Cloud–based manufacturing (CBM) interoperability in Industry 4.0

István Mezgár, Gianfranco Pedone

Hungarian Academy of Sciences Institute for Computer Science and Control, Hungary

E-mail: {mezgar.istvan, pedone.gianfranco}@sztaki.mta.hu

Keywords: Cloud Computing, Cloud Models, IIRA, Cloud Interoperability, Internet of Things, Manufacturing Standardization, RAMI 4.0, Smart Factory.

Cloud-based manufacturing (CBM) interoperability in Industry 4.0

István Mezgár, Gianfranco Pedone

Hungarian Academy of Sciences Institute for Computer Science and Control, Hungary

E-mail: {mezgar.istvan, pedone.gianfranco}@sztaki.mta.hu

ABSTRACT

Cloud computing (CC) is generating new compute and business models thanks to its service-based nature, which enables collaboration and data exchange at higher level, more flexibility with better efficiency and parallel decreasing costs. Manufacturing environments can also benefit from cloud technology and follow fast changes in market demands. In these new scenarios interoperability has vital importance in the operation and interaction among industrial realizations of the Cyber-physical Systems. The paper introduces the different cloud models and the interoperability issues concerning connected enterprise information systems. Various standardization frameworks have been developed for homogeneous integration of IT models in industrial environments: the IIRA and the RAMI 4.0 are the best known. The paper introduces both of architectures, their methodological approach to industrial integration efforts and how integration feasibility might find realization through the OPC Unified Architecture. Last but not least, the authors propose a basic conceptual model for cloud manufacturing.

Keywords: Cyber Physical Systems, Cloud Computing, Cloud Models, IIRA, Cloud Interoperability, Internet of Things, Manufacturing Standardization, RAMI 4.0, Smart Factory.

INTRODUCTION

Information and communications technologies (ICTs) have an extremely fast evolution tendency, which often leads to creating new "digital" economy. Nowadays this evolution has reached a point where it can be (re)called revolution. The evolution of economy, industry and production systems can be split to different phases where the critical turning points in the evolution mean a revolution. The first industrial revolution was when mechanical production facilities were powered by water and steam (1784), the second one was the introduction of mass production with the help of electrical energy (1870), the third big turning point was the application of electronic and IT systems to further automate production worldwide (1969 first PLC).

Now the world is already in the Forth Industrial Revolution phase where in Cyber Physical Systems real and virtual objects, and processes are interlinked (201X - connected industry, smart factories). This phase can be featured as a deep interdisciplinary integration of technologies in the digital, physical, and biological world (*Schwab,* 2015).

This 4th phase can be called as revolution because the speed of transformation is exponential and the effects cover very broad fields, from e.g. industry to the society. Also manufacturing industry belongs to those sectors that are changing basically in all of their component systems (control, ITC, fabrication). New paradigms are coming into being, e.g. additive manufacturing, nanotechnology, cloud based manufacturing. The integration of subsystems has of vital importance and interoperable solutions provide the technical background for this fusion. The importance of this fourth industrial revolution (digitalization of economy, production and society) is so important that governments are launching initiatives, global national (framework) research projects to support the research and implementation activities in the most important ITC fields (*European Commission,* 2016; *NSF Program Announcements,* 2017).

The high-tech ICT-based strategic program of the German government, called "Industry 4.0", essentially focuses on manufacturing and describes the up-to-date automation and data exchange in manufacturing technologies. Industry 4.0 includes Cyber- Physical Systems, the Internet of Things and cloud computing. In the industrial sector the application of cloud computing is constantly growing. According to a statistics from US (*RightScale,* 2017) 95% of firms use cloud computing technology, while 25% call security as a significant challenge. Based on a former opinion of industrial experts in cloud environments interoperability is a bigger problem than security. "The greatest challenge facing longer-term adoption of cloud computing services is not security, but rather cloud interoperability and data portability" say cloud computing experts from IEEE (*Weissberger,* 2011). According to the above examples it can be stated that interoperability and cloud based manufacturing is really in the focus of the actual research topics.

Cloud-based manufacturing (CBM) make use of the cloud technology in the industry mirroring the service orientation approach of it by applying diverse cloud service- and deployment models that can easily convert and map manufacturing processes and assets into services.

The chapter, after a short overview on cloud manufacturing, provides a comparison of the most relevant features between traditional manufacturing IT systems and new architectures ones. Cloud technology is introduced in the next section with focusing on key fields of such architectures, such as cloud models, their combinations, and cloud interoperability. The forth section contains an overview on cloud manufacturing, highlighting its main characteristics, the different types, a short description of interoperability challenges and how to convert traditional manufacturing to CBM through virtualization. A basic conceptual model for cloud manufacturing (CMCM) is also proposed by the authors. IIRA and RAMI 4.0 are two of the most known standardization frameworks for Industrial Internet environments: aim of this manuscript is also to present the two architectures and highlight their integration compatibility and functional interoperability. As cloud architectures become the basis of most innovative and competitive industrial IT systems, the future role of CBM and IIoT in Cyber Physical Production Systems (CPPS) of the Industry 4.0 (or Smart Factory) has been discussed in the last section of this chapter.

MANUFACTURING ORGANIZATIONS, TECHNOLOGIES

In order to fulfil the actual market demands, production and manufacturing systems have been optimized in their structure, costs and fabrication technology. To be able to compare the different manufacturing systems a short overview is given, starting from traditional automated manufacturing system, towards the FMS (Flexible Manufacturing System), the networked, reconfigurable manufacturing systems (Virtual Enterprises), the CBM and finally the Industry 4.0 domain.
A short summary of the comparison showing the differences and the evolution of these manufacturing systems is shown in Figure 1. The qualification of the categories are not absolute but in comparison with the manufacturing system categories reported on the illustration.
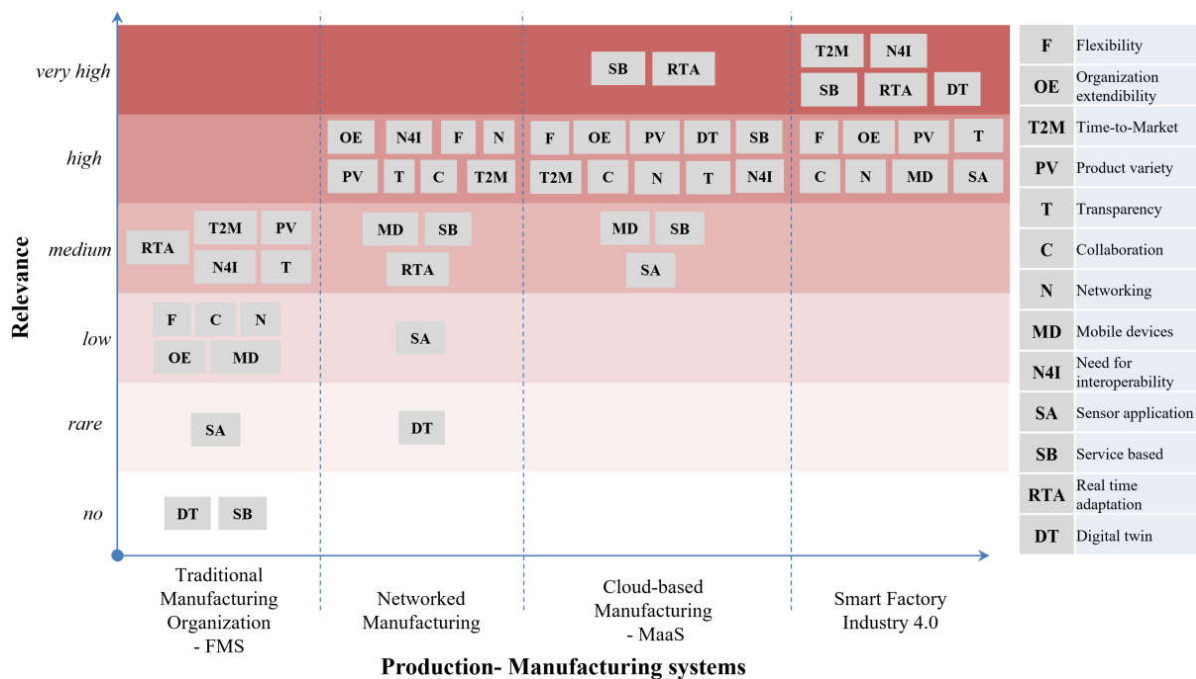
Relevance	Traditional Manufacturing Organization - FMS	Networked Manufacturing	Cloud-based Manufacturing - MaaS	Smart Factory Industry 4.0
very high			SB RTA	T2M N4I / SB RTA DT
high		OE N4I F N / PV T C T2M	F OE PV DT SB / T2M C N T N4I	F OE PV T / C N MD SA
medium	RTA / T2M PV / N4I T	MD SB / RTA	MD SB / SA	
low	F C N / OE MD	SA		
rare	SA	DT		
no	DT SB			

Legend:
F	Flexibility	MD	Mobile devices
OE	Organization extendibility	N4I	Need for interoperability
T2M	Time-to-Market	SA	Sensor application
PV	Product variety	SB	Service based
T	Transparency	RTA	Real time adaptation
C	Collaboration	DT	Digital twin
N	Networking		

Production- Manufacturing systems

Figure 1. Comparison of the different production and manufacturing systems

The FMSs is an integrated manufacturing system that has machines tools automatized on different levels, an automated material handling system and some type of automated store. The FMS is controlled by computers and can contain beside the control SW CAD/CAM and other CAX packages. The volume of production is medium, and the variety of parts is high. Because of the flexibly applicable optimization algorithms e.g. production efficiency, machine utilization could be improved while inventory, throughput time, waste could be reduced. The investment costs of an FMS were high. Different types of standards have been applied (e.g. in CAD/CAM, MAP) but in many cases the manufacturing units were connected with proprietary, "home-made" protocols.
The next step in the evolution was the distributed or networked manufacturing systems. The development of network technology made possible to connect machine tools, complex manufacturing and assembly systems both inside and outside a factory. Information technology (collaborative, agent, holonic, artificial intelligence) made possible to raise the level of collaboration, the efficiency, to reduce time-to-market and costs. The collaboration and cooperation are main characteristics of networked enterprises, so the contacts among the users, the machines and production units have outstanding importance.
The collaborative network paradigm has been developed and described in (*Camarinha-Matos & Afsarmanesh*, 2005) that covers the main characteristic of all different networked units that are autonomous, geographically distributed, and heterogeneous considering their operating goals, environment providing a framework to describe these organizations.
A collaborative network (CN) is a network consisting of different entities (e.g. organization units and humans), social capital and culture. The collaboration is supported by computer network and makes

possible to achieve common or compatible goals easier, thus generating joint value. The organizations participating in the network can be called as collaborative networked organizations (CNOs) that are introduced in (*Camarinha-Matos et al.*, 2009). The reliable exchange of data and information is critical among CNOs, so interoperability and standardization have a vital role.

In case making a systematic comparison between CBM and Smart Factory (instantiation of Industry 4.0 approach) it is clear in what fields are SFs more affective, what technological advantages it has. The concept of Industry 4.0 is broader then CBM where as an addition to CBM Industry 4.0 integrates the IT technologies of CPS and IoT. The main characteristics presented in Table 1 gives the differences of the two manufacturing approaches.

Table 1. Comparison of the main characteristics of CBM and Smart Factory

Factors of comparison	Cloud based manufacturing	CBM in Industry 4.0 – Smart Factory
Production services Flexibility	Limited to certain fields (HMIs are not general)	In all fields – general characteristics of the system (advanced HMIs)
Organization expandability	Limited by virtualized units	Easy to expand based on CPS technologies (virtual-physical objects)
Product variety	High	Very high
Time–to-Market	Short	Very short, immediate (3D print)
Networking	Partially wireless	Fully networked with all ICT technology types
Collaboration	Based on cloud deployment models	Lean structure (Included all devices, equipment, machines, SW, sensors)
Sensors/real-time adaptive management/control	Included machines, equipment and virtual services	Real time, wireless, among all devices and sensors + processes
Transparency	Transparency limited	High transparency
Service based	Basic technology	Basic technology
Virtual twins	Not existing	Basic technology
Application of mobile devices	Limited applications (e.g. communication between machines)	High level, all round applications
Need for interoperability	Many existing standards	Numerous solutions are missing (e.g. standards for M2M communication)
Security (endpoint, network)	Endpoint and communication risk can be limited by increased security	Higher risk because of M2M (IoT)
Basic technologies	Virtualization, cloud computing, service-based technologies	As in CBM + IoT, BigData, sensors, virtual simulation

Wu et al. (2013) present the main fields of cloud manufacturing applications together with their classes of user in a topics map. The applications cover the life cycle of CBM systems but there are some gaps in the system e.g. the human factor challenges. As CBMs fundamentally change work and work systems these parts of the integrated cooperative systems have to be extended (*Golightly et al.*, 2016). In SFs these problems are handled by advanced HMIs as basic services.

As cloud computing provides the core technical background for CBM and enables advanced concepts in manufacturing, such as networked and virtual manufacturing, the limitations are in connection with these fields. The real-time handling of big amount of data obtained during a manufacturing process (e.g. force, temperature), can cause problems in the operation as they are uploaded via Internet to the cloud where data storage and analysis are performed, and the results should provide real time control for manufacturing units (*Wang et al.*, 2015). So, response time, communication channel (data handling) bottleneck, and communication and data storage security can couse different limitations in cloud manufacturing as well.

Handling Industry 4.0 approach as a base on that "Smart Factories" (SF) are developed, in SF physical processes are monitored by cyber-physical systems and create a virtual replica of the physical world (virtual twin) and make decentralized decisions, while all devices are cooperating with each other (including humans as well) in real time using the Internet and IoT components. In SF the need for interoperability, for security are higher (keen communication among all devices), the continuous

collaboration of sensors, mobile devices and humans (through intelligent interfaces) in all fields and levels is significant (*Liu & Xu,* 2016). In Smart Factories BigData based analysis supported by artificial intelligence techniques are applied and the results of real-time virtual simulation (based on virtual twins) are also used.

INTEROPERABILITY IN NETWORKED ENTERPRISES

Interoperability can be defined as the ability of two or more systems or applications to exchange information and to mutually use the information that has been exchanged without special effort on the part of the customer. Interoperability is realized by the implementation of standards (ETSI SR 002 761). In the context of networked enterprises, interoperability refers to the ability of interactions (exchange of information and services) between enterprise systems.

There are different levels in the enterprise where interoperations can take place. In the ATHENA EU project, for example, four layers of interoperability concerns have been defined that can be applied also to networked enterprises (*ATHENA,* 2007). In Table 2 these layers are extended with activities and standards applicable on each level.

Table 2: Layers of networked enterprises for interoperability with activities and standards (Mezgar & Rauschecker, 2014)

VE Layers	Description of functions, activities	Standards
Business	Collaborative modelling, semantic interoperability, company culture	KIF, KQML, UEML 1.0
Process	Cross organizational business process, in NEs integrate different internal processes into a common one	PSL, UEML 1.0
Service, application	Flexible execution and service composition, identifying, composing and making various application functions together	UEML 1.0, APIs, STEP, EDI, HTML, XML, or eb-XML, J2EE, Java, .NET
Data	Information interoperability – to make query languages and different data models working together	UEML 1.0, XML, flat files, DB

There are three different main approaches to solve interoperability challenges in enterprises:

- **Integrated approach**. There is a common format for all models. This format must be agreed by all parties to elaborate models and build systems;

- **Unified approach**. The common format exists at meta-level and provides a means for semantic equivalence to allow mapping between models;

- **Federated approach**. No common format exists. Partners have to solve interoperability promptly and real-time, which means they have to share an ontology to map their concepts at the semantic level.

A detailed description of the above methods can be found in *Chen et al.* (2008), whereas trends and the issues for enterprise integration and interoperability in manufacturing systems are presented in detail in *Panetto and Molina* (2008).

CLOUD TECHNOLOGY, ARCHITECTURES

To date considerable efforts have been concentrated for a commonly accepted (if not definition, at least) vision of Cloud Computing (CC) and technology. Just like clouds may have different names if seen from different physical perspectives, also CC seems to fit in several definitions, derived from different application domains. It is commonly accepted that key benefits have to encapsulate both business and technological views. In general, cloud-based computing is an information technology (IT) architectural model where computing services (both hardware and software) are delivered to customers over the Internet, on-demand, in a self-service fashion, independent of device and location (*Marston et al.,* 2011). In other words, a CC system must enable, according also to the National Institute of Standards and Technology (NIST), a ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and

services) , which can be rapidly provisioned and released with (theoretically) minimal management effort or service provider interaction (*NIST,* 2011). The cloud model embodies unique characteristics and can have various service and deployment models, as reported in the section hereafter.

Service models

This is typically an end-user's perspective in CC industry, where different delivery models refer to different layers of the CC architecture.

The most common and used term perhaps is Software as a Service (**SaaS**), in which the application runs on the vendor's infrastructure and is recognized as a service. The provision is usually guaranteed through a thin client (a web browser) and the consumer is unaware of the application provider's infrastructure and complexity (examples of SaaS include Salesforce, Netsuite or Google Apps as enterprise-level applications; GMail, TurboTax Online, Facebook, or Twitter at personal level).

A Platform as a Service (**PaaS**) facilitates the development and deployment of applications without the cost and complexity of buying and managing the underlying hardware and software layers, like operating system, network, and servers, and development tools (examples of PaaS are Microsoft's Azure Services Platform, Salesforce's Force.com, Google App Engine and Amazon's Relational Database Services).

Infrastructure as a Service (**IaaS**) offers also storage, network and computational capabilities as a service. Amazon's S3 storage service and EC2 computing platform, Rackspace Cloud Servers, Joyent and Terremark are all major examples of IaaS. Refer to *Marston et al.* (2011) for a comprehensive list of key players in the CC industry.

Manufacturing specific Hardware as a Service (**HaaS**) is a standard or technique for consistently describing and serving equipment and its functionality, behaviour, structure, etc. It represents one of the major challenges in service implementation description of core physical equipment in the context of Manufacturing as a Service (**MaaS**).

Characteristics and deployment models

Essential properties of CC systems can be classified (also according to considerations reported by *Xu,* 2012; *Zissis & Lekkas,* 2012; *Wang & Xu,* 2013) into core-, business-, enterprise- and manufacturing specific. Resource abstraction, self-service-centricity, network access mechanisms, rapid elasticity, service measurability, multi-tenacity, load-balancing and virtualization are all core aspects in cloud computing. Business relevant properties comprise quality-of-service (QoS), service level agreement (SLA), user experience (UX), fault-tolerance, auditability and certifiability. When entering the enterprise level, interoperability, deployment models, security and business process management augment the complexity of cloud requirements. Cloud in manufacturing, finally, is expected to provide a solid support for service-oriented environments, simulations and global services.

Cloud hosting deployment models represent the category of cloud environment and are mainly distinguished by the proprietorship, size and access.

They are classified as follows:

- **Public cloud**: this is the major model of cloud computing, in which cloud owner provides Internet-based public services on predefined rules, policies, and a pricing model. Large number of widespread world resources lead to select appropriate resources while considering the QoS;

- **Private cloud**: designed and established to prepare most of the benefits of a public cloud exclusively for an organization or institute, such a system can lead to decreased security concerns, thanks to the utilization of corporate firewalls. Organizations implementing a private cloud are responsible for the entire system, eventually resulting in abundant costs;

- **Community cloud**: based on similar requirements, concerns, and policies, a number of organizations establish a community and share the CC to be used by their community member's consumers. A third-party service provider or a series of community members can be responsible for providing the required infrastructure of the cloud computing. Lowering costs, dividing expenses between community members, and supporting high security are the most important advantages of a community cloud;

- **Hybrid cloud** is a combination of two or more different public, private, or community clouds, which led to the creation of a different cloud model called hybrid cloud. Constitutive infrastructures require standardized or agreed functionalities in order to communicate with each other and interoperate on applications and data. Business- and mission-critical services and sensitive data are kept unpublished, while non-critical services are published for others to share and use.

Illustration on Figure 2 depicts common cloud deployment models with service-critical instantiation orientations.
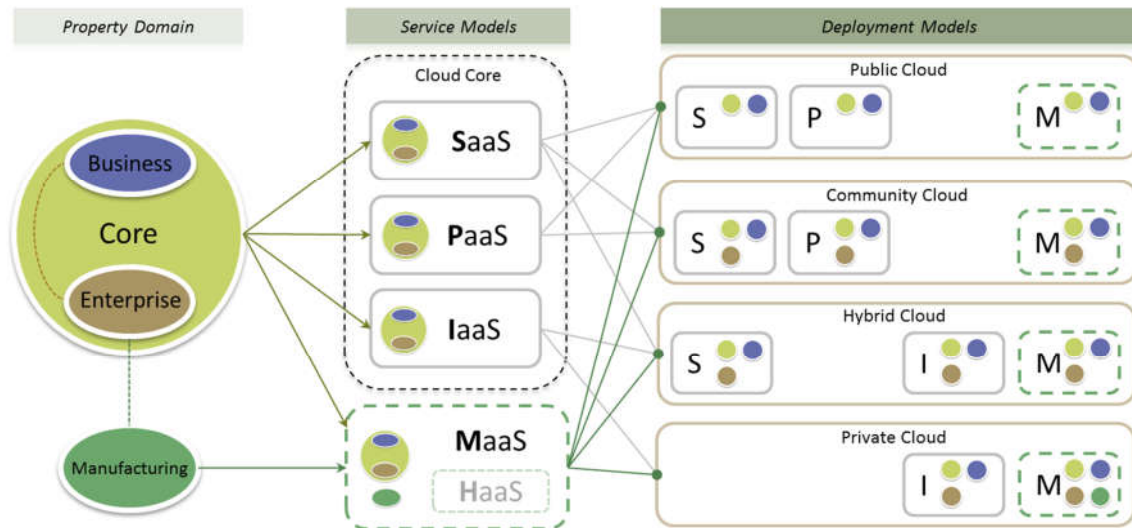
Figure 2. Cloud deployment models with property-specific service instantiation

Two fundamental concepts, directly relating to cloud physical level and deployment models, are migration and disaster recovery. Moving a virtual machine (VM) from of one cloud into another requires adequate ID management systems and, in case of truly hybrid cloud federation, standardization will be mandatory. Standards are crucial in VM description formats, common services and application descriptions: possible standards include OVF from DMTF, TOSCA from OASIS, OpenID, and Oauth.

Stakeholders

CC stakeholders include not only traditional technology roles but also regulators, brokers, certification authorities, auditors and others, as detailed in the following list:

- **Consumer**: effective subscriber, purchasing the use of the system from providers on an operational expense basis;

- **Provider**: delivers services to third parties and performs maintenance and upgrades. They are also responsible cloud-software maintenance and cloud-services pricing;

- **Enabler**: organizations selling products and services, facilitating the delivery and adoption of cloud;

- **Regulator**: sovereign government body or international entity pervading across the CC "value-chain".

- **Auditor**: chosen and entrusted by cloud consumers to conduct independent assessments on services, system operations, performance, and security of CC;

- **Broker**: entity that manages the use, performance, and delivery of cloud services, negotiating between providers and consumers;

- **Certification Authority**: placed between cloud-consumers and -resources, it validates consumer's encrypted connection, authenticates cloud-consumers, itself and the consumers to cloud-providers.

Major recommendations

Next to the perceivable benefits of cloud computing, there are still significant barriers and recommendations to be taken into account for its adoption (*Zissis & Lekkas*, 2012; *Lee*, 2008). See Table 3 for details. Mission-critical services of enterprises need to be locally reinforced, so to ensure continuity in the execution and, therefore, provision of business processes. Consulting companies reported how current cloud services may not be cost-effective for larger enterprises which have achieved best efficiencies from their own computing infrastructure (*Bommadevara et al.*, 2016). Different is the situation for small and medium enterprises (SMEs), which usually do not have the initial capacity to set-up secure, large data centres, and for which prices and SLAs from cloud providers are far more advantageous. Unfortunately, many cloud providers still offer SLAs with rather weak user compensations on outages (measurement of service delivery, method of monitoring performance, and amendment of SLA over time). Finally, some hidden costs need to be carefully balanced, including support, disaster recovery, application modification, and data loss insurance.

Table 3. Barriers and issues related to cloud adoption

Issue	Relevance
Security and privacy	The very first concern CC has to adequately address: privacy regulations (*Zissis & Lekkas*, 2012; *Wei et al.*, 2014) and privacy protection of individuals and business.
Connectivity	The full potential of CC depends on the availability of high-speed access to all.
Reliability	Enterprise applications are so critical that they must be reliable and available 24/7 and recovery plans must take effect smoothly in case of failures or outages.
Physical control and boundaries	Organizations justifiably wary of the loss of physical control over data on the cloud. Providers must be able to guarantee the location of a company's information on specified set of servers in a specified location (legal involvements).
Mission-criticality	Cloud providers still cannot commit to the high QoS and availability guarantees demanded in large organizations. Service Level Agreement (SLA) commitments for annual uptime have different acceptance levels for SMEs but are still insufficient for mission-critical applications in large organizations.
Interoperability	Interoperability and portability of information between private and public clouds are critical enablers for adoption of CC in enterprises: they permit integrity and consistency of company information and processes.

CLOUD STANDARDIZATION

Standardization plays a huge impact on cloud adoption and usage. Cloud service providers usually have their own approach on interactions and Cloud APIs required to users. Complex business applications on the cloud require adequate standards, as the lack of integration between networks makes it difficult for organizations to consolidate their IT systems in the cloud and realize productivity. When writing this manuscript more than 15 different groups, committees and organizations have established a Wiki site for Cloud Standards Coordination (*Cloud-standards*, 2017), whose goal is to document activities from the various Standards Development Organizations (SDOs) and the leading technology.

Table 4 gives an overview on main activities and standards on cloud-service layers.

Table 4. Cloud service layer activities and standards

Target	Activity description	Standards
Client	User interface and hardware mobility	Proprietary standards
Application services	Business processes, social networks, collaboration	Not cloud-specific: XML, SOAP, ebXML, OASIS SCA, SDO, SOA-RM, and BPEL
Application data	Data portability among cloud services	JSON, XML
Run-time	Application developments and testing	J2EE, .NET
Middleware	Customers software modules within the cloud	Not cloud-specific
Operating System	OS drivers transparent management in SaaS/PaaS	OCCI
Storage	Data storing	CDMI, CADF

Generally, physical hardware virtualization (ARL) follows the Open Virtualization Format, while PRL do not evidence the use of any specific standards.

Containers technology and Serverless Computing

Containers represent next step in cloud evolution. Container technology wraps software in a complete filesystem, containing everything needed to run (called image): code, runtime, system tools, and library dependencies. This guarantees that software will always run the same, regardless of its environment: Build Once and Run Anywhere (BORA). Containers main characteristics are: open-source nature, kernel-level application isolation, virtualization replacement, application closing, closed-tiered containers, and centralized storage of containers descriptors. Deployments and relocation to new environments occur by means of so called images. Docker.com containers, for example, one of the most popular providers, are based on open standards, enabling containers to run on all major Linux distributions and Microsoft Windows, as well as on top of any infrastructure. Containers are isolated but, with respect to traditional Virtual Machines, share OS and (when appropriate) bins and libraries (Figure 3).
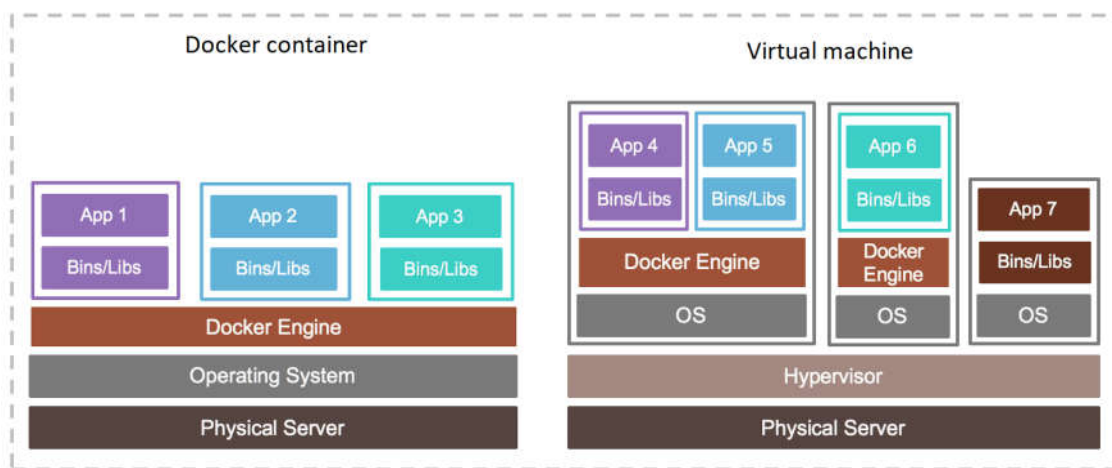

Figure 3. Docker container vs. Virtual machine architecture

Serverless Computing, or *Function as a Service* (FaaS), is a form of cloud-based computing similar to VMs and containers running on a cloud provider. This does not mean there are no servers, but instead that the management of servers, scaling, and capacity planning are taken care by the underlying cloud provider. Application developers only need to focus on functionality and business logic.
Before Serverless computing, many enterprises adopted micro-services, a form of Service Oriented Architecture (SOA). Micro-services enabled applications to be organized as a collection of loosely coupled services connected together through APIs. Each service is an entirely separate mini

application in its own process/container/VM. The main benefit is modularity and separation of concerns. However, with the advent of micro-services, infrastructure and operations work have greatly increased. Many more continuous integration/continuous delivery pipelines needed to be tracked and there is complex orchestration to manage many more architectural pieces. Logging context is scattered across many individual processes and much more effort is placed on integration testing. Companies like Basecamp[1] argues that monolithic architecture can make sense for certain small companies like start-ups.

On the other hand, with Serverless computing the infrastructure, orchestration layers, and deployment are taken away. There are still servers and VMs, but they are fully managed by the cloud provider. Application developers only have to write business logic and functionality and leave the rest to the cloud provider.

Serverless computing can reduce computational costs, also. While most cloud providers will charge an hourly rate for reserving a VM, Serverless computing can use a consumption-based pricing model and there is no charge if the application isn't actively using compute or memory resources.

There are several providers of Serverless computing, like IBM Cloud Functions, Webtask from Auth0, Iron.io, but for simplicity, we report the comparison of major aspects only for the biggest three (Table 5): AWS Lambda, Azure Functions and Google Functions.

Table 5. Comparison of biggest FaaS providers

Target	Activity description	Standards	Google Functions
Languages support	Node.js, Java, C#, Python	Node.js, C#, F#, Python, PHP, Java	Node.js
Languages support with 3rd party	Golang by using Node.js shim	With batch files can run anything	
Monitoring	Cloudwatch, Dashbird	Azure Application Insights	Stackdriver Monitoring
Pricing	$0.20/million requests with 1 million requests per month for free	$0.000016/GBs, 400,000 GBs/month for free $0.20/million executions, with 1 million executions/month for free	$0.40/million invocations with 2 million invocations for free $0.0000025/GB-sec with 400,00 GB-sec/month for free $0.0000100/GHz-sec with 200,000 GHz-sec/month for free
Limits	Memory allocation range: Min. 128 MB / Max. 1536 MB Ephemeral disk capacity ("/tmp" space): 512 MB Maximum execution duration per request: 300 seconds	Allows only 10 concurrent executions per function No limitations on max. execution time limit	Number of functions: 1000 Max function duration: 540 seconds Function calls per second: 1,000,000 per 100 seconds

Most Serverless applications have to adhere to an Event-Driven Architecture (EDA): this allows for a more responsive application because systems are by design asynchronous at an unpredictable scale. In terms of security, some aspects are handled by the FaaS operator, but this doesn't assure that a specific application will be free of security issues. All the advantages/disadvantages of FaaS are briefly summarized as follows.

---

[1] https://basecamp.com/

Advantages:
- Management of infrastructure - buying and configuring servers is costly in terms of initial investment and specialized staff required;
- Security of infrastructure - users do not need to worry about Linux, Tomcat, etc. updates;
- Easy deployment - developers don't have to wait for OPS, DBA, etc.;
- Scalable & HA - Amazon, Microsoft and Google are better at scaling than anything most of users can hire;
- Costs – users pay only for what resources they use.

Disadvantages:
- Latency - FaaS adds some latency so for a high-performance application it might not be the best idea to use FaaS;
- Limits - memory (1500MB on AWS), execution time (300 secs on AWS, 500 secs on Google);
- Monitoring & debugging - there are some solutions that are maturing and allow for local or offline debug/test, but at this moment it is still a limitation;
- No local stored data - your application has to be stateless, so for most of them this is really a good thing but it is a limit nevertheless;
- Vendor lock-in – users depending on, for example, AWS, Azure, need to a have a recovery, exit strategy if their vendor went bankrupt in few years.

CLOUD BASED MANUFACTURING

The architecture of the organizations is in a recursive connection with the IC systems; the IC technology offers new possibilities for restructuring the organization (and its business processes) itself, in other cases the new demands of a business process force the development of a special IC solution. The final goal of all information systems is to provide secure data-, information-, knowledge-, or different services for the users (human beings), and for firms, enterprises (*Mezgar & Rauschecker*, 2014).

The idea of Cloud based manufacturing has been developed based on the cloud computing technology (*Li et al.*, 2010). The basic idea was to apply cloud computing architecture, tasks and service structure to manufacturing systems. The result of this mirroring was a new service-oriented networked manufacturing model called Cloud Manufacturing (CMfg). Since the first basic publication numerous papers have been written with different approaches to the theme. The name of the paradigm varied from "Manufacturing Cloud - MCloud" (*Zhang et al.*, 2010), to "Cloud Based Manufacturing - CBM" (*Wu et al.*, 2014) but the content behind the words is the same.

The reason of the novelty of CBM lays in multi-tenancy and virtualization; these are the two main characteristics that make difference between CBM and networked manufacturing. According to Wu (*Wu et al..*, 2013) "the differences and similarities between CBM and web- and agent-based systems will be articulated from a number of perspectives, including (1) computing architectures, (2) data storage, (3) operational processes, (4) information and communication, (5) business models, and (6) programming models". As the differences show in every category significant positive output for CBM the closing statement of the analysis was; CBM is a new paradigm.

The definitions of cloud manufacturing also reflect how this paradigm expanded in the last years (*Zhang et al.*, 2011). There are numerous definitions of cloud manufacturing their content depend on the main approach of the application field. Some examples are reported as follows:

"Cloud manufacturing is a computing and service-oriented manufacturing model developed from existing advanced manufacturing models (e.g., application service providers, agile manufacturing, networked manufacturing, manufacturing grids) and enterprise information technologies under the support of cloud computing, the Internet of things (IoT), virtualization and service-oriented technologies, and advanced computing technologies" (*Li et al.*, 2010).

"Cloud manufacturing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable manufacturing resources (e.g., manufacturing software tools, manufacturing equipment, and manufacturing capabilities) that can be rapidly provisioned and released with minimal management effort or service provider interaction" (*Xu*, 2012).

Qanbari et al. (2014) define "Cloud manufacturing is a distributed manufacturing execution model, where underlying resources envisaged in the internet of things are elastically exposed and utilized as cloud services, then composed and orchestrated for utilization as cloud services, then composed and orchestrated for a manufacturing task in an on demand fashion".

According to Ren and his co-authors "Cloud manufacturing is a smart networked manufacturing model that embraces cloud computing, aiming at meeting growing demands for higher product individualization, broader global cooperation, knowledge-intensive innovation and increased market-response agility" (*Ren et al.*, 2014).

The main advantages of CBM for the manufacturers are:

- On-demand leasing/releasing of manufacturing assets

- Flexible reconfiguration of manufacturing assets according to the needs,

- "pay-as-you-go" business model based on the measured really used service.

The advantages are really attractive so, there are three main areas where cloud computing can be applied in manufacturing companies.

- Manufacturing software is supported as a service in the manufacturing cloud. This can be handled as the manufacturing version of cloud computing.

- Inter-factory collaboration that has an extended scope; services like supply chain visibility, transportation management, supplier/contract negotiation. Partners can create cloud computing modules to address other manufacturing issues, e.g. supply chain execution, shop floor planning, demand planning and production scheduling.

- High performance computing that use digital models to (1) virtually test the products or manufacturing system, (2) understand the business environment better through business intelligence and (3) make decisions. The models are typically highly parallelizable and fit well for a cloud environment.

Standardization and interoperability solutions have of vital importance in order to be able to use effectively cloud technology in the above applications. The control hierarchy in industry in most cases is based on the well-known ISA-95 standard (*Andrew,* 2015). On top-level can be found the ERP (Enterprise Resourse Planning), one level below the MES (Manufacturing Execution System), then the SCADA (Supervisory control and data acquisition) takes place. In Figure 4 it is shown how conventional manufacturing tasks/processes in ISA-95 layers are converting to cloud services.

Virtualization is the most important phase of transforming a conventional manufacturing system to cloud manufacturing. Virtualization means to dynamically divide different resources into virtual units and later combine them flexibly (with other resources) to a logic unit to meet diversified demands as encapsulated services. The conversion of conventional manufacturing to cloud manufacturing has three phases; (a) identification of manufacturing resources, (b) virtualization of these resources and (c) grouping the virtualized resources into cloud manufacturing services.

The quality of virtualization determines the robustness of a cloud infrastructure. Good virtualization can effectively assist sharing of cloud facilities, managing of complex systems, and isolation of data and application.

Cloud manufacturing virtualization is more complex compared to cloud computing virtualization as both computing and manufacturing resources have to be virtualized. As manufacturing system organizations have dynamic characteristics and high uncertainty in their operations a flexible mapping strategy has to be applied. A manufacturing system can be virtualized in different levels/layers, e.g. capacity of a job shop vs. capability of a machine tool.

Different virtualization approaches have been described in details e.g. in *Zhang et al.* (2017) a service encapsulation and virtualization access model, in *Ning & Xiaoping* (2012) a cloud manufacturing virtualization framework has been described, that contains three layers: manufacturing resource layer, virtual description layer, and service encapsulation layer.

Manufacturing resources and capabilities can be shared over a manufacturing cloud platform as cloud services after having been virtualized and encapsulated.
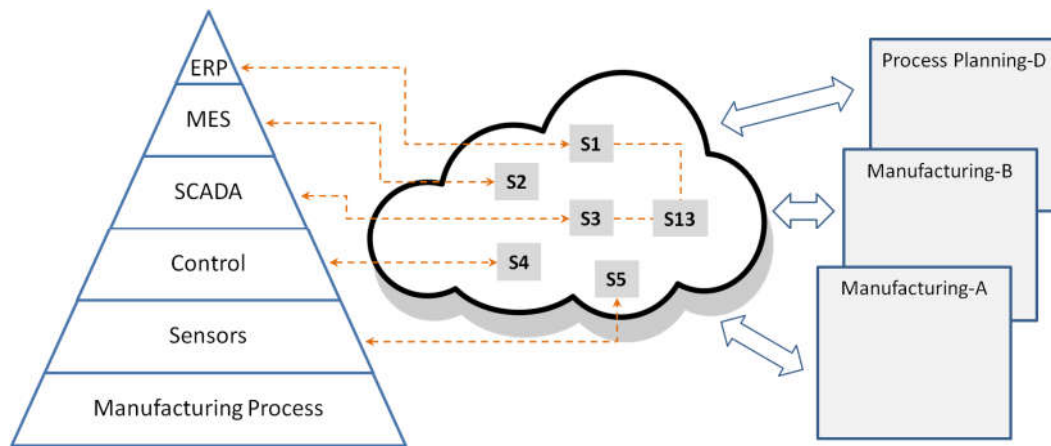
Figure 4. ISA-95 layers converting into cloud

INTEROPERABILITY IN CLOUD MANUFACTURING

Interoperability is an essential requirement for both service providers and enterprises. Services with interoperability allow applications to be ported between clouds, or to use multiple cloud infrastructures before business applications are delivered from the cloud. Interoperable solutions make easy migration and integration of applications and data between different cloud service providers (*Xu*, 2012). Interoperability can be divided into two main groups in cloud-based systems/applications:

- The interoperability of the cloud systems itself (cloud layer), and

- The interoperability of the applications (Application layer – industry specific, HW/SW specific).

In the field of cloud based manufacturing numerous solutions, systems have been developed with different focus fields. *Yadgarova and Taratukhin* (2016) introduced an integrated framework for building cloud-based manufacturing environment that allows to develop future production systems as the class of adaptive distributed systems with virtual cloud model and simulation. A dependency model for equipment interaction has been defined too.

Mourad et al. (2016) identified interoperability as "a key enabler for cloud manufacturing". They proposed a framework called "C-MARS" for realisation of interoperability across heterogeneous computer aided manufacturing systems (handling CAD and NC files). Using this framework, manufacturing resources can be shared by a large number of clients based on requirements and priorities.

A service-oriented, interoperable Cloud manufacturing system is proposed in *Wang & Xu* (2013). Service methodologies have been developed to support customer and enterprise users, along with standardized data models describing Cloud service and relevant features. By using ICMS, standardized (STEP-based) communication methodologies have been deployed to support collaborative interactions in the Cloud environment.

In the next chapter a general cloud manufacturing conceptual model (CMCM) is proposed by the authors, that applies among others the container technology.

Cloud Manufacturing Conceptual Model

CM is the necessity for manufacturing enterprises to be described, realized, componentized, virtualized and integrated in an interoperable manufacturing cloud. As already mentioned, major research works by *Wang and Xu* (2013) propose service oriented architectural solutions, highlighting how the need of implementing a mechanism to organize and control cloud services at upper-level is of crucial importance. Based on research projects developed so far, the authors isolated and derived a general CM conceptual model (CMCM, illustrated on Figure 5), which encompasses the following elementary classes of actors:

- **Cloud Provider** (CP): responsible for providing platform services both to internal components and external stakeholders. Main functionalities encompass the management of:

applications patterns deployment, cloud adapters, cloud simulations workflow, privacy and security policies, virtual machines (VMs) and Docker containers;

- **Cloud Consumer** (CC): the node receiving output for service requests within the platform (i.e. digital factory model management, big data storage, applications ontology management and maintenance) through the CMCM service orchestrator;

- **Cloud Certification Authority** (CCA), **Cloud Broker** (CB) and **Cloud Auditor** (CA) have already been introduced before.

CMCM architectural back-bone should provide services through a standardized, homogeneous, application-level (i.e. HTTP) interface (C-API).
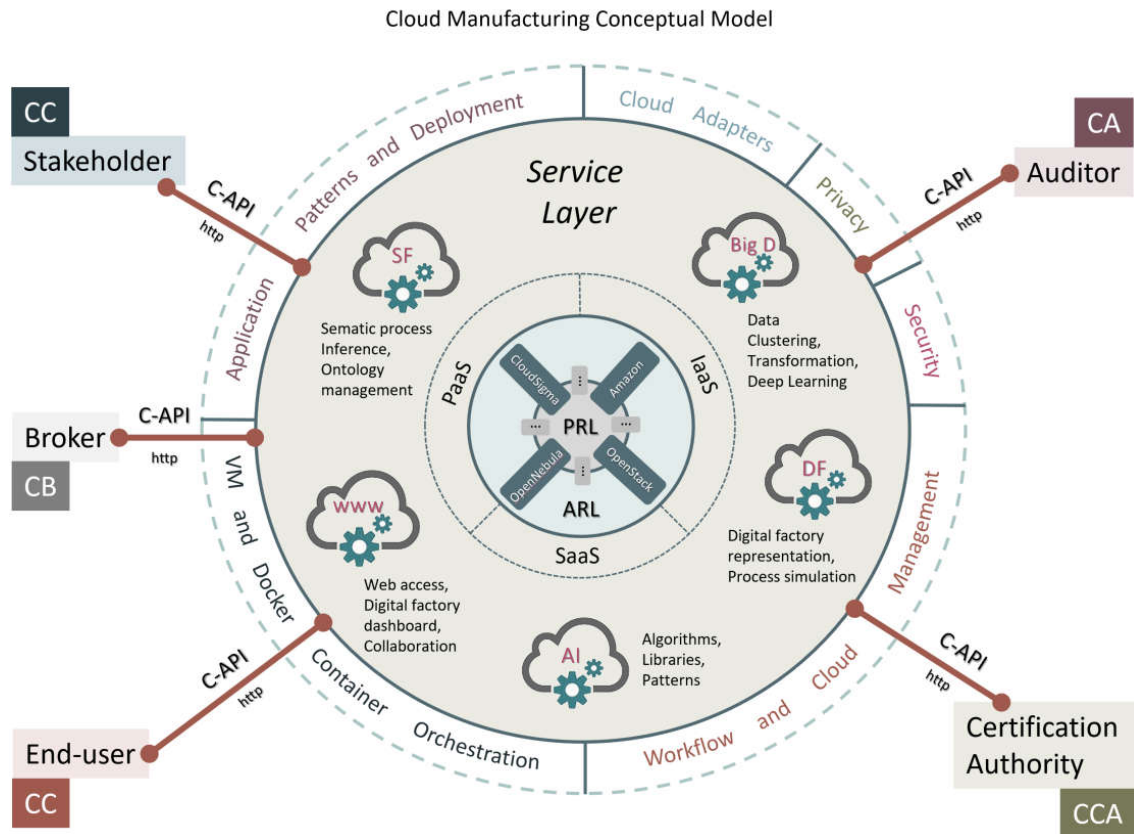
Figure 5: Cloud Manufacturing Conceptual Model

Major components of CMCM architecture are as follows:

- **Service Layer** (SL): conceptual abstraction boundary of CMCM functionalities. It enables the creation of complex virtual services, like simulation workflows;

- **Abstract Resource Layer** (ARL): cloud-specific back-end layer managing the dynamic allocation, bounding and routing of resources to the underlying physical level (PRL). Platform users register their applications and create VMs leveraging also supported external clouds (e.g. Amazon, CloudSigma, OpenStack or OpenNebula);

- **Physical Resource Layer** (PRL): dynamically discovers and assigns resources to ARL. For instance, CPS elements have corresponding PRL layers running over the physical resource, eventually exploiting manufacturer-specific APIs for specific functions.

SL main components can be classified as follows:

- **Semantic Framework** (SF): services related to the semantic description of CMCM resources with their orchestration and management, and the inferential discovery of knowledge;

- **Big Data** (Big D): noSQL-compliant data collection at CPS level. Specific mechanisms enable data processing and aggregation, reducing the volume and communication traffics;

- **Web applications** (WWW): complex functionality (like simulation workflows) provided to end-users by means of higher-level web-based solutions;

- **Cloud Application Programming Interface** (C-API): abstracts and standardizes access to cloud-services, by means of well-defined provision technology and protocols (micro REST services, Web-services or OPC-UA).

SERVICE ARCHITECTURE INTEROPERABILITY IN INDUSTRY 4.0

The integration of new technologies produces new organization structures in industry, and the frameworks, platforms, standards used so far have to be modified: the new manufacturing architectures that apply IoT, cloud, and mobile technology cannot be handled by rigid architectures. The control hierarchy in industry is fundamentally based on ISA-95 standard (developed on Purdue model) but the modern control and communication systems need a lean structure. As seen, smart connected assets communicating directly with each other are requiring the development of new architectures, and existing standards concepts can act as the basis of new standardization evolutions. Service composition is one of the most effective approaches investigated by IT researchers and applied by cloud-technology providers so far (a comprehensive literature review on cloud-service composition can be found in (*Jula et al.*, 2014).

The selection of appropriate services from service pools, management of their composition, specification of QoS parameters, understanding of requirements dynamic and rapid changes in service-provision interface are just some of the important issues addressed to meet end-users' satisfaction. Service orchestration techniques were first applied in CC systems in 2009 (*Cheng et al.*, 2009). Because of the exuberant growth of offered services, cloud-service brokers are facing high competition in providing quality of service. Such competition leads to difficulties in designing services in a way that their selection, orchestration, deployment and management be suitable in the cloud environment. Aspects like selection of appropriate services from service pools, composition restrictions, and substitutability of services due to emerging changes in requirements and network seem to have the higher impact on service provision. Investigations in *Jula et al.* (2014) demonstrate that CC service-orchestration can be categorized into five major groups: classic and graph-based, combinatorial, machine-based, structural and with frameworks.

With proper IT methodological approaches companies can create flexible business applications through elementary service composition: this is the essence of Service Oriented Architectures (SOAs), where services find a common approval both in business and IT domains, as they are considered as repeatable, independent, self-describing, business tasks or modules for externalized service invocation. SFs aim at integrating heterogeneous systems, by interlinking IT infrastructures, cloud-platforms, CPPSs and business processes through services. A comprehensive introduction to SOA-applied manufacturing (SOAm) can be found in SOA Manufacturing Guidebook of *MESA International and IBM* (2010). This architecture allows for the creation of composite business processes from independent, self-describing, and interchangeable code modules called services, arranged together using process choreography and used via a services bus. Core of MESA-IBM SOA is the Enterprise integration Service Bus (ESB), which assures the operations between service providers and requestors: routing of messages, conversion of transport protocols, transformation of message formats and handling of business events from different sources.

Based on SOA and MESA-IBM ESB (*Fraunhofer IPA*, 2016a), Fraunhofer IPA has developed an integration model built upon the cloud platform "Virtual Fort Knox", for networking factories (*Fraunhofer IPA*, 2016b). The platform is equipped with a homogeneous, service-based integration layer called Manufacturing Service Bus (MSB). MSB connects cyber-physical manufacturing systems to digital services via encrypted channels and acts as a multi-functional digital bridge between the cyber-physical environment and the digital tool. It integrates all IT-specific production control systems, including strategic and analytical business services, by means of routing, data transformation and orchestration services.

Standardization

The global networking of production resources and processes, as well as the use of globally interconnected applications crucially require the adoption of uniform standards. Two major proposals are hereafter presented, which are currently "de-facto" considered as reference guidelines in conceiving future business organizations: the Reference Architecture (IIRA) for Industrial Internet Systems (IISs) and the Reference Architecture Model for Industry 4.0 (RAMI 4.0).

IIRA

IIRA is a standard-oriented open architecture for IISs (*IIC Working TWG & SWG*, 2015), which aims at extending industry applicability and interoperability, and guiding technology standards development. IIRA architecture is generic-purposed and promises high level of abstraction to support broad industry applicability. IIRA abstracts common characteristics, features and patterns derived from use cases defined in the Industrial Internet Consortium (IIC). The architecture framework is based on ISO/IEC/IEEE 42010 (*ISO/IEC/IEEE*, 2011) standard specification, and codifies conventions and common practices for internet oriented architectural concepts, like **concern** (topics of interest in the system), **stakeholder** (entities having an interest in the system) and **viewpoint** (conventions framing the description and analysis of specific system concerns). Concerns of IIS are classified and grouped in four different viewpoints: business, usage, functional and implementation. This paper primarily focuses on the functional and implementation viewpoints. In functional viewpoint, IIS is decomposed into five sub-domains: control, operations, information, application and business. These domains represent the building blocks of an IIS and illustrate how data and control move across them. The implementation viewpoint describes the general architecture, the technological components of an IIS, and the interfaces, protocols and behaviours among them. Popular implementations are encapsulated under various architectural patterns, all including the three-tier and gateway-mediated edge connectivity patterns. IIRA is not a standard but provides guidelines on how a safe, secure and resilient IIS can help realize the vision of the Industrial Internet (II).

Figure 6 illustrates the three-tier architecture and functional domains of IIRA:

- **Edge tier**: responsible for collecting data from edge nodes, using the proximity network. Distribution, location, governance scope and nature of the proximity network vary depending on the specific use cases. The edge tier (comprising asset, sensors and gateways) implements the control domain;
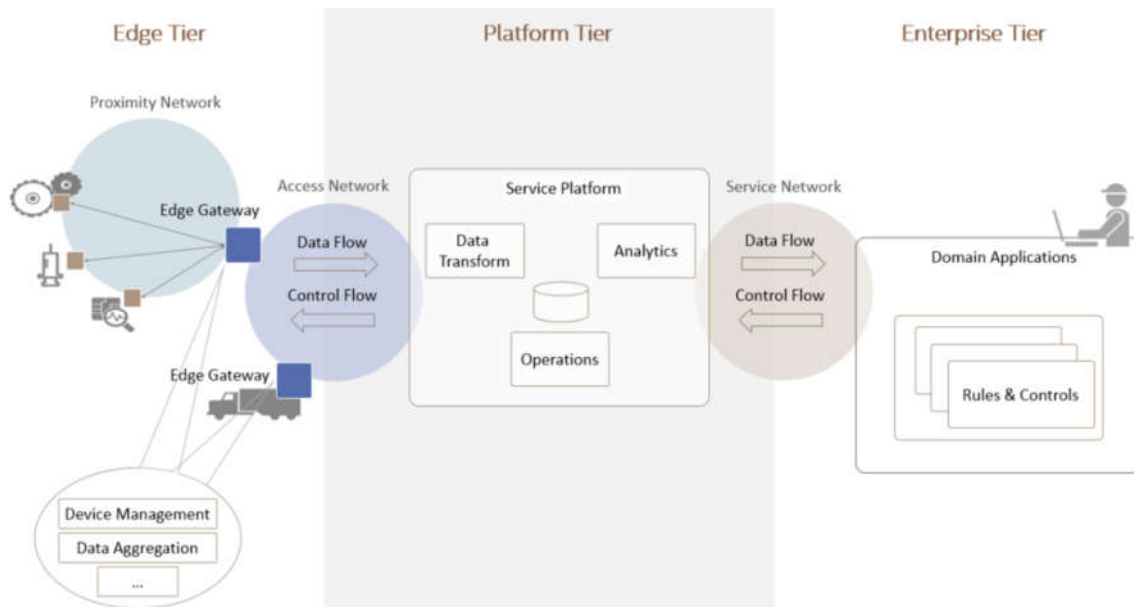

Figure 6. Mapping between three-tier architecture and functional domains in IIRA

- **Platform tier:** receives, processes and forwards control commands from the enterprise tier to the edge tier. It consolidates processes and analyses data flows from the edge tier and other tiers. It provides management functions, data query and analytics for devices and assets;

- **Enterprise tier:** implements domain-specific applications, decision support systems and provides interfaces to end-users, receiving data flows from the edge and platform tiers and originating control commands to the platform and edge tiers. It implements domain specific functionality such as MES, SCM and ERP;

- **Proximity network:** connects the sensors, actuators, devices, control systems and assets to a gateway that bridges to other networks, and enables data and control flow between the edge and platform tiers using typically XMPP/TLS;

- **Access network:** enables connectivity for data and control flows between the edge and the platform tiers (corporate network, or an overlay private network over the public Internet, or a 4G/5G network);

- **Service network:** enables connectivity (usually using TLS protocols) between services in the platform and enterprise tiers (an overlay private network over the public Internet or the Internet itself). A possible alternative may be utilize OPC UA between the various assets of the enterprise.

RAMI 4.0

The final goal of I4.0 is an interconnected factory capable of producing highly customizable products, realized through flexible mass production. Central in RAMI 4.0 is the concept of CPS – analogous to the IIS in IIRA – where autonomy is localized and participating systems make decisions on their own. The reference architecture model for RAMI 4.0 (*VDI et al.*, 2015) is the convergence of multiple (stakeholders) visions on how I4.0 might be realized, and it is built upon existing communication standards and functional descriptions.

The six layers of the vertical axis define the nature of IT components in I4.0: business applications, functional aspects, information handling, communication and integration capability, and ability of the asset to implement I4.0 features. This layered architecture basically breaks the complexity into manageable parts. The life-cycle of products, machines, orders and factories are captured along the life cycle and value stream axis (IEC 62890), whereas the hierarchy levels represent various functions of enterprise IT and control systems (IEC 62264 and IEC 61512). An important feature of RAMI 4.0 is the identification of objects as instantiated types: an object can be a product, asset, software, machine, or even a factory.

Industrial Interoperability: IIRA versus RAMI 4.0

Interoperability of systems can be ascertained at various levels in the enterprise: technical (communication protocols), syntactic (data formats and communication protocols), semantic (automatic interpretation of data and results), conceptual (fully specified but implementation independent model), business (value creation through cooperation with business partners and IT-supporters). Interoperability helps business stakeholders obtain higher-level system performances and benefit from the cross-implementation of industrial internet architectures.

Three-tier architectural pattern of IIRA shares distinctive features associated with RAMI 4.0 layered architecture, especially when considering interoperability at the functional level. Fundamental to IIC is the concept of **digital twin**, the computerized, often cloud-based counterpart of physical assets, which uses data from sensors, actuators, power supply and network interfaces installed on physical objects to represent their near real-time status, working condition (properties and states) or position (typical example is the use of 3D modelling to create a digital companion of the physical objects to be projected into the digital world) and efficiently schedule predictive maintenance of the asset. An important aspect of the digital twin is that it can be adapted for different environments to take advantage of data generated by other virtual machines in the ecosystem, according to different designs and requirements.

The central concept in RAMI 4.0, on the contrary, is built around the I4.0 compliancy of components, such as product, asset, software, or machine that be, referring to as objects that have the ability to

communicate independently, by using I4.0 compliant communication. Non-I4.0 compliant components are made I4.0 compliant by deploying an Administration Shell (AS), which essentially provides a virtual representation and a description of the entire life-cycle of the object or asset.
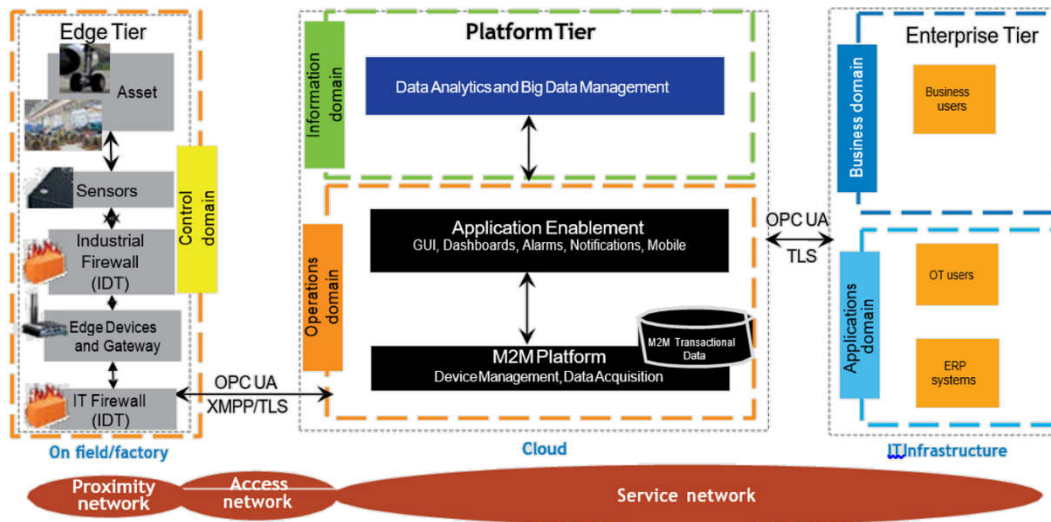

Figure 7. IIRA 3-tier architecture for IIC IDT and AE testbeds, functional domains from IIRA, and communication networks (*IIC & Plattform Industrie 4.0*, 2017)

It corresponds to the IIC "digital twin" of an asset in the I4.0 domain, and contains its lifecycle, technical functionality, and even procedures for sensor data integration and monitoring. In experimental results from *IIC & Plattform Industrie 4.0* (2017), authors demonstrated how Industrial Digital Thread (IDT) and Asset Efficiency (AE) (two paradigms for realizing testbeds in connected industrial organizations, *Infosys and IIC* (2015a; 2015b) can be applied to evidence a mapping between IT layers in RAMI 4.0 and functional domains under IIRA. Both IIRA and RAMI 4.0 stipulate the need for a SOA encapsulation of functionalities into services.

Illustration on Figure 7 shows IIRA 3-tier architecture for IIC IDT and AE testbeds, along with the functional viewpoints, the associated functional domains from IIRA, as well as all relevant communication networks associated with each tier. Mapping between the IIRA 3-tier functional viewpoints with the IT layers associated to RAMI 4.0 architecture for IDT and AE testbeds are reported on Figure 8.
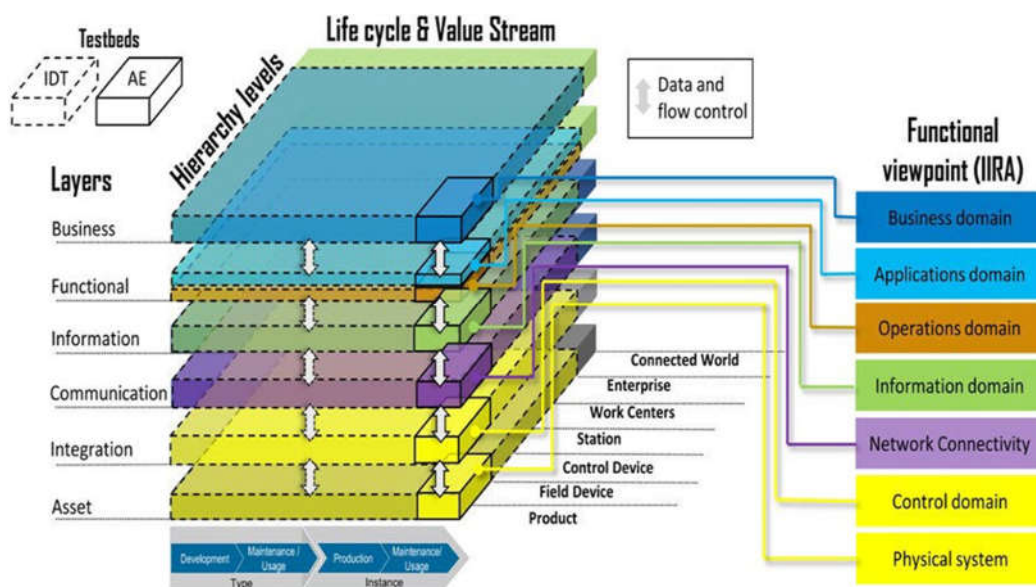

Figure 8. Mapping between IIRA 3-tier functional viewpoints with IT layers associated to RAMI 4.0 for IDT and AE testbeds

20

IIoT solutions, such as testbeds relying on a SOA, can be considered to be semantically interoperable at a functional level between the two architectures.

OPC UNIFIED ARCHITECTURE

Both IIRA and RAMI 4.0 do not indicate any specific solution for developing the infrastructural layer underlying functionality and communication and it is here that OPC UA can play a strategic role for the realization of the service interoperability asset in industry. OPC UA is a product of the OPC Foundation (*OPC UA*, 2017), an industry consortium which creates and maintains standards for open connectivity of industrial automation devices and systems.

The way OPC UA enables interoperability is by mapping concepts between property sets from different physical domains and/or frameworks: the more aligned is the conceptual modelling of functions (or services) the more natural will be the mapping between referenced concepts and their direct use.
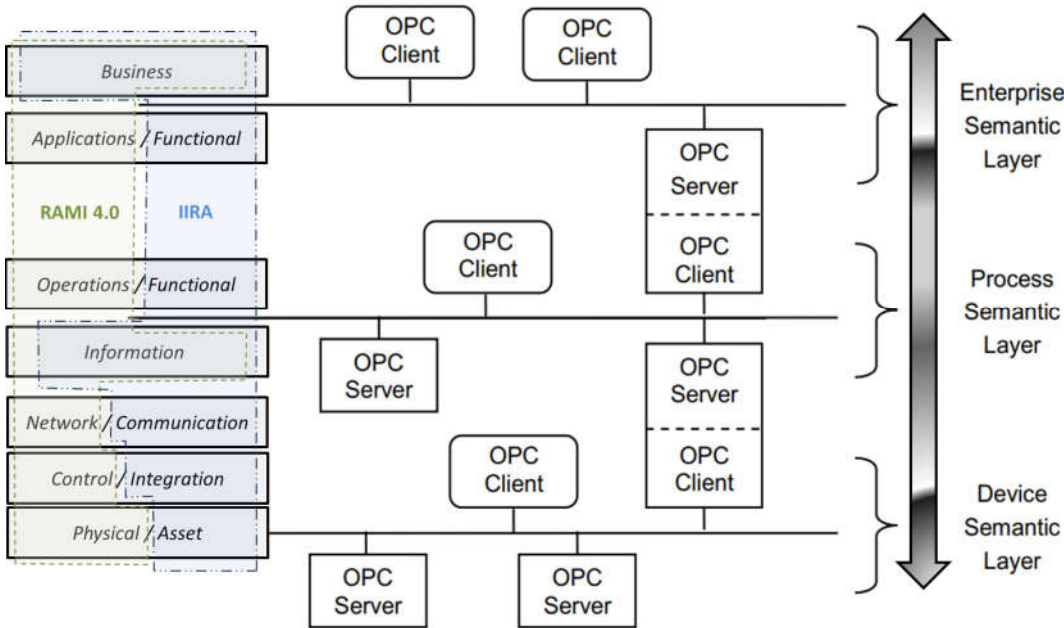

Figure 9. Example of chained OPC UA servers projected on RAMI 4.0 and IIRA layers

OPC UA, as a physical modelling environment and a communication standardizing architecture, can be leveraged to make different layers of both industrial architectures compliant with I4.0 guidelines, as it makes, for example, application testbeds semantically interoperable across RAMI 4.0 and IIRA architectures (Figure 9). For instance, it is interesting to note how RAMI 4.0 and OPC UA show modelling similarities in the specification of the following concepts: assets in vs. nodes; asset properties vs. object attributes and variables; sub-models vs. object methods (services) composition and orchestration.

An example of effective ongoing initiative demonstrating the integration compatibility of OPC UA into RAMI 4.0 AS is the Open Asset Administration Shell, an Open Source Project (openAAS) which aims at implementing an OPC-UA-based AS (expected by the end of 2017).

OPC UA is a service-oriented, platform-independent standard through which different systems and devices can communicate by sending messages between clients and servers, over various types of networks.
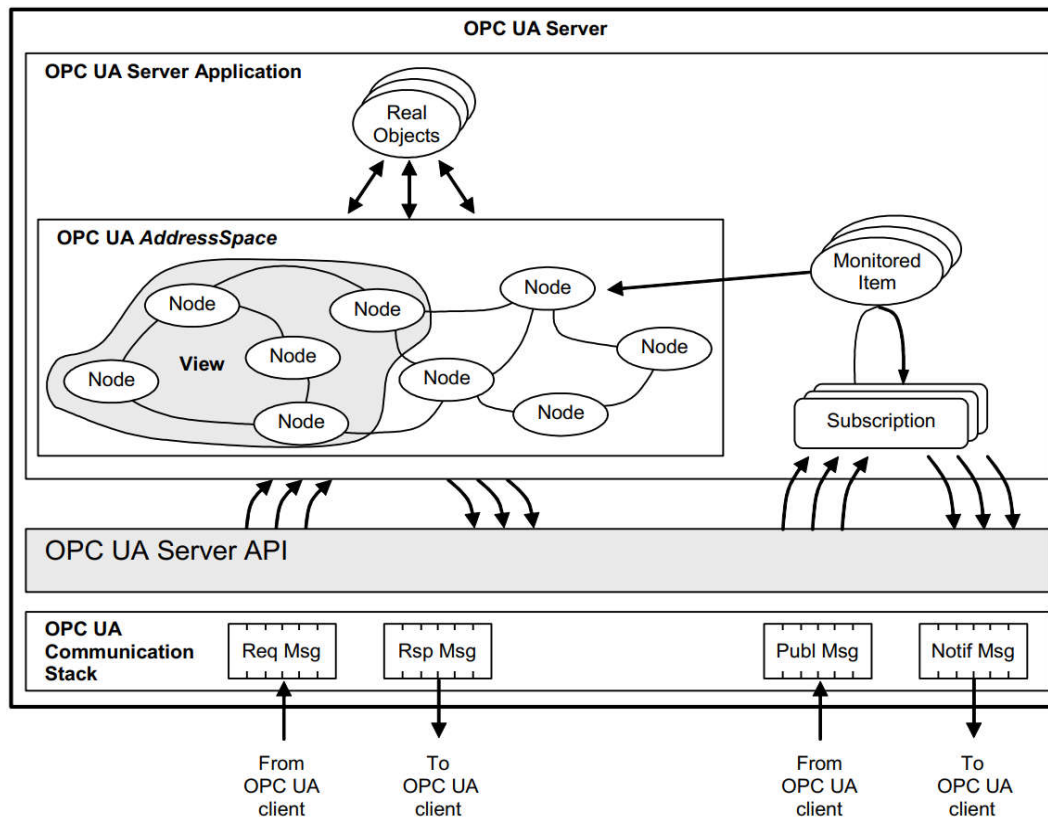
Figure 10. OPC UA Server architecture[2]

The OPC UA Server architecture models the Server endpoint of client/server interactions. Previous Figure 10 illustrates major elements of the OPC UA Server and how they relate to each other:

- **Real objects**: are physical or software objects that are accessible by the OPC UA *Server* application or that it maintains internally. Examples include physical devices and diagnostics counters;

- **OPC UA server application**: is the code that implements the function of the *Server*. It uses the OPC UA *Server* API to send and receive OPC UA *Messages* from OPC UA *Clients*. Note that the "OPC UA *Server* API" is an internal interface that isolates the *Server* application code from an OPC UA Communication Stack;

- **OPC UA address space and nodes**: the *AddressSpace* is modelled as a set of *Node*s accessible by *Clients* using OPC UA *Services* (interfaces and methods). *Node*s in the *AddressSpace* are used to represent real objects, their definitions and their *References* to each other. The use of *References* between *Node*s permits *Servers* to organize the *AddressSpace* into hierarchies, a full mesh network of *Node*s, or any possible mix;

- **Address space views**: are subsets of the *AddressSpace*. *Views* are used to restrict the *Node*s that the *Server* makes visible to the *Client*, thus restricting the size of the *AddressSpace* for the *Service* requests submitted by the *Client*. *Views* may hide some of the *Node*s or *References* in the *AddressSpace*. *Clients* are able to browse *Views* (often organized into hierarchies) to determine their structure;

- **Information models**: The OPC UA *AddressSpace* supports information models. This support is provided through:
  a) *Node References* that allow *Objects* in the *AddressSpace* to be related to each other;
  b) *ObjectType Node*s that provide semantic information for real *Objects* (type definitions);
  c) *ObjectType Node*s to support subclassing of type definitions;
  d) Data type definitions exposed in the *AddressSpace* that allow industry specific data types to be used;

---

[2] OPC UA Part 1 - Overview and Concepts 1.03 Specification

e) OPC UA companion standards that permit industry groups to define how their specific information models are to be represented in OPC UA *Server AddressSpace*;

- **Monitored items**: *MonitoredItems* are entities in the *Server* created by the *Client* that monitor *AddressSpace Node*s and their real-world counterparts. When they detect a data change or an event/alarm occurrence, they generate a *Notification* that is transferred to the *Client* by a *Subscription*;
- **Subscriptions**: are endpoints in the *Server* that publish *Notifications* to *Clients*. *Clients* control the rate at which publishing occurs by sending Publish *Messages*.

The standard specification is organized into several documents related to concepts, security model, address space model, services, information model, mappings, profiles, data access (DA), alarms and conditions (AC), programs (Prog), historical access (HA), discovery and aggregates, respectively. OPC UA defines the sets of services that servers can provide, mapped onto different communication protocols and in which data is encoded in various ways.

Basically, two data encodings are defined, XML/text and UA Binary; transport protocols support OPC UA TCP, SOAP/HTTP and HTTPs. Information models for a specific domain are defined on top of base specifications and organizations can build their own models on top of the UA base or on top of the OPC information model, exposing their specific information via OPC UA. Examples of standards already working on mappings to OPC UA are Field Device Integration (FDI) combining Electronic Device Description Language (EDDL) and Field Device Tool (FDT), both used to describe, configure and monitor devices and PLCopen, a standard for PLC programming languages.

FUTURE OF CLOUD MANUFACTURING IN INDUSTRY 4.0

As cloud architectures become the basis of most innovative manufacturing IT systems, the future role of cloud-technology in Cyber Physical Production Systems (CPPS) has to be properly investigated, as interoperability has vital importance in this field. The term "Industry 4.0" has nowadays totally gained the attention of scientific and economic forums. Conceived as a German national initiative (*Wolf-Dieter*, 2011), it has rapidly evolved to a broader definition used to identify what is commonly considered as the next radical transformation in industry, the fourth industrial revolution, and describes the up-to-date automation and data exchange in manufacturing technologies. It includes CPS, IoTand CC.

The control hierarchy in industry is based on ISA-95 standard (developed on Purdue model) but the modern control and communication systems need a lean structure. Smart products, smart devices and smart-X-objects (including humans), smart connected assets intend to communicate directly with each other, so based on the existing architectures the development of new architectures is needed. The concepts from existing standards can be the base of new standardization developments.

The evolution of IC technologies is continuously going on and not only the separate development of the different technologies but their integrations as well. There are several new concepts, paradigms in the manufacturing industry that are very close to each other, covering nearly the same field. The content of the Industrial Internet (II), for instance, is the internet of things, machines, computers and people, enabling intelligent industrial operations using advanced data analytics for transformational business outcomes.

Cyber Physical Production Systems (CPPS) are based on the newest and foreseeable further developments in computer science, information and communication technologies, and manufacturing science and technology. Based on Industry 4.0 approach, so called "Smart Factories" (SF) are nowadays developed, in which cyber-physical systems monitor physical processes, create a virtual counterpart of the physical world and make decentralized decisions, all while communicating and cooperating with each other (and with humans) in real time using the Internet and IoT components. Besides these comprehensive technologies there are additional new developments like Big Data Analysis, BYOD (Bring Your Own Device), social networks, mobile technology, artificial intelligence, robotics, additive manufacturing and the base of all, the integration background, is the Cloud Computing. The IoT is the novel technology concept which is currently transforming and redefining virtually all markets and industries in fundamental ways. IoT is a new computing, communication concept, according to which any device/object (e.g. sensors, mobile equipment) and

humans can be connected to the Internet (wired, wireless), also to each other, and be able to identify themselves to other devices. The relationships can be built between people, people-things and things-things.

The significance of (I)IoT is that if an object can represented itself digitally and is connected to surrounding objects forming a group of objects and/or (e.g. to a database) it becomes digitally greater than the object by itself as connected to surrounding objects and database data. Objects can be parts of machines (e.g. a touch-sensor of a machine tool, a jet engine of an airplane), wearable devices and anything that can contain a sensor, a chip.

IHS consultation firm forecasts that the IoT market will grow from an installed base of 15.4 billion devices in 2015 to 30.7 billion device in 2020 and 75.4 billion in 2025 (*Lucero*, 2016). IHS also defines an IoT platform as "cloud-based and on premise software packages and related services that enable and support sophisticated IoT services". With the application of IoT platforms enterprises can manage thousands and even millions of devices and connections across multiple technologies and protocols. In the manufacturing sector IoT software enables developers to combine device and connection data with enterprise-specific customer and ERP data as well as data from third-party sources e.g. by direct using marketing data creating more valuable IoT applications. With the integration of Cloud and IoT new systems are developed, which offer effective information sharing and handling. *Botta et al.* (2016) call this approach as "CloudIoT" paradigm, whereas *Zhou et al.* (2013) developed in this sense the "CloudThings" architecture.

CONCLUSION

In industry and in business systems Cloud and Cloud-based manufacturing play nowadays a central role. Direct communication and networked collaboration among objects and human actors in manufacturing architectures are generating crucial challenges for interoperability. Establishing global interoperability in the IIoT is a prerequisite for its proliferation but unfortunately, ensuring interoperability is a complex endeavour. It is not a matter of agreeing on a small set of standards to rule the IIoT world, but about carefully orchestrating complex and partially competing protocols and standards on multiple levels, including device integration, gateway technologies, short-range wireless communication, long-range wireless communication, messaging, event processing, data management, analytics, cloud operations, and so on. No single organization can deliver the one standard which solves all interoperability problems, as there is also the need to involve stakeholders and address domains of interoperability through an integrative, holistic perspective.

Standards usually include both a specification and a set of industry practices that emerge from that specification. RAMI 4.0 and IIRA are two of the most remarkable attempts which convey their efforts into this direction.

The paper focused on the evidence of integration compatibility (at functional level) of their architectural layers over the Cloud and also illustrated how OPC UA technology can play a relevant role (as a horizontal IIoT-enabling solution) to interoperability orchestration and service support, thanks to its modelling semantics affinity to functional elements of the RAMI 4.0 and IIRA architectures. To enable bidirectional data and information flows between IIoT systems heterogeneously built on IIRA and RAMI 4.0, functional requirements driving the interoperability have to be defined and from there functions and semantics that can be understood by these systems have to be specified, with unique "understanding" for properties and the assets (parts, components and machines) and means of communications between these systems. Therefore, identification, communication, semantic and functional mapping between IIRA and RAMI 4.0 is a foundational step toward enabling interoperability between systems build upon the two architectures.

Interoperability is still an open issue in the current panorama of cloud-based Industry 4.0 implementations: if from one hand the removal of methodological incompatibilities in designing new manufacturing scenarios will surely be accelerated by the adoption of standardized solutions, from the other, adequate attention needs to be dedicated also to governance processes and partnerships, nevertheless required to establish an open and interoperable IIoT. Achieving interoperability may require not only sound architectural principles but also dynamic and attractive negotiation between cloud providers and stakeholders of the industrial domain.

ACKNOWLEDGEMENT

REFERENCES

Andrew, H. (2015). *ISA-95 Unplugged: Putting MOM Things Together for IoT*. Retrieved January 06, 2018, from http://blog.lnsresearch.com/isa-95-unplugged-putting-mom-things-together-for-iot.

ATHENA European project (2007). *Specification of Interoperability Framework and Profiles, Guidelines and Best Practices*. ATHENA Integrated Project (Deliverable D.A4.2, p. 215).

Bommadevara, N., Kaplan, J., & Starikova, I. (2016). *Leaders and laggards in enterprise cloud infrastructure adoption*. McKinsey & Company. Retrieved January 06, 2018, from http://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/leaders-and-laggards-in-enterprise-cloud-infrastructure-adoption

Botta, A., Donato, W. D., Persico, V., & Pescape, A. (2016). Integration of Cloud computing and Internet of Things: A survey. *Future Generation Computer Systems*, *56*, 684–700.

Camarinha-Matos, L. M., & Afsarmanesh, H. (2005). Collaborative networks: a new scientific discipline. *Journal of Intelligent Manufacturing, 16*, 439–452.

Camarinha-Matos, L. M., Afsarmanesh, H., Galeano, N., & Molina, A. (2009). Collaborative networked organizations – Concepts and practice in manufacturing enterprises. *Computers & Industrial Engineering, 57*, 46–60.

Chen, D., Doumeingts, G., & Vernadat, F. (2008). Architectures for enterprise integration and interoperability: Past, present and future. *Computers in Industry, 59*, 647–659.

Cheng, Z., Guo, X., Ou, W., & Han, D. (2009). Cloud Computing Service Composition and Search Based on Semantic. In *Proceedings of the 1st International Conference on Cloud Computing* (vol. 1, pp. 290-300). Martin Gilje Jaatun, Gansen Zhao, and Chunming Rong (Ed.) Springer-Verlag, Berlin, Heidelberg.

Cloud-standards (2017). *The Cloud Standards Wiki*. Cloud-standards.org. Retrieved January 06, 2018, from http://cloud-standards.org

European Commission (2016). *European Commission sets out path to digitise European industry*. Retrieved January 06, 2018, from https://ec.europa.eu/digital-single-market/en/news/commission-sets-out-path-digitise-european-industry

Fraunhofer IPA (2016a). *Manufacturing Service Bus. The Flexible Solution for Digitizing Production Processes*. Fraunhofer IPA. Retrieved January 06, 2018, from http://www.ipa.fraunhofer.de/fileadmin/user_upload/Kompetenzen/Digitale_Werkzeuge_in_der_Produktion/Cloud-Plattformen/English_Documents/Productsheet_Manufactoring_Service_Bus.pdf

Fraunhofer IPA (2016b). *Virtual Fort Knox. The Only Open It Cloud Platform For Manufacturing Companies*. Fraunhofer IPA. Retrieved January 06, 2018, from http://www.ipa.fraunhofer.de/fileadmin/user_upload/Kompetenzen/Digitale_Werkzeuge_in_der_Produktion/Cloud-Plattformen/English_Documents/Productsheet_Virtual_Fort_Knox.pdf

Golightly, D., Sharples, S., Harshada Patel, H., Ratchev, S. (2016). Manufacturing in the cloud: A human factors perspective. *International Journal of Industrial Ergonomics*. Volume 55, 12-21.

IIC & Plattform Industrie 4.0 (2017). *Industrial Internet Consortium & Platform Industrie 4.0*. Accessed January 06, 2018, from https://www.infosys.com/engineering-services/white-papers/Documents/industrial-internet-consortium-architecture.pdf

IIC Working TWG and SWG (2015). *Industrial Internet Reference Architecture*. Industrial Internet Consortium. Retrieved January 06, 2018, from https://www.iiconsortium.org/IIRA.htm

Infosys & IIC (2015a). *IIC - Asset Efficiency Testbed*. IIC. Retrieved January 06, 2018, from http://www.iiconsortium.org/asset-efficiency.htm

Infosys & IIC (2015b). *IIC - Industrial Digital Thread Testbed*. IIC. January 06, 2018, from http://www.iiconsortium.org/industrial-digital-thread.htm

ISO/IEC/IEEE (2011). *ISO/IEC/IEEE 42010:2011 Architecture description*. Retrieved January 06, 2018, from http://www.iso-architecture.org/ieee-1471/

Jula, A., Sundararajan, E., & Othman, Z. (2014). Review: Cloud computing service composition: A systematic literature review. *Expert Systems with Applications, 41(8)*, 3809-3824,

Lee, E.A. (2008). *Cyber Physical Systems: Design Challenges*. University of California. Retrieved June 30, 2017, from https://www2.eecs.berkeley.edu/Pubs/TechRpts/2008/EECS-2008-8.pdf

Li, B.H., Zhang, L., Wang, S.L., Tao, F., Cao, J.W., Jiang, X.D., Song, X., & Chai, X.D. (2010). Cloud manufacturing: a new service-oriented networked manufacturing model. *Computer Integrated Manufacturing Systems, 16(1)*, 1–7.

Liu, Y., & Xu, X. (2016). Industry 4.0 and Cloud Manufacturing: A Comparative Analysis. *Journal of Manufacturing Science and Engineering, 139(3).*

Lucero, S. (2016). *IoT platforms: enabling the Internet of Things*. IHS. Retrieved January 06, 2018, from https://cdn.ihs.com/www/pdf/enabling-IOT.pdf

Marston, S., Zhi, L., Bandyopadhyay, S., Zhang, J., & Ghalsasi, A. (2011). Cloud computing - The business perspective. *Decision Support Systems, 51(1)*, 176-189.

MESA International, & IBM (2010). SOA Manufacturing Guidebook. *MESA International*. Retrieved January 06, 2018, from ftp://public.dhe.ibm.com/software/plm/pdif/MESA_SOAinManufacturingGuidebook.pdf

Mezgar, I., & Rauschecker, U. (2014). The Challenge of Networked Enterprises for Cloud Computing Interoperability. *Computers in Industry, 65(4)*, 657-674.

Mourad, M., Nassehi, A., & Schaefer, D. (2016). Interoperability as a Key Enabler for Manufacturing in the Cloud. *The Sixth International Conference on Changeable, Agile, Reconfigurable and Virtual Production (CARV2016)*. Aydin Nassehi and Stephen Newman Edition (vol. 52, pp. 30–34). Procedia CIRP.

Ning, L., & Xiaoping L. (2012). A Resource Virtualization Mechanism for Cloud Manufacturing Systems. Marten Sinderen et al.. eds. (IWEI), Sep 2012, Harbin, China. Springer, *Lecture Notes in Business Information Processing*, LNBIP-122, pp.46-59.

NIST (2011). *The NIST Definition of Cloud Computing,* NIST Special Publication 800-145. October 25, 2011, Retrived January 8, 2018 from https://www.nist.gov/news-events/news/2011/10/final-version-nist-cloud-computing-definition-published

NSF Program Announcements (2017). *Cyber-Physical Systems (CPS).* Document Number: nsf17529, December 8, 2016. Retrieved January 06, 2018, from https://www.nsf.gov/pubs/2017/nsf17529/nsf17529.pdf

OPC UA (2017). *OPC Unified Architecture.* OPC Foundation. Retrieved January 06, 2018, from https://opcfoundation.org/developer-tools/specifications-unified-architecture

Panetto, H., & Molina, A. (2008). Enterprise integration and interoperability in manufacturing systems: Trends and issues. *Computers in Industry, 59,* 641–646.

Qanbari, S., Zadeh, S.M., Vedaei, S. & Dustdar, S. (2014). CloudMan: A platform for portable cloud manufacturing services. *IEEE International Conference on Big Data,* 1006-1014. IEEE.

Ren, L., Zhang, L., Wang, L., Tao, F. & Chai, X. (2014). Cloud manufacturing: Key characteristics and applications. *International Journal of Computer Integrated Manufacturing,* 37-41.

RightScale (2017). *State of the Cloud Report.* Retrieved July 04, 2017, from http://assets.rightscale.com/uploads/pdfs/RightScale-2017-State-of-the-Cloud-Report.pdf

Schwab, K. (2015). *The Fourth Industrial Revolution: What It Means and How to Respond.* Foreign Affairs, Science & Technology. Retrieved January 06, 2018, from https://www.foreignaffairs.com/articles/2015-12-12/fourth-industrial-revolution

VDI, VDE, & ZVEI (2015). *GMA Status Report: Reference Architecture Model Industrie 4.0 (RAMI 4.0).* Retrieved July 04, 2017, from https://www.zvei.org/fileadmin/user_upload/Presse_und_Medien/Publikationen/2016/januar/GMA_Status_Report__Reference_Archtitecture_Model_Industrie_4.0__RAMI_4.0_/GMA-Status-Report-RAMI-40-July-2015.pdf

Wang, P., Gao, R.X., Fan, Z. (2015). Cloud Computing for Cloud Manufacturing: Benefits and Limitations. ASME. *Journal Manufacturing Scientific Engineering.* 137(4), 040901-040901-9.

Wang, X.V., & Xu, X.W. (2013). An interoperable solution for Cloud manufacturing. *Robotics and Computer-Integrated Manufacturing, 29 (4),* 232–247.

Wei, L., Zhu, H., Cao, Z., Dong, X., Jia, W., Chen, Y., & Vasilakos, A.V. (2014). Security and privacy for storage and computation in cloud computing. *Information Sciences, 258(10),* 371–386. DOI: http://dx.doi.org/10.1016/j.ins.2013.04.028

Weissberger, A. (2011). *IEEE CIO Says Cloud Interoperability a Bigger Problem than Security!* Retrieved February 2, 2017, from http://community.comsoc.org/blogs/alanweissberger/ieee-cio-says-cloud-interoperability-bigger-problem-security

Wolf-Dieter, L. (2011). Industrie 4.0: Mit dem Internet der Dinge auf dem Weg zur 4. industriellen Revolution. *VDI nachrichten, 13.* Retrieved July 7, 2017, from http://www.wolfgang-wahlster.de/wordpress/wp-content/uploads/Industrie_4_0_Mit_dem_Internet_der_Dinge_auf_dem_Weg_zur_vierten_industriellen_Revolution_2.pdf

Wu, D., Greer, M., Rosen, D., Schaefer, D. (2013). Cloud manufacturing: Strategic vision and state-of-the-art. *Journal of Manufacturing Systems.* Volume 32, Issue 4, 564-579.

Wu, D., David, W.R., Wang, L., & Schaefer, D. (2014). Cloud-Based Manufacturing: Old Wine in New Bottles? In *Proceedings of the 47th CIRP Conference on Manufacturing Systems* (vol. 17, pp. 94-99). Variety Management in Manufacturing.

Xu, X. (2012). From cloud computing to cloud manufacturing. *Robotics and Computer-Integrated Manufacturing, 28(1)*, 75–86.

Yadgarova Y., & Taratukhin, V. (2016). An Interoperable Cloud Environment of Manufacturing Control System. In Mertins K., Jardim-Gonçalves R., Popplewell K., Mendonça J. (Ed.) *Enterprise Interoperability VII. Proceedings of the I-ESA Conferences, vol 8*. Springer, Cham

Zhang, L., Luo, Y.-L., Tao, F., Ren, L., & Guo, H. (2010). Key technologies for the construction of manufacturing cloud. *Jisuanji Jicheng Zhizao Xitong / Computer Integrated Manufacturing Systems, 16(11)*, 2510-2520.

Zhang, L., Luo, Y., Tao, F., Li, B., Ren, L., Zhang, X., Guo, H., Cheng, Y., Hu, A., & Liu, Y. (2011). Cloud manufacturing: a new manufacturing paradigm. *Enterprise Information Systems, 8(2)*, 167-187.

Zhang, Y., Zhang, G., Liu, Y., & Hu, D. (2017). Research on services encapsulation and virtualization access model of machine for cloud manufacturing. *Journal of Intelligent Manufacturing*, 28(5), 1109-1123.

Zhou, J., Leppanen, T., Harjula, E., Ylianttila, M., Ojala, T., Yu, C., Jin, H., & Yang, L.-T. (2013). CloudThings: A common architecture for integrating the Internet of Things with Cloud Computing. In *Proceedings of the 2013 IEEE 17th International Conference on Computer Supported Cooperative Work in Design* (pp. 651-657).

Zissis, D., & Lekkas, D. (2012). Addressing cloud computing security issues. *Future Generation Computer Systems, 28(3)*, 583-592.

KEY TERMS AND DEFINITIONS

Cloud Computing: in general, an information technology architectural model where computing services (both hardware and software) are delivered to customers over the Internet, on-demand, in a self-service fashion, independent of device and location.

Cloud Interoperability: enabling the Cloud Computing ecosystem whereby individuals and organizations are able to widely adopt Cloud Computing technology and related services in such a fashion that multiple cloud platforms can exchange information in a unified manor and ultimately work together seamlessly.

Cloud Manufacturing: the basic idea was to apply cloud computing architecture, tasks and service structure to manufacturing systems. The result of this mirroring is a new service-oriented networked manufacturing.

Cloud Models: this is typically an end-user's perspective in Cloud Computing industry, where different delivery models refer to different layers of the cloud architecture.

Cyber-physical Production System: a dual-systems in which they monitor physical processes, create a virtual counterpart of the physical world and make decentralized decisions, all while communicating and cooperating with each other (and with humans) in real time using the Internet and IoT components.

IIRA: IIRA is a standard-oriented open architecture for Internet Industrial Systems by IIC Working TWG & SWG which aims at extending industry applicability and interoperability, and guiding technology standards development.

Industrial Internet of Things: the IoT is a network of intelligent computers, devices, and objects that collect and share huge amounts of data. The collected data is sent to a central Cloud-based service where it is aggregated with other data and then shared with end users in a helpful way. The application of the IoT to the manufacturing industry is called the IIoT.

Industry 4.0: high-tech ICT-based strategic industrial program which focuses on manufacturing and describes the up-to-date automation and data exchange in manufacturing technologies. Industry 4.0 includes Cyber- Physical Systems, the Internet of Things and cloud computing.

Interoperability: can be defined as the ability of two or more systems or applications to exchange information and to mutually use the information that has been exchanged without special effort on the part of the customer.

RAMI 4.0: the reference architecture model of Industrie 4.0, conceived around the Cyber-physical Production Systems as in an interconnected factory capable of producing highly customizable products, realized through flexible mass production.

Smart Factory: a flexible system that can self-optimize performance across a broader network, self-adapt to and learn from new conditions in real or near-real time, and autonomously run entire production processes.