Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 55-10 (2022) 2950–2955

Three-Faceted Manufacturing Knowledge Representation in Cloud Environments ⋆

Damian Drexel * **Ralph Hoch** *

* *Research Center Digital Factory, Vorarlberg University of Applied Sciences (e-mail: {damian.drexel,ralph.hoch}@fhv.at)*

Abstract: A trend from centralized to decentralized production is emerging in the manufacturing domain leading to new and innovative approaches for long-established production methods. A technology supporting this trend is Cloud Manufacturing, which adapts technologies and concepts known from cloud computing to the manufacturing domain. A core aspect of Cloud Manufacturing is representing knowledge about manufacturing, e.g., machine capabilities, in a suitable form. This knowledge representation should be flexible and adaptable so that it fits across various manufacturing domains, but, at the same time, should also be specific and exhaustive. We identify three core capabilities that such a platform has to support, i.e., the product, the process and the production. We propose representing this knowledge in semantically specified knowledge graphs, essentially creating three through features interconnected ontologies each representing a facet of manufacturing. Finally, we present an exemplary implementation of a Cloud Manufacturing platform using this representation and its advantages.

Keywords: cloud manufacturing, knowledge representation, ontologies, abstraction, distributed manufacturing

1. INTRODUCTION

Decentralized production will be of great interest for manufacturers in the future, with increasing costs of logistics, mass customization and other aspects being major factors for this trend (Matt et al., 2015). This will not just affect physically distributing manufacturing efforts, but will also require new coordination efforts between manufacturing locations. To achieve the shift from centralized production to a distributed and possibly flexible manufacturing network, technologies and concepts supporting such a paradigm have to be explored. In recent years research efforts for distributed manufacturing drew inspiration from the emergence of cloud technologies (Xu, 2012) and introduced Cloud Manufacturing (CMfg) as a possible solution for the challenges of distributed manufacturing.

The main objective of a platform realizing CMfg is to connect parties wanting to manufacture a given product, with parties capable of doing so. The parties involved in this process can be categorized into three groups: *Customers*, *Manufacturers* and *Cloud Providers*. Customers are interested in finding a way to *manufacture* their *product* and manufacturers provide *manufacturing means*, while cloud providers contribute the environment in which customers and manufacturers are brought together. Each of the involved parties, or stakeholders, is interested in different aspects of the CMfg platform.

Such a CMfg platform requires an appropriate representation of knowledge and information regarding manufacturing in order to facilitate connecting the stakeholders. We argue that this representation has to follow the three facets (1) *Product*, (2) *Process* and (3) *Production*, emerging from the underlying manufacturing nature. Hence, we propose using complementary ontologies for knowledge representation in a CMfg platform, with each ontology representing a specific facet of the overall manufacturing procedure. To bring the involved stakeholders, e.g., customers and manufacturers, together, appropriate interfaces are required for information exchange. These interfaces are built upon digital representations of physical manufacturing resources and products represented by the two facets *Production* and *Product*. As with representations in the classical manufacturing domain, these two fields can come together in the description of the production process – the third facet *Process*. Using these facets enables decoupling knowledge representations for each stakeholder while providing the means for connecting them through defined concepts. Through these common concepts abstraction of complexity, e.g., customized specifications for domains, becomes possible. Furthermore, it facilitates ontological reasoning for deriving knowledge, e.g., automatically linking products and fitting processes. We present an implementation of a CMfg platform that builds upon ontological knowledge representation using these three facets.

The paper is organized in the following manner. First, we discuss related literature and give some background information for making the paper self-contained. Section 3 develops the core capabilities expected by a CMfg system by exploring it's parts and stakeholders. Section 4 describes an ontological approach of modelling such a knowledge representation. Section 5 demonstrates an exemplary implementation of a CMfg system based on the knowledge representation described in Section 4. Finally,

⋆ The Cloud-Manufacturing platform, data analytics, and IT- and OT-Security efforts are financed by FFG-Project No. 866833 "CIDOP".

10.1016/j.ifacol.2022.10.180

we reflect on our work, discuss open research directions and conclude.

2. BACKGROUND AND RELATED WORK

The paradigm of CMfg with it's various research directions received significant interest during the last decade (Liu et al., 2018). CMfg draws inspiration from cloud computing as it aims at providing manufacturing means as services, in a highly dynamic and demand oriented way (Xu, 2012).

An aspect often discussed is the architecture of such a cloud based manufacturing system with various research suggesting a layer based architecture. Xu (2012) separates the system into application, global service, virtual service, and manufacturing resource layer. Tao et al. (2011) propose a 10 layer model and integrate security as well as knowledge into their view on the system. Liu et al. (2017) propose a four layer architecture similar to Xu (2012).Ding et al. (2016) also presents a four layered manufacturing scenario. Fei Tao et al. (2014) three layers. We also use a layered architecture for the implementation of the CMfg platform.

To virtualise manufacturing resources for using them as services in CMfg, Xu (2012) proposes description languages, i.e., ontologies. Ontologies enable a machine interpretable representation of not only manufacturing resources, but also other manufacturing related knowledge required by CMfg. For these languages Web Ontology Language (OWL) is a possible candidate (Tao et al., 2015). Similarly, Lu et al. (2014) states that an ontology is required not only for the virtualisation of manufacturing resources, but also to enable interoperability. Wu et al. (2013) mentions OWL as well as other technologies being suitable for composing services in CMfg.

Järvenpää et al. (2018) propose to use an ontology for representing capabilities of manufacturing resources facilitating their reuse. Similar to the approach presented in our work, they identify different views within their ontology. They introduce an ontology for products as well as an ontology for manufacturing resources. To connect these two ontologies a capability ontology, describing what manufacturing resources can do, is introduced and used with the process taxonomy.

We propose using *Description Logics (DL)* for knowledge representation. DL describe a family of logics for knowledge modeling that are essentially fragments of first order predicate logic (Hitzler et al., 2008). In ontology languages based on DL, semantics are expressed with rules, which relate to a semantically predefined vocabulary. For building an ontology based on description logics, concepts, sets of objects, and roles, denoting binary relations between instances of those concepts, are used as semantic entities, which can be atomic or complex. Complex concepts and roles are created using constructors.

OWL, a standard from the World Wide Web Consortium (W3C), is a language for representing an ontology. Currently there are OWL 1, where OWL 1 DL is the language based on Description Logics, and OWL 2, which is an extension of OWL 1 DL. For a detailed description of the extended features of OWL 2 as compared to OWL 1 see OWL2Features [1] . The language builds upon Extensible Markup Language (XML) and Resource Description Framework (RDF) and uses a triple structure where a triple consists of subject, predicate and object. Such a triple can be seen as a Node-Edge-Node construct, and a set of triples builds an RDF graph (Klyne and Carroll, 2004).

3. KNOWLEDGE REPRESENTATION FOR CLOUD MANUFACTURING

CMfg adapts concepts from cloud computing and tries to offer manufacturing capabilities as services. Furthermore, CMfg employs architectures similar to the ones seen in cloud computing. Commonly used architectures divide the cloud system into layers. Each of these layers has it's tasks and responsibilities, like providing a way to connect physical manufacturing resources to the digital cloud. In this section we identify stake holders of a CMfg platform and their requirements. Furthermore, we investigate the layers of CMfg platforms and their tasks, required to fulfill the stake holder's requirements. This analysis supports the definition of the three main facets of CMfg, inherited from it's manufacturing nature, which are the foundation of the knowledge representation, presented in this paper.

Since knowledge representation is only feasible for defined concepts, we first have to identify the requirements a CMfg has to fulfill. For identifying these requirements we analyze the interests of each stakeholder and derive requirements and successively knowledge concepts.

A customer requires a way to state their wish for a product to be manufactured. This can be achieved via various interaction possibilities for humans, like web or mobile frontends, as well as interfaces for programmatic interaction with the cloud. Besides the product itself, a customer may have additional constraints on the manufacturing process, like cost, quality, or time constraints.

Manufacturers are providing their manufacturing resources to the platform. They can be added and removed depending on factors like utilization and availability. This leads to a highly dynamic and constantly changing manufacturing environment. Offering manufacturing resources in such a way is only viable if the process is highly automated. Furthermore, additional interests of manufacturers have to be considered when digitising manufacturing resources and connecting them to the cloud. Exact machine configurations may be considered as highly sensitive data and, therefore, may not be published to the cloud. Furthermore, the manufacturers have to decide about the degree of control over the published manufacturing resources they want to share with the cloud. This can range from only receiving orders which have to be manually checked by humans to automatically producing an order received from the cloud. A further complication is that various domains are served by different manufactures, each with their special needs and specifications.

Bringing together customers and manufacturers, the cloud provider contributes the actual platform. This comprises services for the various tasks to be performed, the knowledge required by these tasks, as well as interfaces to in-

[1] `https://www.w3.org/TR/owl2-new-features/`

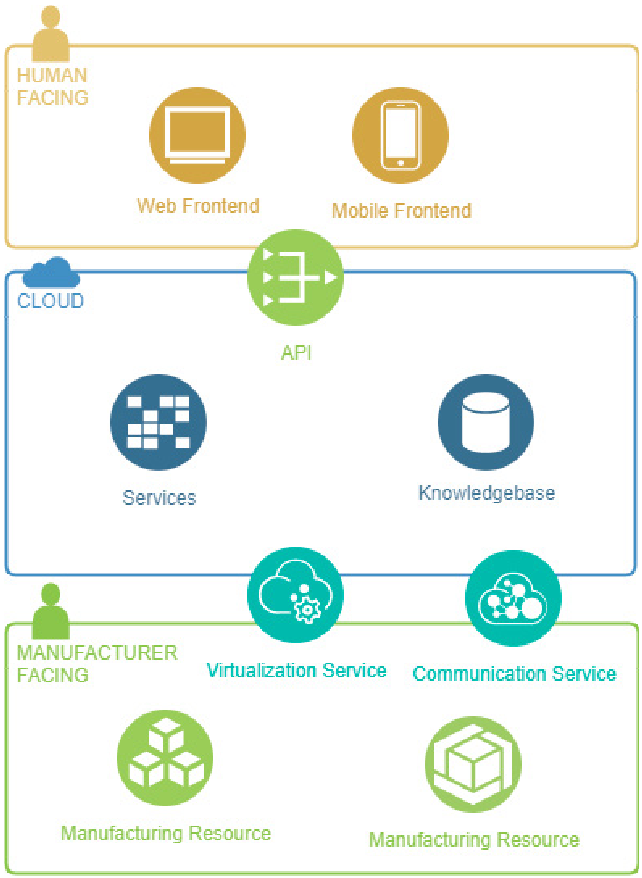

Fig. 1. The layers of a CMfg platform

teract with the CMfg platform. These three aspects are the core of the system. Some of these interfaces are to be used to enable the interaction of humans, especially customers, with the platform. Others have to enable the virtualisation of manufacturing resources and the communication between them and the CMfg system. According to the previous observations, a CMfg system can be divided into layers, with each layer fulfilling tasks to meet the requirements identified before.

The three layers of a CMfg platform, depicted in figure 1, are the *Cloud Core Layer*, the *Human Facing Layer* and the *Cloud Core Layer*. The services of the *Cloud Core Layer*, for tasks like scheduling, process coordination, or resource management, provide the functionalities of the CMfg system. This layer and especially it's services builds on a well defined *knowledge representation*. The *Human Facing Layer* allows interactions with the system to create orders and interact with the cloud. It contains human oriented interfaces to the CMfg platform, like web shops. While customers are the main interested parties, these interfaces can be of use for manufacturers, too. The third layer, the *Manufacturer Facing Layer*, consists of digital as well as physical parts. The virtualisation of manufacturing capabilities, for example machines on the shopfloor, comprises a major part of the knowledge in the *Cloud Core Layer*. It describes the abilities of manufacturing resources as well as interfaces to them. Furthermore, ways to enable communication between the cloud and the shopfloor have to be established at this layer.

The tasks linked to these layers serve towards three core capabilities a CMfg platform has to provide in order to fulfill the interests of the three main stakeholders: The (1) creation of a request for manufacturing a product, the (2) creation of a manufacturing process to fulfill this request, and the (3) coordinated execution of this manufacturing process. These core capabilities require knowledge about the product, the process, as well as the manufacturing means. Knowledge about the product includes the description of it's parts and their variations (product configuration), as well as the information for the manufacturing process to be defined by the customer (manufacturing step configuration). In order to create the manufacturing process, knowledge about the manufacturing steps capable of producing a (part of a) product is needed. Furthermore, information about the availability of manufacturing resources has to be made available to the CMfg platform. For the system to be able to execute the manufacturing process and to coordinate between the involved manufacturing resources, interfaces for the communication have to be known. The status of a single step in the execution of the manufacturing process, as well as an overall status are of importance to the system, as well.

While the knowledge required on the shopfloor has to be of great detail, as seen in Lemaignan et al. (2006), the CMfg platform's requirements for the knowledge provided allow a more abstract view. There is no need to know the exact configuration of a manufacturing resource, as long as the platform knows whether or not the resource can be utilized for a specific manufacturing step.

These considerations and the identification of the three core capabilities of a CMfg platform allow us to specify the knowledge needed in such a system.

Building upon the three core capabilities of a CMfg platform we can define the knowledge required by it. Furthermore, we can categorize this knowledge into the three facets (1) product, (2) process, and (3) production knowledge. As with the layers of a CMfg system, these categories of knowledge serve specific aspects of the overall system, but they cannot be seen independently. They overlap, share information required by their associated tasks, and have to be seen as linked to each other to form a knowledge base suitable for a CMfg environment.

To represent this knowledge we propose the utilization of ontologies and associated technologies to build the CMfg knowledge base. A common terminology builds the basis for such an ontology. For each of the identified categories of knowledge, namely product, process and production, specialized ontologies refine the concepts from this upper ontology where required and can still operate on more abstract terms where possible. This decoupling of each field of knowledge not only conforms with the findings from the previous section, but also supports the modular and service oriented architecture inherent to cloud systems in general and CMfg systems in particular.

4. MANAGING COMPLEXITY THROUGH ABSTRACTIONS IN KNOWLEDGE-BASES

Based on the findings above we developed ontologies for capturing knowledge required by a CMfg platform. We propose using separate ontologies instead of a holistic one, with each part representing a facet of the overall CMfg platform. The ontologies are an *Upper Ontology*, a *Product Ontology*, a *Process Ontology*, and a *Production Ontology*.

The upper ontology defines concepts and relations between them. These concepts are of interest for one or multiple of the specialized ontologies. It, therefore, connects the three areas of a CMfg platform and allows interactions between them. For example, a *Product* is used by both, the product and the process ontology. Figure 2 shows an excerpt of these common concepts.

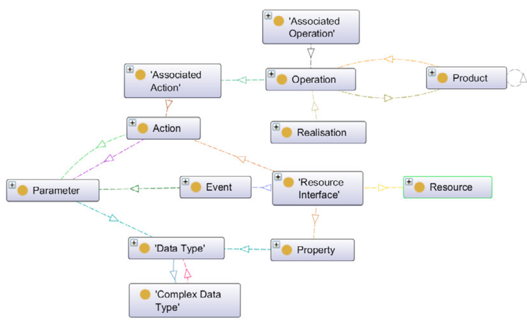

Fig. 2. Upper ontology

While the process ontology uses the *Product* defined in the upper ontology, the product ontology, shown in figure 3, refines this concept. The ontology differentiates between a *Configurable Product* and a *Configured Product*. A *Configurable Product* describes possible variations of a product, as well as other aspects that have to be configured by the customer. The result of this configuration process is the *Configured Product*. While this distinction and additional details about the *Configured Product* are of importance for the *Human Facing Layer*, for the creation and execution of the manufacturing process solely the *Product* to be manufactured, with a fixed subset of all options, is relevant.

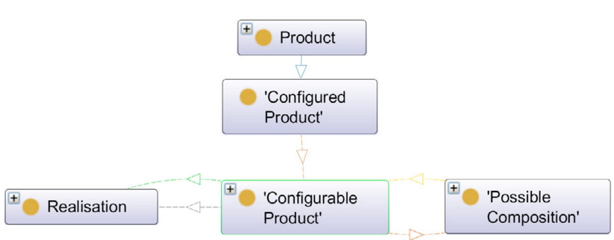

Fig. 3. Product ontology

Therefore, an order by the customer, as defined by the process ontology, shown in Figure 4, only contains *Products*. Furthermore, it requires knowledge about the manufacturing steps to be executed, which combined with the order, are the basis for the manufacturing process.

The execution of the manufacturing process itself requires communication and interaction with the involved manufacturing resources. The ontology defining the types of these resources, like machines, humans, and others, is shown in figure 5. Human resources, for example are required for manual tasks. Additionally, this ontology describes the interface available to interact with the resources. As a result, this ontology is an integral part of the virtualisation of the available resources.

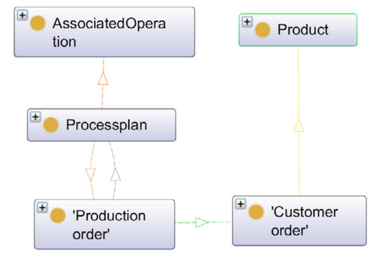

Fig. 4. Process ontology

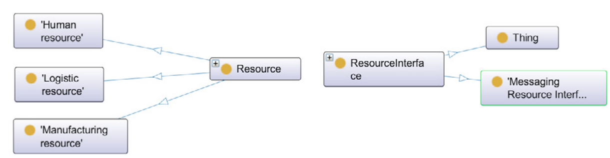

Fig. 5. Production ontology

Compared to ontologies from the 'classical' manufacturing domain, like MASON (Lemaignan et al., 2006), the ontologies described here are on a higher level of abstraction. The CMfg platform cannot replace manufacturing control on site. Rather, the platform acts as an intermediary, only having an overview of possible ways to manufacture a specific part. It does not distinguish between a single machine or a combination of machines. To fulfill it's core capabilities a much higher level of abstraction suffices.

Furthermore, having three distinct ontologies representing facets of the overall manufacturing, allows customization through specialisation. For example, if a product has specific constraints or requires specific specifications, the existing ontology can be extended without breaking the overall system or making rebuilding the knowledge base necessary. This is due to the common concepts of the upper ontology that facilitate the interaction decoupled of specialized concepts. If specialized concepts are necessary for performing a more efficient linking then all involved ontologies can be specialized without the need for changing the existing ontologies and, hence, breaking the current model building. Similarly, the production ontology can be extended without interfering with the other ontologies. New kinds of resources can be added and interfaces to them can be defined, without having to change how process or product related parts of the system use this knowledge.

Another advantage of using description logic for knowledge representation is the possibility for automatic reasoning. Concepts can be linked by means of inference and, e.g., subsumption reasoning.

5. REFERENCE IMPLEMENTATION

According to the concepts discussed before and based on the knowledge representation described in this work, we developed an implementation of a CMfg platform. Figure 6 shows an overview of the parts of this platform. At the core lies a knowledge base, mainly comprised by ontologies with additional databases enhancing them.

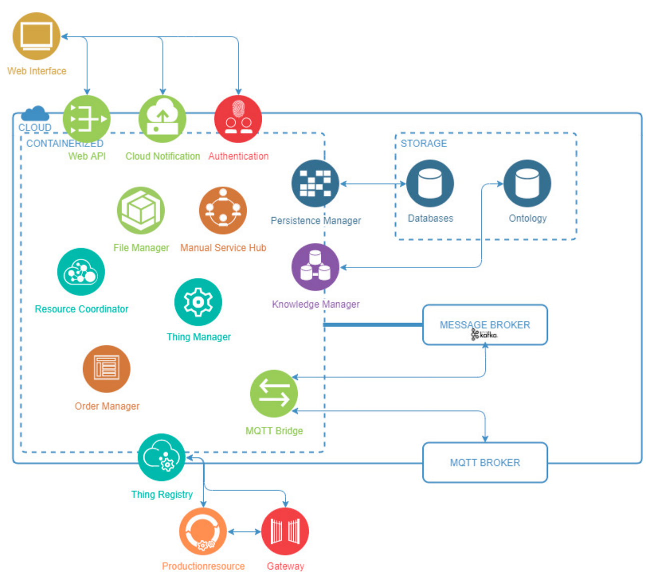

Fig. 6. CIDOP CMfg platform

Together with various services this knowledge base builds the core of the cloud layer of the CMfg platform. APIs, message brokers, and authentication services offer interfaces for the other two layers to interact with it. A web based interface allows customers to configure products and create orders. Additionally, manufacturers can query information about their provided manufacturing resources. Manual tasks, like design tasks, can also be offered via a web based application.

To automate the virtualisation process as much as possible and simplify the connection process of shopfloor resources to the cloud a gateway can be setup on-site. It defines the interface to the manufacturing resource for the CMfg platform and acts as intermediary between the shopfloor and the cloud.

In figure 6 the CMfg platform is visualised. The area labeled *Cloud* corresponds to the cloud core layer. Various services provide the functionalities to fulfill the requirements of this layer. For example, the order manager is essential for defining the manufacturing process. Nevertheless, the depicted services are only part of the required functionality and additional services, not listed here, are part of the implementation. Communication between the services uses different concepts, like the publish-subscribe pattern and Representational State Transfer (REST) paradigm, found in cloud computing systems, too. The knowledge required by the services is represented using the ontologies described previously in this paper.

The knowledge manager serves as interface to this knowledge, which is stored in a GraphDB [2] instance. The knowledge base is enhanced by additional database systems, like relational databases and object stores. The data stored in these databases is used to provide additional features to the users of the cloud platform, such as push notifications.

The cloud core layer defines interfaces, i.e., the *Web API*. These interfaces are used to develop web applications, like web shops and product configurators. These applications form the human facing layer of the CMfg platform.

Other interfaces, such as the *Thing Registry*, are used for connecting the cloud core layer to the manufacturer facing layer. On this layer a *Gateway* is located. It's initial task is the virtualisation, especially the semantic description, of the manufacturing resources connected to it. Furthermore, it enables the communication between these manufacturing resources and the cloud system.

6. CONCLUSION

In this paper, we identified requirements that a CMfg platform has to fulfill by analyzing the involved stakeholders and their needs. Successively, we used these requirements for formalizing fitting knowledge representations so that manufacturing can be supported in a cloud environment. We organized this knowledge in complementary ontologies with each representing a specific facet of the overall manufacturing. In contrast to previous works, we argue that knowledge representations in CMfg should abstract away details, e.g., the exact configuration of a manufacturing resource, and that it is of higher importance to determine whether a product can be produced by a production step. In an implementation of a CMfg platform we demonstrate how such knowledge representations can be included for providing services to connected stakeholders. Technically speaking, we use a combination of OWL 2 and GraphDB for knowledge representation.

Since CMfg has a highly dynamic nature and resources, e.g., products or machines, are constantly added and removed, it is necessary to constantly adapt the knowledge base. To minimize manual effort future work will have to investigate the possibilities of an automatic linking of concepts across the different facets by means of, e.g., graph learning through embeddings.

REFERENCES

Ding, T., Yan, G., Lei, Y., and Xu, X. (2016). Research on Critical Technologies of Manufacturing Execution Based on Cloud-Service. In *2016 3rd International Conference on Information Science and Control Engineering (ICISCE)*, 538–542. IEEE, Beijing, China. doi:10.1109/ICISCE.2016.121.

Fei Tao, Ying Cheng, Li Da Xu, Lin Zhang, and Bo Hu Li (2014). CCIoT-CMfg: Cloud Computing and Internet of Things-Based Cloud Manufacturing Service System. *IEEE Transactions on Industrial Informatics*, 10(2), 1435–1442. doi:10.1109/TII.2014.2306383.

[2] `https://graphdb.ontotext.com/`

Hitzler, P., Krötzsch, M., Rudolph, S., and Sure, Y. (2008). *Semantic Web: Grundlagen*. Springer, Berlin. doi:10.1007/978-3-540-33994-6.

Järvenpää, E., Lanz, M., and Siltala, N. (2018). Formal resource and capability models supporting re-use of manufacturing resources. *Procedia Manufacturing*, 19, 87–94. doi:https://doi.org/10.1016/j.promfg.2018.01.013. URL `https://www.sciencedirect.com/science/article/pii/S2351978918300131`. Proceedings of the 6th International Conference in Through-life Engineering Services, University of Bremen, 7th and 8th November 2017.

Klyne, G. and Carroll, J.J. (2004). RDF Resource Description Framework Concepts and Abstract Syntax. W3C recommendation, W3C. `https://www.w3.org/TR/rdf-concepts/\#section-data-model`.

Lemaignan, S., Siadat, A., Dantan, J.Y., and Semenenko, A. (2006). MASON: A Proposal For An Ontology Of Manufacturing Domain. In *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)*, 195–200. IEEE, Prague, Czech Republic. doi:10.1109/DIS.2006.48.

Liu, X.F., Shahriar, M.R., Al Sunny, S.N., Leu, M.C., and Hu, L. (2017). Cyber-physical manufacturing cloud: Architecture, virtualization, communication, and testbed. *Journal of Manufacturing Systems*, 43, 352–364. doi:10.1016/j.jmsy.2017.04.004.

Liu, Y., Wang, L., and Wang, X.V. (2018). Cloud manufacturing: latest advancements and future trends. *Procedia Manufacturing*, 25, 62–73.

Lu, Y., Xu, X., and Xu, J. (2014). Development of a Hybrid Manufacturing Cloud. *Journal of Manufacturing Systems*, 33(4), 551–566. doi:10.1016/j.jmsy.2014.05.003.

Matt, D.T., Rauch, E., and Dallasega, P. (2015). Trends towards Distributed Manufacturing Systems and Modern Forms for their Design. *Procedia CIRP*, 33, 185–190. doi:10.1016/j.procir.2015.06.034.

Tao, F., Zhang, L., Venkatesh, V.C., Luo, Y., and Cheng, Y. (2011). Cloud manufacturing: a computing and service-oriented manufacturing model. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 225(10), 1969–1976. doi:10.1177/0954405411405575.

Tao, F., Zhang, L., Liu, Y., Cheng, Y., Wang, L., and Xu, X. (2015). Manufacturing Service Management in Cloud Manufacturing: Overview and Future Research Directions. *Journal of Manufacturing Science and Engineering*, 137(4), 040912. doi:10.1115/1.4030510.

Wu, D., Greer, M.J., Rosen, D.W., and Schaefer, D. (2013). Cloud manufacturing: Strategic vision and state-of-the-art. *Journal of Manufacturing Systems*, 32(4), 564–579. doi:10.1016/j.jmsy.2013.04.008.

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.