achieved with standardized data formats and communication protocols. This includes standards like HTML, XML or JSON. XML is widely considered in the internet community for markup in documents of arbitrary structure. XML is designed for markup in documents of arbitrary structure.

A major limitation of this type of interoperability is that it just considers the data format and gets the information from one place to another intact. It does consider the meaning of the transferred information nor applies logic to the fact being transferred and used.

Semantic Interoperability

Semantic interoperability ensures that the exchanges between requesters and providers of data make sense, and have a mutual understanding of the "meanings" of the demanded data.

Ontologies are necessary to prevent semantic issues. Therefore, numerous amounts of work are done using ontologies. Context modelling facilitates interoperability of manufacturing systems. To this end, (Bettini et al., 2010) compares different context modeling and reasoning techniques by describing requirements like heterogeneity, mobility, relationship and dependencies and efficient context provisioning for the context models and context management systems. Considering these requirements, the authors discussed and compared object-role based, spatial and ontology-based modeling techniques. Contextual ontological models provide clear advantages both in terms of heterogeneity and in terms of interoperability and is obtained only by implementing communicative languages. Consequently, a substantial number of authors employed ontologies in their works to achieve interoperability in the industrial domain. (Kumar et al., 2019) presents a survey of the ontologies for Industry 4.0, including different domains such as aerospace, construction, steel production, etc., and manufacturing processes such as packaging, process engineering, resource configuration, etc. This work gives a broad overview of the current state of the art ontologies for industry 4.0 and the standardization efforts. Authors discuss different ontologies such as Core Ontology for Robotics and Automation (CORA), Ontology for Autonomous Robotics (ROA), Ontology for Robotic Architecture (ORArch), Ontology for Industry 4.0 (O4I4) and their benefits in the representation of vocabulary to describe the key concepts in Industry 4.0. Although there are a variety of ontologies, there is a need for standardization of ontologies currently lacking in the literatures.

Interoperability in the manufacturing domain is challenged with different terms that people working within a particular group develop their vocabulary for particular elements or activities with which they often work. Ontologies provide a solution for this problem by using a common understanding of manufacturing-related terms and affecting manufacturing knowledge sharing.

Factory Interoperability

Factory interoperability is factories' ability to exchange information within and between each other in a logical and consistent manner. The latest advances in ICT have shifted the factory environment from data-drive to cooperative and knowledge driven. Factory integration and interoperability are key in tackling challenges in this environment. Some enablers to achieve integrated factories include knowledge sharing, web-based developments, use of common best practices and open-source applications. These enablers also help in achieving interoperability among factories.

Supply Chain – A network from suppliers to customers – is the most dominant structure for exchange of information in factories today (both business & technical). These information (passed using paper and telephone conversations before) has to be passed now electronically in a coherent manner throughout the supply chain considering international and regional standards. These standards along with corporate & national cultures and use of different products along the supply chain adds more challenge in sharing of coherent information. These challenges arise the need for a standard interoperability infrastructure. A slightly outdated study (Brunnermeier & Martin, 1999) on a $1 billion economic loss due to improper interoperability among the supply chain in U.S. automobile industry shows the impact of factory interoperability on manufacturing cost.

There are three principal approaches used in achieving interoperability, namely machine-to-machine solution, industry-wide standardization and Open Standards or Platforms. In machine-to-machine solution, each pair of partners has customized solution for exchanging information. The idea behind this approach is to make each machine interoperable with all its linked machine. This requires translation of syntax for each machine and clear understanding of its semantics. In Industry-wide standardization, the Original Equipment Manufacturer (OEM) commands its supply partners to have a common solution, usually an expensive proprietary one. In Open Standards, the infrastructure is built on a neutral and open standard form. This is the most effective approach in achieve interoperability as it tackles both the scalability issues in machine-to-machine solution and does not force an expensive solution as in Industry-wide standardization. This solution also provides long term stability for data storage which is especially useful for products with long life cycles like aerospace.

The reference models developed to address factory interoperability issues can be categorized into physical, functional and allocated architectures. One of the most widely discussed architecture model for manufacturing is RAMI 4.0. This model incorporates existing approaches (OPC-UA, IEC, AutomationML, ProSTEP, Field Device Integration) into the interoperability stack. Industrial Internet Reference Architecture (IIRA), another important reference model designed for Industrial Internet of Things was proposed by Industrial Internet Consortium (Lin et al., 2017). Even though IIRA was not designed for factories, IIRA shares lot of similarities with RAMI 4.0: similar layers, same tasks distribution and applies OPC-UA for network communications. Other reference architecture for achieving factory interoperability include IBM 4.0 Reference Architecture for Industry 4.0 and NIST service-oriented architecture.

Cloud Manufacturing Interoperability

Cloud computing - providing resources and services over internet - is a key enabler of smart manufacturing. Cloud computing can be adopted in manufacturing in two ways: as direct adoption of cloud computing services or as cloud manufacturing. Cloud manufacturing is an extension of cloud computing where physical assets are managed in a centralized manner by encapsulating them as cloud services. There is a need to introduce additional types of interoperability for cloud manufacturing like transport, behavioural and policy interoperability.

A typical cloud-based manufacturing architecture consists of five layers: application, interface, core service, virtualization, and physical resource. Interoperable Cloud-based Manufacturing System (ICMS) proposed by (X. V. Wang & Xu, 2013) could be taken as an example for explaining interoperability in a service-oriented system. ICMS comprises three cloud layers: user, smart and manufacturing. Common data models supported by control rules are fundamental requirement for incorporating such architectures for a wide product scheme. The data models in this case should support a common data standard like STEP/STEP NC for interoperability purpose. A detailed explanation of cloud-based services, platforms, different layers, and interoperability of cloud manufacturing is explained in next section.

Interoperability of Cyber-Physical Production Systems Through Emerging Technologies

Interoperability in CPPS could be explained by considering the various emerging technologies and its impact on the interoperability. This section explains interoperability based on agent technologies, service-oriented technologies and computing technologies like Cloud, Fog and Edge.

Agent Based Interoperability

Multi-Agent Systems (MAS) have been extensively applied in distributed artificial intelligence and software engineering to implement software units with intelligent capabilities. Michael Wooldridge (Wooldridge, 2009) defines agent as "a computer system capable of autonomous action in order to meet its design objectives". Agents have cognitive capabilities i.e. they can acquire external information, reason and perform defined preprogramed tasks.

MAS as part of a society, do not have global contextual knowledge of the operation environment; instead, they present a partial local understanding and the global reasoning is the result of the social ability, communicating needs and objectives among all the entities in the group.

Agent Communication

JADE supports the development of agents under the FIPA-ACL protocol. It is implemented in the JAVA language that facilitates the implementation of agents as objects. Furthermore, JADE can distribute agents over the networks. JADE also offers the deployment and testing of the agent communication using a graphical user interface. These properties have made JADE to be used in countless industrial applications and uses cases.

Multi-Agent Systems in Manufacturing

Traditional centralized industrial control approaches are not prepared for dealing with novel business paradigms i.e. mass customization. Furthermore, the development of ICT technologies, industry 4.0 and the globalized economy have brought the need of industries to have lower production costs, higher flexibility, higher quality of production and the need for a rapid response in case of disruptions. For this purpose, the future of manufacturing systems should be highly automated, flexible, modular, interoperable, and easily changeable (Mourtzis & Doukas, 2012). MAS appear as a powerful technological enabler for having not just modularity and autonomy but also interoperability among the resources in the workshop (Leitão, 2009). This general conception takes the definition of resource virtualization, which means the encapsulation of the digital behavior, properties and functionalities of physical resources as intelligent entities, in this case agents. These smart units can interact, have social abilities and communicate their necessities. In manufacturing this is translated to machines that can communicate with mobile elements, resources, products, humans, etc. For these reasons, agents are considered as very powerful enablers of integration and interoperability for CPPS.

Multi-Agent Systems for Achieving Interoperability and Integration in CPS

The general conception of CPS requires the confluence of physical devices with communication and computation aspects, which as stated before can be achieved using agent technologies. Agent based-CPS also provide several characteristics that improve integration and interoperability in smart manufacturing, some of these are summarized below (Cruz Salazar et al., 2019; Leitão, Colombo, & Karnouskos, 2016; Ribeiro, 2017):

- The vertical and horizontal integration is possible through the resource agentification and its communication over the network.

- The resulting integration can be improved by the intelligence capacity of agents optimizing energy, time and resource utilization.
- Human resources can be also represented by agents, which makes feasible its integration.
- The integration of CPS becomes very robust since it can handle unexpected situations and even behave autonomy.

The CPS standardization is a key aspect for the design and development of industrial CPS. However, there are currently not many standards available for its implementation and it is a topic of continues research. Present works rely to a large extend in reusing and combining IT technologies e.g. service-oriented approaches. Agents and web services can be integrated to provide the best of both worlds, considering the autonomy of agent technology and the interoperability provided by service-oriented architectures. Generally, the components at the lowest level e.g. controllers or PLCs provide their functionalities as services using the DPWS (Device Profile for Web Services) protocol (Colombo, AW; Karnouskos, S; Mendes, 2015). This creates virtual resources which highest control and interoperability can be implemented using a multi-agent approach.

The integration of low-level controllers with MAS normally relies on legacy PLC programming standards like IEC-61131 or IEC-61149. These standards are based on a control logic and in service-based function blocks respectively. In higher levels this control logic is managed by a multi-agent based approach (Leitão, Colombo, & Karnouskos, 2016). This execution is generally designed in a higher level that can adapt its behavior to different scenarios. To complete the orchestration, this process is normally governed by business or higher functions that manage the whole enterprise integration (Colombo, AW; Karnouskos, S; Mendes, 2015).

The application of agent technologies in smart manufacturing is wide in terms of the type of functionalities or level of integration. As resources are abstracted by single mechatronic agents those in turn can also be abstracted by different alliances according to the needed collaboration and skills (Onori et al., 2012). This resource virtualization includes management functionalities and customer operations allowing a broader integration i.e. in cloud platforms (Vogel-Heuser et al., 2014). This also influences the interoperability of various entities in the value chain. This evidence suggests that MAS have been implemented using different patterns depending on the type of applications. A complete discussion and classification of these patterns is made in (Cruz Salazar et al., 2019) and it is summarized below.

- **Resource access:** this type of agents generally includes the abstraction of field devices, resources and their operational control. Besides, they promote modularization and integration with higher layers.

- **Communication agent:** This pattern includes agents that manage and unify the communication of resources through upper layers. For example, resource agents normally are integrated using OPC-UA, FIPA, broker agents, ontologies, etc.
- **Process agent:** This type of pattern usually orchestrates and manages resource agents (locally). They normally do not interact with field devices but with digital entities. They are in charge of the coordination, diagnosis and supervision of processes and of their proper execution.
- **Agent management system:** It usually has a global supervisory role. Unlike, the process agent that refers to local supervision, this pattern can have a broader vision of the process.

Indeed, multi-agent technology through its inherent capabilities of autonomy and communication has brought opportunities to face with many of the current industrial integration and interoperability requirements that has been also enhanced with various technological enablers and standards.

Service Based Interoperability

An increasing complexity in manufacturing systems is often composed as a set of numerous multi-disciplinary and heterogeneous systems, such as maintenance, engineering, warehouse and management. Those systems are considered as active components offering their capabilities as services representing mechatronic functions of equipment. It leads to the requirements of communication, data processing and interconnections among these services to fully utilize the benefits of flexibility, adaptability, scalability, seamless and effective integration. The requirements are also well-recognized limitations of traditional manufacturing systems whose architectures have encountered the following main issues (Cândido et al., 2009; Ye et al., 2018):

- Complex and time-consuming reconfiguration to adapt the market changes.
- Highly centralized resource utilization, unidirectional information flow and discrete decision-making.
- Exponential complexity in scalability.
- Incompatibility between different manufacturing equipment and standalone specialized engineering tools in both internal and external business systems.
- Machines and other operating units in shop-floor systems are still commonly isolated from higher-level business environments, although manufacturing execution systems (MES) or similar enterprise management systems are becoming increasingly available in the industry.

At the same time, manufacturing today requires a dynamic environment to meet the turbulent market demand for highly customized products with high quality at low cost, fastest possible time-to-market via a complex supply chain from product level to connected business world. These limitations could be overcome by leveraging the information technology (IT) infrastructure of web services with the concept of service-oriented architecture (SOA) emerged at multiple organizational levels in business, and applied into the factory automation domain.

Generally, a SOA is composed by consumers and services that are participated and coordinated, meaning that they interact with one another to request services and resolve these requests via interactions determined by the service interface.

Main SOA Principles

To have an effective and sustainable implementation of SOA, there are more than technical capabilities. As other business systems, a successful SOA needs to integrate and embrace critical design principles related to development and management. In the context of SOA, a set of design principles is to define the framework of guidance in which service provides and business customers will plan for collaboration during the design and development of the system. Even though, the essential design principles have still been under discussion(Legner & Heutschi, 2007), below some of these critical designed principals are summarized:

- **Business values:** a SOA service is defined by focusing on how the service may fit in a larger business process context by following an outside-in approach and adapting a business process centric instead of technology centric approach where the service often represents a business task (Jammes et al., 2005).
- **Discoverability:** as the SOA service is designed, discovering a service is the first step to service consumption and reuse (Uddin et al., 2012).
- **Reusability:** a SOA service should be developed in the right extent of generalization so that the original users as well as new users can exploit its functionality (Legner & Heutschi, 2007).
- **Loose coupling:** when this principle is applied to the SOA design, the purpose is to protect the individual independency of each SOA user and SOA service to mitigate the impact of changes in underlying technology and behavior.
- **Stateless:** this principle is also represented by the granularity of services where a service interface is mostly coarse-grained and based on the exchange document and messages.

A successful building architecture of SOA and services deployment will incorporate services and artifacts that take business values, discoverability, reusability, loose coupling, stateless, interloper-ability into account. The designers also need to have a framework with a set of rules from which compliance to these principles can be measured, monitored and even the contingency plan for appropriate remediation of noncompliance can be made.

Main SOA Applications and Supporting Standards

It is very clear that SOA paradigm is currently expanding its impact in many fields of technologies, not only in the ICT sector where it originated, but also in other domains of applications in which industrial applications have been adopted with several business collaborative initiatives (Jammes et al., 2005). Many service-oriented solutions have been proposed by European research projects, such as the Internet of Things at Work, Production Logistics and Sustainability Cockpit, Architecture for Service-Oriented Process - Monitoring and Control(IMC-AESOP)(Ismail & Kastner, 2017). Those SOA solutions are not specific to any technology, vendor, or product, but there is a combination of different technology capabilities enabling SOA functionality, such as Enterprise Service Business, Service Registry and Repository, Business Process Management, Business Activities Monitoring, and Web Services Management (WSM) that is considered as one of the most common SOA service types.

Cloud Based Interoperability

The manufacturing sector is undergoing a change in which the demand of customized product is increasing and the supply chains are taking a globalized perspective. This globalization of manufacturing and supply chains has brought with itself the need and use of globally distributed, scalable, sustainable and service-oriented manufacturing platforms. A platform that takes this into account and builds on computing technologies, cloud computing, semantic web and associated service-oriented architecture is Cloud Manufacturing that caters to resource sharing, distribution and management of manufacturing services across the network. However, the system in the manufacturing environment in itself needs to be setup for combination and interoperability to cater the requirements for cloud manufacturing. The effective utilization of these modelled resources is carried out in manufacturing cloud to establish a framework for manufacturing process. For such kind of integration, it is essential that a neutral API (Application Programming Interface) is utilized to establish a direct connection to manufacturing environment without changing enterprise wide structure.

Cloud Manufacturing Architecture

Several architectures have been proposed for cloud manufacturing environments with principles catered to integration at all manufacturing levels. (Tai & Xu, 2012) developed a five-tier cloud manufacturing system that dealt with co-operation between manufacturing resources, resource management, portal for cloud manufacturing, a unified cooperation platform and cooperation support application layer. A cloud manufacturing solution for automotive sector was explored by (Jin, 2013) in which it was treated as a Software as a Service (SaaS) based on four-layer architecture (core service, business service, cloud sub-system service and related business service). A detailed architecture for cloud computed manufacturing environment was established by (Tao et al., 2011) comprising of resource, perception, resource virtualization, cloud service, application, portal, knowledge, cooperation layer, security and internet layer. (Ai et al., 2013) based their six-tiered architecture on product information sharing and integration of cloud security modules on cloud platforms. (Zhang et al., 2014) expanded the standard cloud manufacturing architecture by introducing internet of things, service-oriented technologies and high-performance computing into the mix. The research built up a prototype system for cloud manufacturing for targeting TQCSEK (Faster time to market, higher quality, lower cost, better service, cleaner environment and high knowledge). However, the prime issue with such kind of manufacturing environments is the lack of fluidity in centralized management, lack of proper service distribution mechanism, efficiency, quality and timeliness. The realization of resources is usually carried out by a perception layer that is comprised of perception, connection, information technology and processing. Service layer builds on the perception layer to establish service pool of resources and capabilities. The working layer is responsible for interaction protocols, extensive transactions and management of tasks. The application layer is primarily concerned with interacting with users through APIs and cloud-end interface.

Cloud Manufacturing Frameworks

Cloud manufacturing frameworks provide support to developed architectures. This define the principle how the layers of the architecture communicate. An idea for cloud manufacturing task scheduling for resources was developed by (Li et al., 2012) wherein tasks were decomposed and matched with resource requirement by matching static properties. A framework for sensor-driven process planning environment for distributed setups was established by (L. Wang, 2008) name Wise-Shop Floor dealing with scheduling, monitoring, control and planning of resources. A service and web-oriented architecture framework with SaaS offering collaboration of internal operation with customers and supply chain network was established by

(Y. K. Lu et al., 2012). The majority of literature builds the cloud environment in manufacturing on grounds of manufacturing resources, manufacturing capability and cloud manufacturing services. The containerization here is of vital importance as the concept can be considered as a container wherein the resource is contained inside capability and the capability deployed in the manufacturing cloud as a cloud service. Design capability, production capability, experimentation, management and communication capability set up the baseline for manufacturing capability concept. On the other hand, the resources could be broken down into hard and soft resources with combination of them and capabilities yielding into a capability description model. Manufacturing cloud can be further extended to public and private cloud. Private cloud is organizational whereas the public cloud is society oriented. In both cases, the cloud services comprise of service layer, transmission network layer, resource layer, perception layer, virtual access layer and terminal application layer.

Cloud Manufacturing Platforms

Research into Cloud manufacturing platforms involves integration of data and resources across environment. (Valilai & Houshmand, 2013) presented a cloud-manufacturing platform XMLAYMOD that supported manufacturing collaboration and data integration on ISO 10303 (step standard). Distributed Integrated Manufacturing Platform (DIMP) by (Xu, 2012) provided basis for integrative CAx environment in production. The interaction happens on requests and task from the user. Cloud Agent for integrating services in platform was developed by (Jiang et al., 2012). For cloud-based manufacturing environment a cloud service for resource sharing was discussed by (Ding et al., 2012). Functionality of cloud manufacturing services and control was explored by (Xin-yu & Wei-jia, 2011). A communicational ability embedded cloud platform was presented by (Ferreira et al., 2014) to enhance interoperability. The work proposed a cloud-based web platform to support dashboard integrating communicational services and described an experimentation to prove that efficient interoperability in dynamic environment could be achieved only with human intervention.

Fog and Edge Based Interoperability

The increase in IoT device usages in complex scenarios like monitoring manufacturing devices, controlling production applications, energy optimization and other applications has brought about new opportunities and challenges. Cloud computing follows a centralized structure wherein the resources are centralized to a region or distributed on a remote Cloud server. A major drawback in cloud computing is presented towards reduced latency and real-time response. This is going to further increase only as more and more devices are connected to IoT networks.

Edge computing may be referenced as the accompanied tasks performed at the very Edge of the in house platform that is in direct communication with the cloud environment. The major advantages of Edge computing are to minimize latency, reduce cost and reduce bandwidth industry 4.0 applications. Edge computing application caters the problems of Cloud computing by optimizing storage and computing process before processing them to cloud services. So majorly processing then plays out at a prior stage before being sent to remote servers. Also, these processing tasks utilize essential parts of storage and computing near to the 'Edge' of device locally and no longer require Cloud services for these. This benefits in reducing the associated cloud costs, network traffic overloads, computational requirements for IoT applications.

Just like challenges served by edge computing i.e. high latency, low capacity and network failure, the fog computing brings devices closer to cloud. Locally available data processing and data storage at the device is offered by fog computing hence faster response and better quality. This makes fog computing an enabler for efficient and secure services for IoT device. Fog computing is considered to be an extension of cloud computing having nodes closer to end devices themselves. Fog devices act as an intermediary between cloud and end devices bringing processing, storage and networking closer to end devices deployed anywhere in a network.

Edge Computing Architecture

The reference architectures of Edge Computing consist of recommended structures, products and services that form industry specific standards, suggestions, best practices and optimal technologies that act as an enabler for edge computing. These architectures provide a means of collaboration and communication in an organization around an implementation project.

An architecture based on Edge Computing and Distributed Ledge Technologies (DLT) was developed to aid the adoption of decentralized automation as a part of H2020 FAR-Edge project. The architecture in itself presents a framework for implementing FAREdge project platform. On a general level, the architecture could be divided on scopes and tiers. The scope consists of elements that form the industrial environment such as machines, field devices, SCADA, MES and ERP system among others. Tiers on the other hand detail the system components and their association with each other. The architecture consists of three fundamental layers, namely field layer, edge layer and cloud layer. Edge Computing Consortium (ECC) jointly working with Industrial Internet Alliance (IIA) presented by Edge Computing Reference Architecture (EC-RA) 2.0 based on international standards like ISO/IEC/IEEE 42010:2011. Edge Computing Reference Architecture 2.0

(Edge Computing Consortium & Alliance of Industrial Internet, 2017) follows vertical and horizontal services and layer model. The vertical services involved are management, data life-cycle and security offering intelligence-based service for complete life cycle. Horizontally EC-RA 2.0 open interface layer model is projected with smart layer, service fabric and connectivity and computing fabric. Industrial Internet Consortium (IIC) like ECC also developed its own reference architecture using ISO/IEC/IEEE 42010:2011 standard. This standard assists in identification of convention, principle and practices for coherent architectures and frameworks. This architecture majorly consists of three layers mainly edge, platform and enterprise layer. Edge layer is responsible for data acquisition from edge nodes through its nearest proximity devices. The constituents feature of the layer is depth of distribution, nature of proximity network, the location of nodes and devices and governance policy. The intermittent platform layer is mainly responsible for sending command from Enterprise to Edge layer. It groups the processes for analysis of data flows and manages the active devices by consultation and analysis through domain servers. Enterprise layer houses support platforms responsible for generating control commands to Platform and Edge layer and receiving data flow.

The architectures provide a means of complimenting Cloud services as the last level of architecture rather than replace them. This presents a beneficial case whenever significant population of IoT data exists. Edge nodes in such cases could be the initiating point for accumulation, controlling and reducing the data before passing it forwards to cloud services.

Fog Computing Architecture

Traditional fog computing architectures consists of six layers, namely physical and virtualization, monitoring, pre-processing, temporary storage, security and transport layer. The physical and virtualization layer involves physical nodes, virtual nodes and sensor networks. The management of nodes in this case is dynamic depending on their types and service demand. Sensor network deployed over geographical locations sense the surroundings and send data to higher layers via gateways for analysis and processing. Monitoring layer on the other hand keeps check on availability and usage of resources, sensors, fog nodes and network infrastructure. This layer also deals with the type of tasks that need to be performed by this and consecutive nodes. Energy consumption may also serve as a driver for this layer as fog nodes uses many devices with varying energy requirement conditions. Data management is primarily deal in at pre-processing layer. Data collected, analyzed, filtered, and trimmed to drive useful information. The data stream from this layer is then housed in temporary storage layer. The data transferred to cloud is redacted from the local temporary storage layer. The transmission of data is affected by security layer where the encryption and decryption of data comes into play. Moreover, privacy and integrity features extend the security of data making it prone from tampering. This data, which is now hosted in transport layer, uploads the data in cloud for further usage. Fog computing enables segments of data to be uploaded in cloud through a smart gateway that manages data distribution to cloud. This emphasizes on proper communication protocols for fog computing with efficient, lightweight and customization of data stream to be major concerns. Therefore, fog computing communication protocols depend on application scenario of fog.

USE CASE ANALYSIS

Several state of the art industrial use cases show the continuous research effort to achieve integration and interoperability in the context of industry 4.0. In this section, the European project PERFoRM (Production harmonized Reconfiguration of Flexible Robots and Machinery) will be discussed as it covers a wide variety of interoperability and integration features that fit within the context of this chapter. Additional reading about this project can be found in (Angione et al., 2017; Leitão, Barbosa, Pereira, et al., 2016).

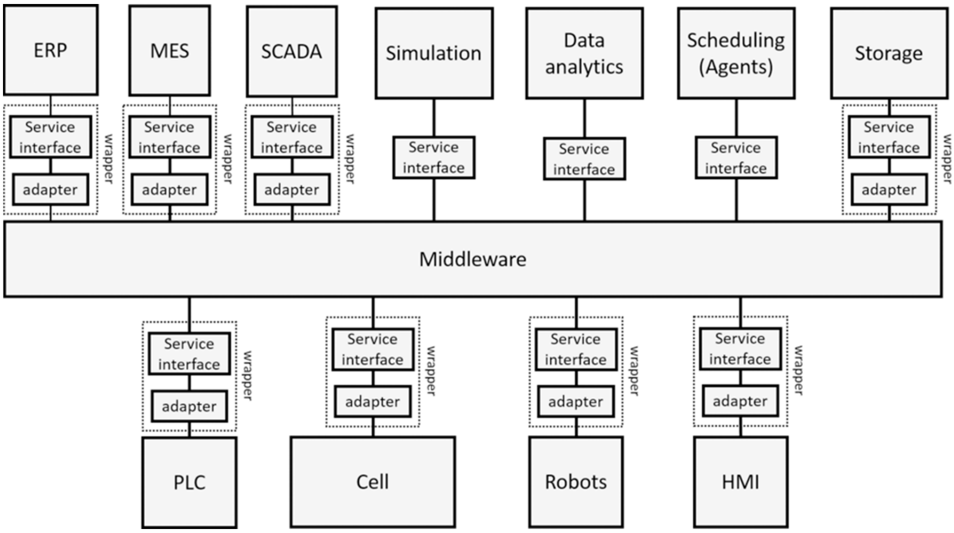
Context of PERFoRM and General Requirements

The H2020 PERFoRM is based on various European projects and has envisioned a set of best practices to develop an innovative distributed control architecture. PERFoRM highlights industry 4.0 compliant requirements based on three fundamental aspects **high interconnectivity** of components, a **dynamic reconfiguration** to prevent unexpected situations and the integration of **novel functionalities** like simulations and data analytics to promote industry 4.0 autonomy. The implementation of cyber-physical components rely on service-oriented approaches and a centralized middleware that provides the needed **infrastructure** to integrate, encapsulate and dynamically distribute services. This flexibility allows the integration of various manufacturing components e.g. people, legacy systems, software tools, etc. Additionally, enabling technologies and standards enhance the applicability of PERFoRM. An extract of this architecture is presented in Figure 1 and more details about its configuration are described in the following subsections.

The Role of Interoperability and Integration in PERFoRM

The service-oriented design of PERFoRM facilitates the exposure of atomic functionalities of the components as services. These services can be thereafter

Figure 1. PERFoRM architecture
Source: (Leitão, Barbosa, Pereira, et al., 2016)



easily integrated. Examples of this process servitization are found at single manufacturing machines (e.g. robots), industrial cells, human machine interfaces (**human integration**), servitization of ERP, MES, etc. This whole integration breaks the traditional pyramidal and hierarchical control schema. As a result, all elements in the factory are interconnected making manufacturing systems more efficient, robust and ready to business changes. The servitization promotes **factory interoperability** creating an integrated enterprise concept. Additionally, the use of **cloud architectures** make possible the integration of resource planning and data analytics tools. In PERFoRM, the integration of cloud technologies paves the way towards the introduction of external services for instance to implement supervisory or remote control. Furthermore, those services can be interconnected with the value chain creating a well stablished inter-enterprise model (**horizontal integration**).

The integration and events in PERFoRM rely to a large extend in multi-agent technology. This autonomy provides an intelligent environment for dynamic task scheduling in which the operational activities are virtualized utilizing agent technology and subsequently autonomously allocated. Such a way of task representation makes the process **robust** and **self-sufficient**. Another point of consideration in PERFoRM is the utilization of **technological adapters.** Adapters allow the integration of **legacy systems.** The utilization of adapters is based on the creation of **data models** compliant with the PERFoRM architecture but primarily with the ability to abstract functionalities of legacy systems in a detailed way so that they can be integrated.

We should to note the role of **standardization** in PERFoRM. On the one hand it is worth mentioning the capability of resources to be abstracted as part of the **RAMI 4.0 administration shell** and on the other hand the application of **AutomationML** and **OPC-UA** vendor independent standards. In the first case, the compliance with the RAMI 4.0 suggests a promising applicability of the architecture for practitioners and an adequate level of formalization to integrate heterogeneous resources into this formalized model. In the second case, the use of AutomationML and OPC-UA ensure standard interfaces and enables vendor independent integration of resources, assuring a **syntactic interoperability**.

The overall results achieved through the discussion of the PERFoRM architecture and its fundamental characteristics suggest different levels of manufacturing integration and interoperability, in which the servitization of all functional resources plays a fundamental role. Additionally, the use of enabling technologies increases the potentiality of PERFoRM, applying cloud platforms, agent technologies and adding a certain degree of autonomy and supplementary capabilities.

Particular Applications of PERFORM Aligned With Recent Interoperability Approaches

The PERFoRM architecture has been applied in several use cases e.g. large compressor, micro-electric vehicle and microwave ovens producer (Leitão, Barbosa, Foehr, et al., 2016). This has generated a technological assessment of methodologies to validate the applicability of the framework. In this sense, we have collected some of these results to validate them against the presented cutting-edge interoperability approaches.

The compressor producer use case consists of complex structural systems with heterogeneous components and which machines that have limited production, normally developed for specific applications. It also includes heavy equipment that cannot be set up multiple times. This cause maintenance tasks to be purely reactive and thus cannot be scheduled accordingly. Therefore, the objective of the application to stablish a proactive maintenance system reducing also possible delays. Specific blocks of this architecture include legacy hardware devices and software tools. Some examples are Maintenance task list, Data analytics and Simulation Reconfiguration for Maintenance services. Additionally, we should highly the presence of databases for maintenances tools and machines and for order equipment efficiency (OEE) services with specific interfaces (legacy systems). The whole architecture is integrated by the industrial middleware.

The micro-electrical vehicles producer use case aims to automate a manual factory that makes electrical vehicles. With the application of PERFoRM, it is envisaged to support efficiency, re-configurability and the integration of PLCs of various robotic cells. Resources of this architecture are integrated to the central middleware via web services. Some examples include welding robotic cells, testing stations and MES. Additionally, an agent based simulation, as well as scheduling tools are considered to increase efficiency and flexibility of the approach. A human machine interface (HMI) and related efficiency measures are also integrated to increase the traceability of the system.

While previous examples introduce general ideas of the description and implementation of PERFoRM, it is necessary to describe some of the specific requirements aligned with this architecture, as shown below.

Syntactic Interoperability

The selection of the adopted data exchange format is imperative considering standard interfaces for the applicability and interconnection of legacy hardware e.g. PLCs, SCADAs, MES and databases. Additionally, the compatibility with various automation layers (considering ISA 95) e.g. L1 (automation control), L2 (supervisory control), L3 (manufacturing operations management) and L4 (business planning and logistics) was considered. After an assessment with various associated criteria e.g. domain specific concepts, performance analysis, quality monitoring, material resource management, maintenance, etc., it was determined that a joint solution of B2MML (Business To Manufacturing Markup Language) and AutomationML (Peres et al., 2016) was successfully at fulfilling PERFoRM requirements. Particularly AutimationML highlights the lower level data integration and B2MML emphasizes standards for the integration of Enterprise control systems.

Human Integration

Under the umbrella of the PERFoRM architecture, two human integration types were considered: Human-in-the-Loop (HitL) and Human in the Mesh (HitM) (Fantini et al., 2016). HitL includes overseen and adjustment of set points, direct commands with the system and the capability of humans as a data source. Such activities refer to direct interaction with the CPS network and interaction with other humans. In general, specific requirements of HitL covers collaboration between humans and CPS, integrating the physical world sensing and controlling devices and digitizing resources.

HitL can be utilized for different application domains e.g. for planning (Fantini et al., 2016). They are more related to organizational methods that can influence human behaviour and its performance. Some of its requirements include simulation and continuous extraction of decision making with the aim of empowering human skills and CPS production scenarios.

Middleware

A middleware is a software application, responsible for the connectivity of the actors involved in the industrial communication. It can receive, translate and forward data from and to various components. (Gosewehr et al., 2016). A general assessment of requirements and functionalities for PERFoRM resulted in three candidates to be used depending on the use case scenario (Gosewehr et al., 2016). Those are WinCC OA (Siemens), mBS (Prosyst) and FuseESB (Red Hat). *Siemens WinCC OA* has a very flexible architecture and provides direct PLCs interfaces (device interoperability). Additionally, it has various components compatible with SCADA interfaces and even with Raspberry pi' hardware. *Prosyst mBS is* also an alternative for PERFoRM. Several advantages can be considered from the low processing RAM required. In addition, the solution seems very promising considering the compatibility with OPC-UA/DA support. Finally, *Red Hat Fuse ESB* highlights the utilization of technologies like *Apache Camel*, which can be used over almost any operative system that has a java virtual machine.

System Integration

The integration of the adapters utilized in PERFoRM consider REST services (Angione et al., 2017). Those are instantiated in Java and are linked to MySQL databases. Data models are described using for instance JSON objects (syntactic interoperability). These models harmonize the communication of various actors e.g. machines, adapters, middleware, etc. The messages are consumed using **REST services** and are routed using the middleware. Additionally, the PERFoRM **service provider** includes **WADL** (Web application Description Language) as an integrated description into each service and can be used to expose and discover available services via the common API. XML and JSON are used to interface the various services and as a parsing mechanism decreasing the necessary integration.

This discussion does not claim to be a complete analysis of the PERFoRM architecture, but intends to show how new research effort is managing new challenges in the interoperability and integration of CPPS and how these results can bring new possibilities in the future of manufacturing.

CONCLUSION

Emerging technologies and paradigms in manufacturing like CPPS, cloud manufacturing, smart manufacturing, Internet of Things etc., follow a highly interoperable and decentralized structure. This arises a need for integration among shop-floor devices, services and between enterprises and cloud service platforms. This chapter gives a brief overview of different interoperability approaches in smart manufacturing. The chapter also explains in detail the emerging technologies like MAS, SOA, Cloud and Edge/Fog for achieving interoperability. Future work in this direction will consider developing communication protocols built upon industrial standards e.g. OPC-UA and focusing on individual level of the network in smart manufacturing.

ACKNOWLEDGMENT

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie [grant No. 814078].

REFERENCES

Ai, Q., Mo, K., Wang, Y., & Zhao, L. (2013). Research of product information sharing system based on cloud manufacturing. *Applied Mechanics and Materials*, *248*, 533–538. doi:10.4028/www.scientific.net/AMM.248.533

Alcácer, V., & Cruz-Machado, V. (2019). Scanning the Industry 4.0: A Literature Review on Technologies for Manufacturing Systems. *Engineering Science and Technology, an International Journal, 22*(3), 899–919.

Angione, G., Barbosa, J., Gosewehr, F., Leitão, P., Massa, D., Matos, J., Peres, R. S., Rocha, A. D., & Wermann, J. (2017). Integration and Deployment of a Distributed and Pluggable Industrial Architecture for the PERFoRM Project. *Procedia Manufacturing*, *11*(June), 896–904. doi:10.1016/j.promfg.2017.07.193

Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., & Riboni, D. (2010). A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, *6*(2), 161–180. doi:10.1016/j.pmcj.2009.06.002

Brunnermeier, S. B., & Martin, S. A. (1999). *99-1 Planning Report: Interoperability Cost Analysis of the U.S. Automotive Supply Chain*. Academic Press.

Cândido, G., Barata, J., Colombo, A. W., & Jammes, F. (2009). SOA in reconfigurable supply chains: A research roadmap. *Engineering Applications of Artificial Intelligence*, *22*(6), 939–949. doi:10.1016/j.engappai.2008.10.020

Chaplin, J. C., Bakker, O. J., De Silva, L., Sanderson, D., Kelly, E., Logan, B., & Ratchev, S. M. (2015). Evolvable assembly systems: A distributed architecture for intelligent manufacturing. *IFAC-PapersOnLine*, *48*(3), 2065–2070. doi:10.1016/j.ifacol.2015.06.393

Colombo, A. W., Karnouskos, S., & Mendes, J. L. P. (2015). Industrial Agents in the Era of Service-Oriented Architectures and Cloud-Based Industrial Infrastructures. In *Industrial Agents* (pp. 67–87). Morgan Kaufmann. doi:10.1016/B978-0-12-800341-1.00004-8

Cruz Salazar, L. A., Ryashentseva, D., Lüder, A., & Vogel-Heuser, B. (2019). Cyber-physical production systems architecture based on multi-agent's design pattern—Comparison of selected approaches mapping four agent patterns. *International Journal of Advanced Manufacturing Technology*, *105*(9), 4005–4034. doi:10.100700170-019-03800-4

Ding, B., Yu, X. Y., & Sun, L. J. (2012). A cloud-based collaborative manufacturing resource sharing services. *Information Technology Journal*, *11*(9), 1258–1264. doi:10.3923/itj.2012.1258.1264

Edge Computing Consortium & Alliance of Industrial Internet. (2017). *Edge Computing Reference Architecture 2.0*. http://en.ecconsortium.net/Uploads/file/20180328/1522232376480704.pdf

Fantini, P., Tavola, G., Taisch, M., Barbosa, J., Leitao, P., Liu, Y., Sayed, M. S., & Lohse, N. (2016). Exploring the integration of the human as a flexibility factor in CPS enabled manufacturing environments: Methodology and results. In *IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society* (pp. 5711-5716). IEEE.

Ferreira, L., Putnik, G., Cruz-Cunha, M. M., Putnik, Z., Castro, H., Alves, C., & Shah, V. (2014). Dashboard services for pragmatics-based interoperability in cloud and ubiquitous manufacturing. *International Journal of Web Portals*, *6*(1), 35–49. doi:10.4018/ijwp.2014010103

Garcia, M. V., Irisarri, E., Perez, F., Estevez, E., Orive, D., & Marcos, M. (2016). Plant floor communications integration using a low cost CPPS architecture. In *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)* (pp. 1-4). IEEE. 10.1109/ETFA.2016.7733631

Geraci, A. (1991). *IEEE standard computer dictionary: Compilation of IEEE standard computer glossaries*. IEEE Press.

Gosewehr, F., Wermann, J., & Colombo, A. W. (2016). Assessment of industrial middleware technologies for the PERFoRM project. In *IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society* (pp. 5699-5704). IEEE. 10.1109/IECON.2016.7793611

Ismail, A., & Kastner, W. (2017). Surveying the Features of Industrial SOAs. In *2017 IEEE International Conference on Industrial Technology (ICIT)* (pp. 1199-1204). IEEE. 10.1109/ICIT.2017.7915533

Jammes, F., Smit, H., Martinez Lastra, J. L., & Delamer, I. M. (2005). *Orchestration of service-oriented manufacturing processes. In 2005 IEEE conference on emerging technologies and factory automation* (Vol. 1). IEEE.

Jiang, W., Ma, J., Zhang, X., & Xie, H. (2012). Research on cloud manufacturing resource integrating service modeling based on cloud-agent. In *2012 IEEE International Conference on Computer Science and Automation Engineering* (pp. 395-398). IEEE. 10.1109/ICSESS.2012.6269488

Jin, Z. (2013). Research on solutions of cloud manufacturing in automotive industry. In *Proceedings of the FISITA 2012 World Automotive Congress* (pp. 225-234). Springer Berlin Heidelberg. 10.1007/978-3-642-33747-5_21

Kumar, V. R., Khamis, A., Fiorini, S., Carbonera, J., Alarcos, A. O., Habib, M., Goncalves, P., Li, H., & Olszewska, J. I. (2019). Ontologies for industry 4.0. *The Knowledge Engineering Review*, *34*, 1–14.

Legner, C., & Heutschi, R. (2007). *SOA Adoption in Practice - Findings from Early SOA Implementations*. Academic Press.

Leitão, P. (2009). Agent-based distributed manufacturing control: A state-of-the-art survey. *Engineering Applications of Artificial Intelligence*, *22*(7), 979–991. doi:10.1016/j.engappai.2008.09.005

Leitao, P., & Barbosa, J. (2019). Modular and Self-organized Conveyor System Using Multi-agent Systems. In *International Conference on Practical Applications of Agents and Multi-Agent Systems* (pp. 259-263). Springer. 10.1007/978-3-030-24209-1_26

Leitão, P., Barbosa, J., Foehr, M., Calà, A., Perlo, P., Iuzzolino, G., Petrali, P., Vallhagen, J., & Colombo, A. W. (2016). Instantiating the PERFORM system architecture for industrial case studies. In *International Workshop on Service Orientation in Holonic and Multi-Agent Manufacturing* (pp. 359-372). Springer.

Leitão, P., Barbosa, J., Pereira, A., Barata, J., & Colombo, A. W. (2016). Specification of the PERFoRM architecture for the seamless production system reconfiguration. In *IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society* (pp. 5729-5734). 10.1109/IECON.2016.7793007

Leitão, P., Colombo, A. W., & Karnouskos, S. (2016). Industrial automation based on cyber-physical systems technologies: Prototype implementations and challenges. *Computers in Industry*, *81*, 11–25. doi:10.1016/j.compind.2015.08.004

Li, C., Yang, P., Shang, Y., Hu, C., & Zhu, P. (2012). Research on cloud manufacturing resource scheduling and performance analysis. *Advanced Science Letters*, *12*(1), 240–243. doi:10.1166/asl.2012.2780

Lin, S.-W., Murphy, B., Clauser, E., Loewen, U., Neubert, R., Bachmann, G., Pai, M., & Hankel, M. (2017). Architecture Alignment and Interoperability: An Industrial Internet Consortium and Plattform Industrie 4.0 Joint Whitepaper. *Plattform Industrie 4.0*, 19.

Lu, Y. (2017). Industry 4.0: A survey on technologies, applications and open research issues. *Journal of Industrial Information Integration*, *6*, 1–10. doi:10.1016/j.jii.2017.04.005

Lu, Y. K., Liu, C. Y., & Ju, B. C. (2012). Cloud manufacturing collaboration: An initial exploration. In *2012 Third World Congress on Software Engineering* (pp. 163-166). IEEE. 10.1109/WCSE.2012.39

Monostori, L., Kádár, B., Bauernhansl, T., Kondoh, S., Kumara, S., Reinhart, G., Sauer, O., Schuh, G., Sihn, W., & Ueda, K. (2016). Cyber-physical systems in manufacturing. *CIRP Annals*, *65*(2), 621–641. doi:10.1016/j.cirp.2016.06.005

Mourtzis, D., & Doukas, M. (2012). Decentralized manufacturing systems review: Challenges and outlook. *Logistics Research*, *5*(3-4), 113–121. doi:10.100712159-012-0085-x

Napoleone, A., Macchi, M., & Pozzetti, A. (2020). A review on the characteristics of cyber-physical systems for the future smart factories. *Journal of Manufacturing Systems*, *54*, 305–335. doi:10.1016/j.jmsy.2020.01.007

Onori, M., Lohse, N., Barata, J., & Hanisch, C. (2012). The IDEAS project: Plug & produce at shop-floor level. *Assembly Automation*, *32*(2), 124–134. doi:10.1108/01445151211212280

Peres, R. S., Parreira-Rocha, M., Rocha, A. D., Barbosa, J., Leitao, P., & Barata, J. (2016). Selection of a data exchange format for industry 4.0 manufacturing systems. In *IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society* (pp. 5723-5728). IEEE.

Ribeiro, L. (2017). Cyber-physical production systems' design challenges. In *2017 IEEE 26th International Symposium on Industrial Electronics (ISIE)* (pp. 1189-1194). IEEE.

Rojas, R. A., & Rauch, E. (2019). From a literature review to a conceptual framework of enablers for smart manufacturing control. *International Journal of Advanced Manufacturing Technology*, *104*(1–4), 517–533. doi:10.100700170-019-03854-4

Tai, D., & Xu, F. (2012). Cloud manufacturing based on cooperative concept of SDN. *Advanced Materials Research*, *482*, 2424–2429. doi:10.4028/www.scientific.net/AMR.482-484.2424

Tao, F., Cheng, Y., Zhang, L., Luo, Y. L., & Ren, L. (2011). Cloud manufacturing. *Advanced Materials Research*, *201*, 672–676. doi:10.4028/www.scientific.net/AMR.201-203.672

Uddin, M. K., Puttonen, J., Scholze, S., Dvoryanchikova, A., & Martinez Lastra, J. L. (2012). Ontology-Based context-Sensitive computing for FMS optimization. *Assembly Automation*, *32*(2), 163–174. doi:10.1108/01445151211212316

Valilai, O. F., & Houshmand, M. (2013). A collaborative and integrated platform to support distributed manufacturing system using a service-oriented approach based on cloud computing paradigm. *Robotics and Computer-integrated Manufacturing*, *29*(1), 110–127. doi:10.1016/j.rcim.2012.07.009

Van Der Veer, H., & Wiles, A. (2008). *Achieving Technical Interoperability: the ETSI Approach*. European Telecommunications Standards Institute.

Vogel-Heuser, B., Diedrich, C., Pantförder, D., & Göhner, P. (2014). Coupling heterogeneous production systems by a multi-agent based cyber-physical production system. In *2014 12th IEEE International Conference on Industrial Informatics (INDIN)* (pp. 713-719). IEEE. 10.1109/INDIN.2014.6945601

Wang, L. (2008). Wise-ShopFloor: An integrated approach for web-based collaborative manufacturing. *IEEE Transactions on Systems, Man and Cybernetics. Part C, Applications and Reviews*, *38*(4), 562–573. doi:10.1109/TSMCC.2008.923868

Wang, X. V., & Xu, W. W. (2013). ICMS: a cloud-based manufacturing system. In *Cloud manufacturing* (pp. 1–22). Springer. doi:10.1007/978-1-4471-4935-4_1

Wooldridge, M. (2009). *An introduction to multiagent systems*. John Wiley & Sons.

Xin-yu, Y., & Wei-jia, L. (2011). Research and application of the management and control platform oriented the cloud manufacturing services. In *2011 International Conference on System science, Engineering design and Manufacturing informatization* (Vol. 1, pp. 286-289). IEEE. 10.1109/ICSSEM.2011.6081208

Xu, X. (2012). From cloud computing to cloud manufacturing. *Robotics and Computer-integrated Manufacturing*, *28*(1), 75–86. doi:10.1016/j.rcim.2011.07.002

Ye, X., Park, T. Y., Hong, S. H., Ding, Y., & Xu, A. (2018). Implementation of a Production-Control System Using Integrated Automation ML and OPC UA. In *2018 Workshop on Metrology for Industry 4.0 and IoT* (pp. 1-6). IEEE. 10.1109/METROI4.2018.8428310

Zeid, A., Sundaram, S., Moghaddam, M., Kamarthi, S., & Marion, T. (2019). Interoperability in smart manufacturing: Research challenges. *Machines*, *7*(21), 1–17.

Zhang, L., Luo, Y., Tao, F., Li, B. H., Ren, L., Zhang, X., Guo, H., Cheng, Y., Hu, A., & Liu, Y. (2014). Cloud manufacturing: A new manufacturing paradigm. *Enterprise Information Systems*, *8*(2), 167–187. doi:10.1080/17517575.2012.683812