Using an Auction-based System in Cloud Manufacturing for Selecting Manufacturing-as-a-Service Providers

No Author Given

No Institute Given

Abstract. The shift towards mass customization in production and technological advancements lead to new manufacturing paradigms. Such paradigms facilitate the distribution of production to distributed companies, located all around the globe, essentially creating a cloud-based manufacturing platform. In such a system, manufactures may provide their production facilities as services to others – leading to the paradigm of Manufacturing-as-a-Service (MaaS). One major challenge in MaaS is the discovery of manufacturing resources and the distribution of orders to them. We propose an ontology based template matching method to find fitting manufacturing resources. Based on this selection we introduce a multi agent-based auction system for distributing orders and production planning. To illustrate its practical applicability we integrated the proposed approach in an existing cloud manufacturing platform.

Keywords: Manufacturing-as-a-Service · Cloud Manufacturing · Ontologies · Auction-based Production Planning.

1 Introduction

The manufacturing industry has been subject to major changes in recent years as technological advancements are becoming increasingly versatile. The digitization of manufacturing processes elevated these processes from mere physical hardware components to complex Cyber Physical Systems (CPSs) manufacturing including pervasive mining of process data as well as IT supported production planning and control approaches.

The digitization of manufacturing enables the distribution of orders to multiple manufacturing facilities or even multiple manufacturers. From this approach, which is in contrast to conventional manufacturing systems *Cloud Manufacturing (CMfg)* systems emerged. In these systems production of a single order by a customer may potentially involve distributed manufacturers for each manufacturing part of the overall product. This is due to the increasing customizability [8] of products, which makes on-site production considering all variations difficult and, hence, the outsourcing of some production steps to third-party manufacturers inevitable. Essentially, one manufacturer uses services of other manufacturers for producing goods – this is known as *Manufacturing-as-a-Service (MaaS)*.

In MaaS manufacturers register their offered manufacturing services at a central cloud platform, allowing other manufacturers to select them based on specific attributes and characteristics such as price or quality. This selection process typically involves manual steps, which can be more prone to error with the increasing number of available manufacturers. Furthermore scheduling and production planning in CMfg is still not solved [13], and additionally, most approaches are neither flexible nor can they be automated. To take full advantage of CMfg and MaaS, establishing an automated selection of manufacturing services taking changing attributes such as price quota, is desirable.

In this context, we propose a novel approach for production planning for CMfg and MaaS platforms using an ontology-based template matching algorithm. Our approach allows the automatic extraction of fitting manufacturing services from a knowledge base given a set of required and preferred attributes. For the actual selection of a specific service, we propose using an auction-based approach, which takes constraints, e.g. maximum price, defined by the service providers as well as the customers into consideration.

The remainder of this paper is organized as follows: First, we provide background information on technologies to make the paper self-contained and give an overview of relevant literature. Second, we describe our proposed solution and its implementation. Building on this section we demonstrate our findings in a feasibility prototype. Finally, we conclude with a discussion and sketch future work.

2 State of the Art

Bratukhin and Sauter [3] propose the utilization of agents, which handle information in a distributed system, to increase flexibility in manufacturing systems. In contrast to our approach, the agents are fixed and not selected and spawned on demand.

Vogel-Heuser et al. [14] present a distributed production system utilizing CPPS and agents as a possible solution for increased flexibility and adaptability. Agents can also compete with each other to achieve the best outcome for themselves [15]. For example in [1] the authors propose a MAS, which is capable of scheduling multi-project instances via auction-based communication. In this decentralized approach of decision-making, project agents try to allocate resources, which are represented by resource agents, by bidding on free time slots in an auction held by an exchange agent. We build upon this work and utilize these techniques for selecting a single provider using auctions.

Over the years many frameworks for multi-agent systems have been developed. One well-known framework is JADE [2], which uses the Foundation for Intelligent Physical Agents (FIPA) [10] standard for communication between agents. A more novel example would be SARL [11], a framework that utilizes a statically typed programming language to create multi-agent systems.

In Cloud Manufacturing (CMfg) the distribution, as well as the number of available resources affects the scheduling process [6]. A CMfg platform uses network resources, cloud computing and manufacturing enabling technologies in order to provide manufacturer services to its users [17]. In this field multi-agent systems have been used for service discovery [5] as well as other tasks, and are an effective technology for solving issues in scheduling [6].

Ontologies are commonly used in CMfg [16,4,7], e.g., to realize scheduling systems [12]. We use an existing CMfg platform [9] for showing the practical applicability of our proposed approach.

3 Manufacturing Resource Selection

In this section we first sketch the data model needed for representing manufacturing resources and products. We then show how fitting resources can be selected given a set of characteristics and demonstrate how auctions may be used to determine service providers. Finally, we present an integration with an existing CMfg platform for demonstrating the practical applicability.

3.1 Representing Resources and Template Matching

Formalizing manufacturing resources and products is essential for a feasible implementation of a MaaS platform. Typically, this is accomplished via a knowledge base that provides flexible data structures. We create *templates* for manufacturing steps, which are matched against this knowledge base for automatic selection of fitting service providers.

In our knowledge representation, a concrete instance of a product is represented by a *ConfigurableProduct*. Each ConfigurableProduct has specific instances of manufacturing resources assigned to them. These manufacturing resources are represented by a *Operations*, which consists of *Action*s, and can be provided by different manufacturers. Each *Action* is defined by its *input*s and *output*s, which consist of a *name* and a *type*. We propose using Web Ontology Language (OWL) for implementing such a data model. Figure 1 gives an overview of parts of the data model.

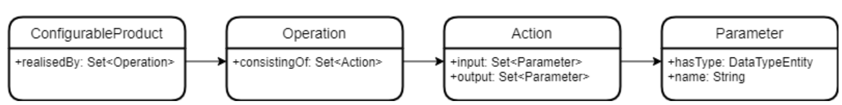

Fig. 1. Overview

Since both, the manufacturing service providers as well as the products use the same data model concepts, we can automatically derive a *template* for selecting resources. The generation of the template is based on an actual operation connected to a product. This template decouples the *Operation*s from a specific product and enables the automated selection of fitting resources. The previous link between an operation and a product is now represented by the template, which serves as a placeholder, and therefore, it may be necessary to enrich it with additional metadata. Essentially, we identify service providers that may substitute the *Actions* specified in the *Action*- and *OrderTemplates*. Figure 2 illustrates the process of selecting eligible *Actions*, where the action in the middle is eliminated as its structure is not fitting.

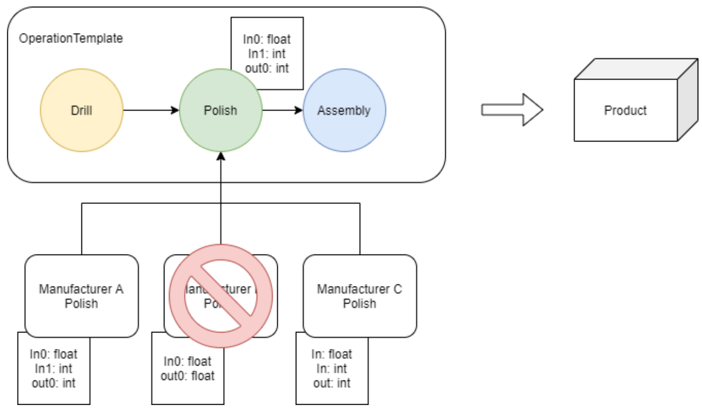

Fig. 2. Selecting fitting resources via template matching

Technically speaking we dynamically create these templates using SPARQL queries. Via this query we are able to filter all fitting structures from the knowledge base and, additionally, filter them by constraints, e.g., price. Each SPARQL query of each ActionTemplate is executed exactly once and returns a list of fitting manufacturing services. An abridged version of the query is shown in Listing 1.1.

```
SELECT ?action WHERE {
    ?action rdf:type cidop:Action;
        cidop:input ?in0;
            ...
        cidop:input ?inN;

        cidop:output ?out0;
            ...
        cidop:output ?outN;

    ?in0 cidop:hasType cidop:integer;
        cidop:name ”inputName0” .
            ...
    ?outN cidop:hasType cidop:string;
        cidop:name ”outputNameN” .

}
```

Listing 1.1. General structure of a SPARQL-Query for *ActionTemplate* matching.

3.2 Auction

To automate the resource selection process we propose using a multi-agent auction system, which includes constraints by customers, e.g., preferred price, due date, or quality, and manufacturers, e.g. lowest acceptable price. According to these constraints, the auction process tries to find appropriate service providers for the individual *Action*s of the product. Hence, an auction is held for each *ActionTemplate* in a given *OperationTemplate*.

A central auction handler coordinates all auctions and sets up new auctions by spawning auctioneer agents for each individual *ActionTemplate*. All previously through the pattern matching process identified *Actions* can participate in the auction and for each one of the *Action*s a representative bidding agent is created. In addition to customer preferences and the bidder configurations by the manufacturers, the auction process depends on the selected auction type, e.g. *English*, *Japanese*, or *Sealed*.

After all auctions have been held for each *ActionTemplate* in an *OperationTemplate*, the results of these auctions, specific *Action*s, are aggregated to form a concrete *Operation*. Because a product may consist of multiple sub-products, and each sub-product may require an *Operation*, an aggregation on this level is necessary as well. The *ConfiguredProduct* is then finalized with the aggregated *Operation*s and can be forwarded to the respective manufacturing scheduling services in the system. By performing auctions for all production steps of a product, the final order may consist of *Action*s of various manufacturers.

3.3 Integration

To apply the methods for template matching and for selecting the most applicable service provider for a production step, a reusable implementation is necessary. We have chosen to implement both parts as single (micro-)services with a well-defined interface. This enables their reuse in different platforms and scenarios, such as CMfg.

To show the practical applicability we integrated our presented services within the existing CIDOP cloud platform [9], as it already utilizes OWL for its knowledge base. Figure 3 gives an overview of its architecture.

This platform uses a central knowledge base implemented in OWL and participating parts are integrated via (micro-)services. They communicate via a message-based paradigm, which makes it easy to embed new services.

We implemented our services using the Java programming languages and utilized SPARQL queries for performing the template matching against the knowledge-base. The implementation of the auction system uses the SARL framework.

3.4 Discussion

While the system produces the optimal results according to the given parameters, e.g., price, work has to be done in order to balance the distribution of orders. For

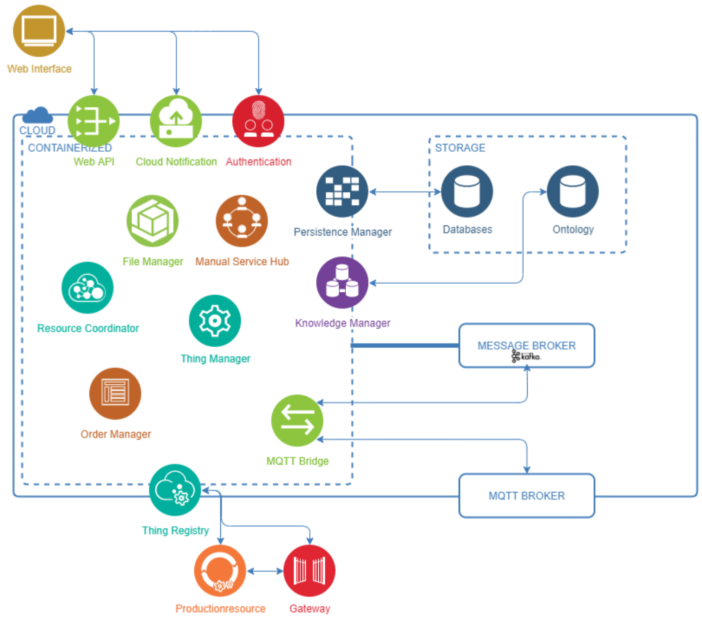

Fig. 3. Conceptual representation of the cloud platform

example, if the same manufacturing resource is constantly selected, some kind of penalty may be required for a more fair distribution of resources. Consequently, manufacturers with fewer orders and slightly less preferable bids would have a chance to be involved in the production of a product.

The parameters for the auctions are fixed for each manufacturing resource. In the future such quotas are to be requested from each service on-demand, which enables the integration of time constraints based on the current workload of a service.

4 Conclusion

With the shift from *Mass Production* to *Mass Customization* the production of goods have to become more flexible. This can be achieved by distributing manufacturing not only to different production plants, but to different companies. A concept supporting the distribution of production is CMfg. In CMfg manufacturers can offer their production capabilities as services creating MaaS. For scheduling production orders in such an environment we use an auction-based approach.

We utilize a template matching approach to select appropriate participants for the auctions. In this paper we identified the knowledge needed to apply a template matching approach based on ontologies and provided the means for automatic matching using SPARQL.

Building upon the selection of fitting resources we created an auction system consisting of a standalone, coordinating service and multiple agents, where each agent represents a service provider participating in the auction. By doing so, we are able to balance customer needs and those of service providers and identify the optimal manufacturer. Furthermore, we can easily integrate additional requirements and support various types of auctions.

The combination of template matching and agent based auctions offers a dynamic and highly scalable approach to the problem of scheduling in a distributed manufacturing environment. Especially, for systems like CMfg this can be a viable solution, though, further work is required.

References

1. Adhau, S., Mittal, M.L., Mittal, A.: A multi-agent system for distributed multi-project scheduling: An auction-based negotiation approach. Engineering Applications of Artificial Intelligence **25**(8), 1738–1751 (Dec 2012). https://doi.org/10.1016/j.engappai.2011.12.003, https://linkinghub.elsevier.com/retrieve/pii/S0952197611002363
2. Bellifemine, F., Poggi, A., Rimassa, G.: JADE – a FIPA-compliant agent framework. In: Proceedings of PAAM. vol. 99, p. 33. London (1999)
3. Bratukhin, A., Sauter, T.: Bridging the gap between centralized and distributed manufacturing execution planning. In: 2010 IEEE 15th Conference on Emerging Technologies & Factory Automation (ETFA 2010). pp. 1–8. IEEE (2010)
4. Cai, M., Zhang, W., Zhang, K.: Manuhub: a semantic web system for ontology-based service management in distributed manufacturing environments. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans **41**(3), 574–582 (2010)
5. Guo, L., Wang, S., Kang, L., Cao, Y.: Agent-based manufacturing service discovery method for cloud manufacturing. The International Journal of Advanced Manufacturing Technology **81**(9), 2167–2181 (2015)
6. Liu, Y., Wang, L., Wang, X.V., Xu, X., Zhang, L.: Scheduling in cloud manufacturing: state-of-the-art and research challenges. International Journal of Production Research **57**(15-16), 4854–4879 (2019)
7. Lu, Y., Xu, X., Xu, J.: Development of a hybrid manufacturing cloud. Journal of Manufacturing Systems **33**(4), 551–566 (2014). https://doi.org/https://doi.org/10.1016/j.jmsy.2014.05.003, https://www.sciencedirect.com/science/article/pii/S027861251400048X
8. Marques, M., Agostinho, C., Zacharewicz, G., Jardim-Gonçalves, R.: Decentralized decision support for intelligent manufacturing in Industry 4.0. Journal of Ambient Intelligence and Smart Environments **9**(3), 299–313 (Apr 2017). https://doi.org/10.3233/AIS-170436, https://www.medra.org/servlet/aliasResolver?alias=iospress&doi=10.3233/AIS-170436
9. Merz, R., Hoch, R., Drexel, D.: A cloud-based research and learning factory for industrial production. Procedia Manufacturing **45**, 215–221 (2020). https://doi.org/https://doi.org/10.1016/j.promfg.2020.04.097, https://www.sciencedirect.com/science/article/pii/S2351978920311392, learning Factories across the value chain – from innovation to service – The 10th Conference on Learning Factories 2020

10. Poslad, S.: Specifying protocols for multi-agent systems interaction. ACM Trans. Auton. Adapt. Syst. **2**(4), 15–es (Nov 2007). https://doi.org/10.1145/1293731.1293735, https://doi.org/10.1145/1293731.1293735
11. Rodriguez, S., Gaud, N., Galland, S.: SARL: A General-Purpose Agent-Oriented Programming Language. In: 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT). pp. 103–110. IEEE, Warsaw, Poland (Aug 2014). https://doi.org/10.1109/WI-IAT.2014.156, http://ieeexplore.ieee.org/document/6928174/
12. Sanko, J., Haav, H.M., Kotkas, V.: Ontology-based customization of a scheduling system for discrete manufacturing. In: DB&IS (Selected Papers). pp. 57–70 (2016)
13. Sokolov, B., Ivanov, D., Dolgui, A.: Scheduling in Industry 4.0 and Cloud Manufacturing. Springer International Publishing (2020). https://doi.org/10.1007/978-3-030-43177-8
14. Vogel-Heuser, B., Diedrich, C., Pantforder, D., Gohner, P.: Coupling heterogeneous production systems by a multi-agent based cyber-physical production system. In: 2014 12th IEEE International Conference on Industrial Informatics (INDIN). pp. 713–719. IEEE, Porto Alegre RS, Brazil (Jul 2014). https://doi.org/10.1109/INDIN.2014.6945601, http://ieeexplore.ieee.org/document/6945601/
15. Weiss, G.: Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. MIT press (1999)
16. Xu, X.: From cloud computing to cloud manufacturing. Robotics and Computer-Integrated Manufacturing **28**(1), 75–86 (2012). https://doi.org/https://doi.org/10.1016/j.rcim.2011.07.002, https://www.sciencedirect.com/science/article/pii/S0736584511000949
17. Zhang, L., Luo, Y., Tao, F., Li, B.H., Ren, L., Zhang, X., Guo, H., Cheng, Y., Hu, A., Liu, Y.: Cloud manufacturing: a new manufacturing paradigm. Enterprise Information Systems **8**(2), 167–187 (Mar 2014). https://doi.org/10.1080/17517575.2012.683812, http://www.tandfonline.com/doi/abs/10.1080/17517575.2012.683812