Towards Process Support for Cloud Manufacturing

Stefan Schulte*, Philipp Hoenisch*, Christoph Hochreiner*, Schahram Dustdar*, Matthias Klusch†, Dieter Schuller‡
*Vienna University of Technology, Austria
Email: {s.schulte, p.hoenisch, c.hochreiner, dustdar}@infosys.tuwien.ac.at

†German Research Center for Artificial Intelligence
Email: Matthias.Klusch@dfki.de

‡Technische Universität Darmstadt, Germany
Email: schuller@kom.tu-darmstadt.de

***Abstract*—Due to increasing competitive pressure, manufacturing companies need to support flexible and scalable business processes – both on the shop floor and in their enterprise software systems. Cloud manufacturing is a recent approach to realize real-world manufacturing processes by applying well-known basic concepts from the field of Cloud computing to this domain.**

To implement Cloud manufacturing, it is necessary to model, enact and monitor according manufacturing processes and virtualize the single process steps. So far, Business Process Management Systems do not explicitly support Cloud manufacturing. This paper analyzes requirements regarding process enactment for Cloud manufacturing and provides a concept for an according software framework.

Keywords*—*Cloud Manufacturing, Elastic Processes, Business Process Management, Cloud Computing

I. INTRODUCTION

Two major trends are currently leading to substantial transformations in the manufacturing industry: First, increasing competition with lower-wage countries forces companies to find new ways to stay in competitive advantage [1]. The organizations are also changing themselves, and manufacturing companies are not only part of sequential, long-term supply chains, but also part of (potentially) extensive manufacturing networks, which require agile collaboration between different organizations [2]. Second, the proliferation of Information and Communication (ICT) and Internet of Things (IoT) technologies provides the manufacturing industry with the means to meet these challenges by realizing manufacturing processes, which are highly flexible and dynamic. Using ICT and IoT technologies, companies are able to realize manufacturing processes which satisfy customer demands for, e.g., large series production, mass customization, changing order situations, and short time-to-market [3].

The potential benefits, but also the arising questions when discussing the manufacturing processes of the future are substantial: Not only in terms of manufacturing assets (e.g., Industrial Cyber Physical Systems – CPS [4], [5]), but especially in terms of the software services, business processes, and enterprise architectures used to support manufacturing. Manufacturing processes may span different organizations and are supported by different computing platforms, e.g., Enterprise Resource Planning (ERP), Manufacturing Execution System (MES), or Product Lifecycle Management (PLM) software.

Companies involved in extensive manufacturing networks need to be able to design, configure, enact, and evaluate a very large number of manufacturing process instances, each representing a different order and supply chain instance. Whereas "traditional" Business Process Management (BPM) orchestration assumes stable and well-defined processes, aspects such as dynamics, efficiency, the increasing complexity and the large amount of real-time data which is available during manufacturing, calls for new solutions [6], [7]. Hence, the support of complex, scalable manufacturing processes by appropriate software solutions is a necessary foundation for the "Factories of the Future" and is therefore a crucial research question [8], [9].

From the business perspective, several projects and papers have observed the composition of manufacturing processes from single services which represent single manufacturing assets or real-world manufacturing processes, e.g., applying the concept of "Virtual Factories" [9] or "Virtual Manufacturing Enterprises" [10]. Naturally, if applying service-based composition to manufacturing processes, these processes can be supported by Service-oriented Architectures (SOAs) [11]–[13]. One very recent approach takes service-oriented manufacturing processes to the next level by proposing to port well-known principles from the field of Cloud computing to real-world manufacturing processes and in turn supporting these processes by Cloud-based software and IT infrastructure. This approach is also known as *Cloud manufacturing* and supports [2], [14], [15]:

- Leasing and releasing manufacturing assets in an on-demand, utility-like fashion,
- Rapid elasticity through scaling leased assets up and down if necessary, and
- Pay-per-use through metered service.

By applying these principles, it is possible to evolve from production-oriented manufacturing processes to service-oriented process networks by modeling single manufacturing assets as services in a similar way as Software-as-a-Service (SaaS) or Platform-as-a-Service (PaaS) are already provided by Cloud providers today [16]. By modeling all process steps and manufacturing assets as services, agile collaboration through flexible and scalable manufacturing processes is facilitated and it is possible to realize cross-organizational manufacturing orchestrations and integrate distributed resources to manufacture products more efficiently. Providers of manufacturing services may offer their capabilities in a pool of configurable manufacturing assets and services. These can then be rapidly provisioned and released with minimal management effort and service provider interaction. By leasing these services, process owners (who want to instantiate a supply chain for a particular product) are able to optimize their manufacturing approach and to react flexibly towards changing situations, including, but not limited to changing order situations or delivery failures by particular suppliers.

While the theoretical foundations for Cloud manufacturing are manifest and easy to follow, to the best of our knowledge, there is a lack of support by Business Process Management Systems (BPMS) and similar software systems. Hence, in this paper, we present a conceptual software framework, which provides the necessary functionalities of a BPMS and is able to quickly scale up and down through the integration of Cloud controller functionalities.

For this, the remainder of this paper is organized as follows: First, we will comment on the related work (Section II) in the fields of Cloud manufacturing and process scalability. Afterwards, we define the scope of the envisioned software framework (Section III) and present the concept itself (Section IV). We assess the usability of the proposed framework by discussing an example scenario and use cases in Section V. Eventually, we identify upcoming research topics and conclude this paper (Section VI).

II. Related Work

Research on the application of Cloud manufacturing and corresponding business processes is still at its beginning. To the best of our knowledge, there is no software framework explicitly aiming at the support of Cloud manufacturing, and existing work focuses on conceptual approaches, analytical work, and the provision of research roadmaps or optimal selection of manufacturing services and computational resources to achieve Cloud manufacturing, e.g., [2], [15], [17]–[19].

Nevertheless, there is work from related fields of research, which should be taken into account: Cloud manufacturing is based on the assumption that in the future, manufacturing assets will be accessible anywhere and traceable in real-time, allowing adaptive decision making based on concurrent simulation and forecasting [3], [5], [20]. The exploitation of such data has also been regarded in different approaches applying SOA and IoT technologies for realizing manufacturing processes, e.g., [13], [21]. Actually, during the last decade, the SOA paradigm has been investigated extensively in the context of initiatives for smart factories and Factories of the Future, e.g., [9], [22]–[24]. More recent approaches take into account the usage of Cloud-based computational resources for the support of service-based manufacturing processes, e.g., [25], [26]. However, in contrast to the work at hand, these approaches do not port basic principles from Cloud computing to the manufacturing domain but use the Cloud only as a technical foundation for interoperability between partners in a supply chain. In addition, there are some approaches to utilize Cloud resources for the execution of general business processes, i.e., software service-based processes without explicitly taking into account the manufacturing domain, e.g., [27].

One particular prerequisite of Cloud manufacturing is flexible and scalable process support. This concept is also known as *elastic BPM* and according processes as *elastic processes* [28]. To the best of our knowledge, there is no work on elastic BPM, which takes into account the specifics of the manufacturing domain. In fact, while there are some other approaches to realize elastic processes, they focus on the scheduling and/or resource allocation of such processes and provide according optimization algorithms, e.g., [29], [30]. Furthermore, Juhnke et al. [31] present an extension to a standard BPEL workflow engine, which allows to use Amazon EC2-based computational resources to execute business processes. Janiesch et al. [32], [33] provide an extensive conceptual model for elastic processes and a fuzzy controller for Amazon EC2. Once again, none of these approaches pays special attention to the manufacturing domain. This is also the case for our own work on the *Vienna Platform for Elastic Processes* (ViePEP, see Section IV and [34]–[37]). For a full discussion of the current state-of-the-art in elastic processes, we refer to [37].

III. Software Support Challenges for Cloud Manufacturing

While the basic assumptions and principles of Cloud manufacturing are evident, the research questions to be answered are numerous and extensive, ranging from data integration issues to actual BPM questions as well as trust and security aspects [2], [15]. We will limit the scope of this paper to particular aspects, namely the actual enactment of virtualized manufacturing processes using BPMS functionalities and Cloud-based computational resources.

In the following paragraphs, we will define the requirements towards a software framework for Cloud manufacturing. A discussion of further topics, which in our opinion are urgently needed to fulfill process support for Cloud manufacturing, can be found in Section VI. The following requirements are based on the assumption that a software framework for Cloud manufacturing needs to support all phases of the BPM lifecycle, i.e., Design & Analysis, Configuration, Enactment, and Evaluation [38].

First, it is necessary to design process models and provide the models to potential customers for instantiation. In fact, a *process model* offered by a manufacturing company reflects a particular product the company is offering and the necessary steps to produce it – this includes the steps provided by other companies. Modeling could be done by either the customer or there could be predefined process models (i.e., products) to be instantiated (see Section IV-A). A *process instance* results from a particular event, e.g., an order from a customer.

Second, the configuration of manufacturing processes includes the need to implement the single process steps. For real-world manufacturing process steps, this means that the steps are virtualized, i.e., a software service representing the process step including a description of the data sources (e.g., sensors) providing monitoring data for a manufacturing task (and therefore process step), needs to be implemented and configured for usage in a particular process step. Notably, configuration includes the scheduling and resource allocation of services to manufacturing assets and computational resources. If necessary and possible, additional resources need to be leased to enact a process instance. Naturally, computational resources should never be a bottleneck, since a forced downtime of manufacturing assets can be very expensive. Hence, the software framework needs to be able to support scalable and flexible computational resources, especially if a company is part of an extensive process landscape. For example, a company could be involved in literally thousands of concurrently running process instances, each representing a different order. The virtualization of single process steps as software service can therefore lead to extensive computational demands, since potentially large amounts of data need to be analyzed to achieve process monitoring. Hence, the software framework and the BPMS that manages the process landscape need to be flexible and scalable. Since Cloud manufacturing follows a service-based approach, it seems naturally to also follow a (Software-as-a-) service-based approach for the necessary software framework, and run the software services in Virtual Machines (VMs). This allows to scale the leased computational resources up or down based on the current demand.

Third, the real-world manufacturing processes need to be enacted. This includes two different aspects – execution of the real-world manufacturing services and of their virtualized software counterparts. Enactment includes the monitoring of the services, e.g., by exploiting CPS or sensor data. We assume that this monitoring is done in a decentralized way, i.e., single supply chain partners analyze monitoring data in the virtualized software services running in their own computational resources or leased Cloud-based computational resources. The goal of monitoring is the identification of significant events, which should be communicated to the customer. If monitoring data indicates that particular process steps are not meeting expected Service Level Objectives (SLOs), the underlying process instance has to be adopted, e.g., by replacing a particular supplier by another one.

Fourth, after (and sometimes during) process enactment, execution logs should be analyzed in terms of optimization aspects or whether there is a need to change process models or instances. For example, if historic log data shows that a particular supplier always delivers a pre-product too late or does not meet quality standards, the supplier should be replaced by another one.

Fifth, while not an explicit part of the BPM lifecycle [38], a BPMS for Cloud manufacturing needs to have a solid knowledge base for the different decisions to be made in every stage of the lifecycle. This includes knowledge about free resources in terms of manufacturing assets and regarding computational resources. Knowledge about internal assets and resources could be stored in a company-internal database, while knowledge about external assets could be provided by marketplaces which are part of a *Manufacturing Business Web*, i.e., a Cloud-based business environment bringing together supply chain partners by providing the necessary infrastructure, applications, content, and connectivity means [39].

IV. A Software Framework for Cloud Manufacturing

As outlined above, a software framework for Cloud manufacturing needs to be able to support the complete business process lifecycle, i.e., provide the functionalities of a BPMS. In addition, it should also be able to lease and release Cloud-based computational resources in a rapid manner so that the software support does not become a bottleneck. Hence, an according software framework also needs to act as a Cloud controller. As mentioned in Section II, business processes executed using Cloud-based computational resources are also known as elastic processes.

As pointed out above, process models are instantiated based on events like orders. In the manufacturing domain, each single process step may take several days, weeks or even more. Therefore, the whole supply chain, including the involved suppliers, has to be planned in advance to calculate the needed duration, amount of resources and potential pitfalls.

For this, a Cloud manufacturing software framework will have to check available real-world manufacturing services from potential suppliers, choose particular services, virtualize them, integrate data from available sources to track and monitor the single process steps, compare expected process behavior (as defined in the SLOs) and actual behavior, and start required countermeasures (such as replacement of a supplier) as soon as delays or other problems are detected. This needs to be done for a potentially extensive process landscape.

So far, scalability and flexibility of manufacturing processes have not been regarded in BPMS. In our former work on ViePEP [34]–[37], we have provided a research BPMS for elastic processes. ViePEP comprises the functionalities of a BPMS and a Cloud controller, but is still focusing on the software level, rather than interacting with external, distributed data sources and representing real-world manufacturing processes. In order to realize inter-organizational processes in an integrated way, BPMSs have to interact with each other and with external data sources. Hence, to address the special characteristics of the manufacturing domain, ViePEP has been extended and refactored, resulting in the conceptual software framework depicted in Figure 1. As it can be seen in the figure, the framework is made up from five top-level components, which will be briefly presented in the following subsections.

A. Client

The *Client* models the business process to be fulfilled to manufacture a certain product. Usually, for off-the-shelf products, such a model is already available and the Client only needs to request its instantiation. The instantiation could also be the result of an external event, e.g., if a customer orders a product, the process instance is automatically created. Notably, there might be several process requests coming into the system simultaneously.

Besides the process, the Client can also specify SLOs to be fulfilled by the process, e.g., a particular delivery time, maximum costs, quality standards, or even the carbon footprint of production. It needs to be noted that a process instance represents both the real-world manufacturing process and the virtualized process, which will be enacted using Cloud-based computational resources. Regarding the potential suppliers, the Client may follow two different approaches: The first one precisely defines which suppliers will be part of the process (tight coupling), while the second one defines only what service needs to be fulfilled. In the latter case, the suppliers will be chosen during process runtime (loose coupling). To choose a particular supplier, semantic matchmaking [40] and rich service descriptions are necessary (see also Section VI).

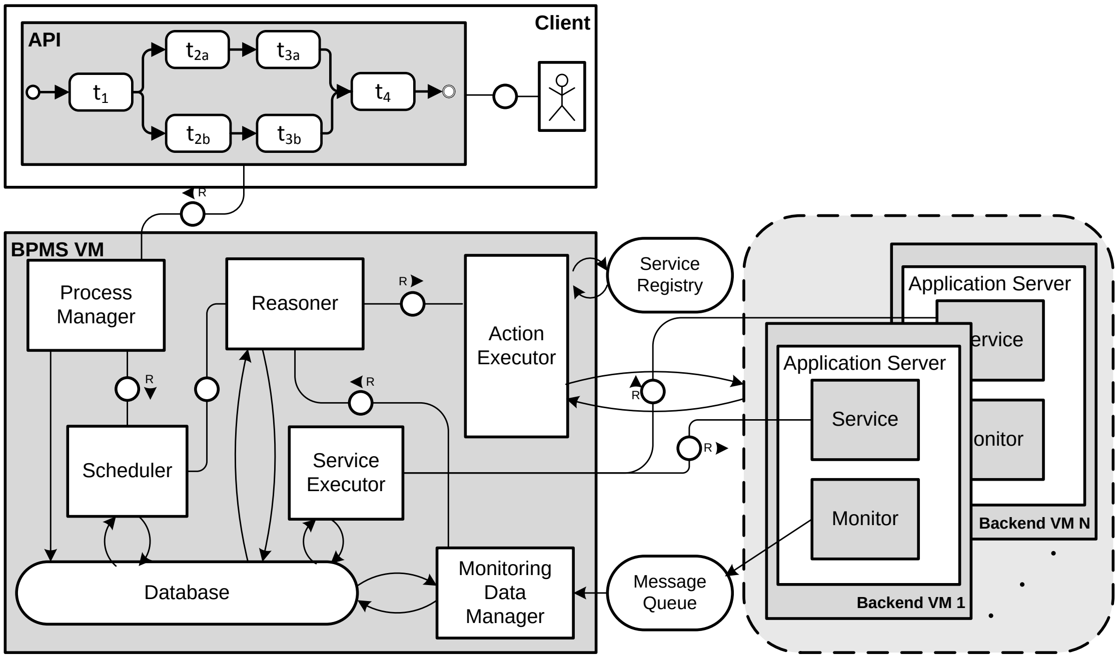

Figure 1: Conceptual Software Framework for Cloud Manufacturing

B. BPMS VM

The *BPMS VM* is the core component of the envisioned software framework. As mentioned before, the underlying BPMS should be able to serve a large number of requests simultaneously. The BPMS will therefore also serve as a Cloud controller and be able to instantiate virtualized manufacturing services on Cloud-based computational resources. Notably, different partners in a supply chain could run their own BPMS instances, which would decrease integration efforts.

The BPMS VM offers different subcomponents, which together realize the needed functionalities to control a Cloud manufacturing process landscape:

The *Process Manager* is the connection point to the Client(s), and exposes its functionality via interface to the outside world, e.g., an order fulfillment system. It serves as the entry point for requests for process instances.

The Process Manager forwards requests to the *Scheduler*, which is responsible for planning the execution of the particular process instance. For this, the scheduler takes into account free resources in terms of manufacturing assets and computational resources. The scheduling plan is again stored in a database. Scheduling is a permanent task, since several events could necessitate to change the schedule, e.g., new process instances are requested, or monitoring data shows that a particular process step will be delayed or cannot be fulfilled at all by a particular supplier.

When a new schedule has been computed, the *Reasoner* is invoked. The Reasoner is responsible for calculating the amount of required (real-world and computational) resources and acquire them in time. For acquiring real-world manufacturing services, the Reasoner needs to check the *Service Registry* (see Section IV-D), which provides information about available services and manufacturing capacities. Scheduling and reasoning about resource allocations are complex tasks on their own, and will not be further discussed in this paper. For an overview on this topic, we refer to [41].

Once the Reasoner has calculated the necessary resource allocation, e.g., additional manufacturing resources are required, or a particular real-world manufacturing service needs to be moved from one supplier to another, required actions are coordinated by the *Action Executor*. For this, the Action Executor needs to interact with BPMS or other software systems of the suppliers. In the future, this will be possible through the envisioned Manufacturing Business Web. Besides, the Action Executor is also the connecting component between the Cloud-based computational resources and the BPMS. It is able to lease or release VMs from Cloud providers and can request the state of the single Backend VMs, e.g., to check if everything is still up and running. Furthermore, the Action Executor is directly connected with each Backend VM and can therefore deploy new services, shut down running services, or move/exchange existing ones.

Within the BPMS VM, the *Service Executor* is responsible for ensuring a faultless execution of the virtualized process instances. If the Service Executor is not able to carry out a service execution, it informs the Scheduler. This mechanism ensures that unsuccessful service executions are considered in process and service scheduling.

The *Monitoring Data Manager* is a utility component and receives monitoring data from the Backend VMs through a *Message Queue* (see Section IV-D). Besides simply storing this data, it is also performing data cleansing and transformation tasks, thus the data is already in the correct format for the Scheduler and Reasoner. In addition, based on service-specific thresholds, the Monitoring Data Manager is able to notify the Reasoner about unexpected events (e.g., delays). If the Reasoner detects a severe deviation, it is able to notify the

Scheduler to arrange a reorganization of the schedule or acquire new additional resources.

C. Backend VM

The *Backend VM* represents a generic standalone VM image, which can be instantiated and hosted on common Cloud providers such as Amazon EC2 or a privately hosted OpenStack-based Cloud. The Backend VM hosts an *Application Server* on which virtualized software services are deployed. These services are invokable from a remote location via according interfaces. Besides the service, a *Monitor* is deployed and responsible for monitoring the service with regard to the needed computing resources such as CPU, memory and storage. This information is sent to the BPMS VM in a regular interval via the Message Queue and can be used to monitor the status of the leased Backend VMs.

The service(s) running on Backend VMs represent virtualized manufacturing services from the real world. While virtualization and description of such services is not in the focus of the work at hand, it should be noted that this is nevertheless one of the key success factors to achieve Cloud manufacturing.

D. Service Registry and Message Queue

The *Message Queue* and *Service Registry* are utility components providing basic functionalities needed by the BPMS VM and Backend VMs. The Message Queue allows communication between the BPMS VM and Backend VMs across organizational boundaries. The Service Registry hosts both information about real-world manufacturing assets and about the virtualized software services. For the former, the Service Registry for now acts as a substitute to marketplaces in the Manufacturing Business Web, but in the future, either a marketplace could be directly integrated or the Service Registry could include information from such marketplaces. Regarding the virtualized software services, the Service Registry hosts single, configurable services as deployable archives. These archives can then be deployed on a Backend VM.

V. Discussion

To discuss Cloud manufacturing enabled by our conceptual software framework, we describe an example scenario, using *JD-Company*, a United Kingdom-based company and manufacturer of car seats, as an example. Figure 2 shows a simplified manufacturing process of JD-Company. Apart from their factory in the United Kingdom, JD-Company also operates a plant in Germany, which provides intermediary steps to the manufacturing process, e.g., springs. In addition, JD-Company gets pre-products from their suppliers, e.g., covers or cushioning. JD-Company offers both off-the-shelf seats as well as customized products. While the former ones are usually ordered in large quantities and with some lead time, customized seats are often ordered with very short time for delivery and in small quantities, down to orders for single seats. As a typical Small and Medium-sized Enterprise (SME), JD-Company is subject to budget constraints and therefore their warehouse does not hold all possible parts for all seat models. Instead, parts are ordered from suppliers when an order arrives. In turn, the suppliers often have to produce the ordered items themselves, depending on the nature of the products.

Notably, Figure 2 shows only a small excerpt of the manufacturing network JD-Company is part of. In reality, the company runs a large number of concurrent process instances for different customers, and subsequently interacts with a large number of suppliers. Also, the figure shows a simplified scenario with a relatively small number of suppliers and process steps; to simplify matters, more complex process patterns (e.g., XOR, AND splits, or loops [42]) are not depicted.

Example Scenario – Process Modeling and Instantiation: Following the principles of Cloud manufacturing, JD-Company is able to model all their products as service-based manufacturing processes, and instantiate a process model whenever a new order arrives (Step 1 in Figure 2). Notably, process instances both represent orders, which contain a number of the same seats (in case of mass orders), or single seats in case a customized seat has been ordered. For each step in a process model, JD-Company is able to define SLOs they require their suppliers to fulfill, e.g., production or delivery times, maximum costs, quality standards, or carbon footprints. The concrete suppliers are automatically chosen based on the predefined SLOs (loose coupling) or a particular supplier is manually chosen (tight coupling), e.g., because JD-Company does not want to replace a long-term business partner.

For the particular example process from Figure 2, the first two process tasks are loosely coupled, i.e., the software framework running the manufacturing processes automatically checks the available services of potential suppliers for tasks t_1 and t_2 (Step 2). These services represent real-world manufacturing assets or services and are offered in a shared pool of resources. This shared pool of resources is realized by an Internet-based marketplace (not explicitly depicted in Figure 2), which is part of the (envisioned) Manufacturing Business Web. On this marketplace, real-world manufacturing assets and services can be advertised, leased and released. In order to provide end-to-end integration of the BPMS and other software systems of supply chain partners, the marketplace needs to provide the means to automatically integrate services into the manufacturing processes of a company. For this, the marketplace needs to provide middleware capabilities, especially the possibility to wrap different kinds of data sources and provide this data in a unified format (see Section VI).

Based on the chosen services, JD-Company's BPMS will send appropriate order requests to the BPMS of the suppliers (Step 3). The other tasks (t_3 and t_4) are tightly coupled and not automatically chosen, since they are provided by JD-Company itself, however at different geographic locations.

Only once the potential suppliers have accepted all orders, the actual process is instantiated and the virtualized software services, which are required to monitor the real-world manufacturing services, are integrated (Step 4). The virtualized software services are provided by computational resources of the partners, however, in the future, partners may even share their computational resources in a "Community Cloud" [14]. For now, we assume that every partner has to provide sufficient computational resources on its own. If the partners run a corresponding software framework, this will be done automatically – in Figure 2, this is the case for JD-Company

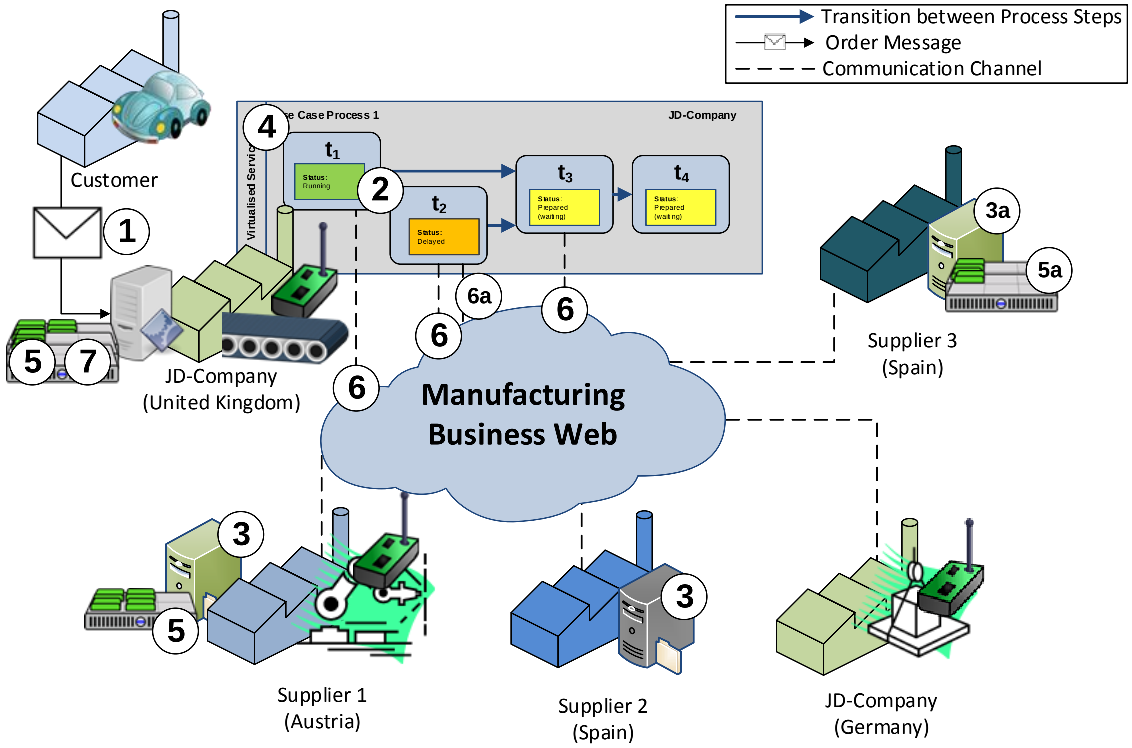

Figure 2: Use Cases 1–3

and Supplier 1 (Step 5), but Supplier 2 has to take care of this by its own means. Notably, JD-Company is able to run its software services on the same VMs for both the German- and UK-based factories.

Example Scenario – Process Execution and Monitoring: Once processes have been instantiated, they have to be monitored and their progress has to be checked against predefined metrics, e.g., if certain suppliers of JD-Company are not able to deliver in time, this should be recognized and according countermeasures should be started. Monitoring of process executions could be done using IoT technologies, e.g., by monitoring data from embedded CPS sensors or smart objects via Wireless Sensor Networks (WSNs), RFID, or other technologies [2], [20]. In the figure, this integration and monitoring is indicated by Step 6.

Based on this data, supply chain partners will recognize if a particular product has already been manufactured or deduce from its location the current status of delivery. It should be noted that IoT-based monitoring could be done both on an inter- and intra-organizational level, i.e., JD-Company is not only able to track processes within their own factories, but also in the factories of their suppliers, based on predefined monitoring protocols.

Regarding the process instance depicted in the figure, Supplier 2's service is late, which could delay the overall process, as this pre-product is needed in JD-Company's German factory for the next process step. Taking this information into account, JD-Company is able to reschedule the process instance, using the machine slots or human resources originally planned for this particular process instance for other production tasks first. If the delay is severe, i.e., JD-Company itself might not be able to deliver the finished seats in time to their customers because of the delay from Supplier 2, the service will be re-allocated to another supplier (Step 7). Based on this, the software framework carries out a re-planning, i.e., Steps 2-4 are carried out again.

When checking for available services from other potential suppliers, Supplier 3 is identified as the best fit and fortunately, this supplier has spare resources to fulfill the complete order. Hence, it is integrated into the process instance. For this, in a similar vein as in Steps 3, 5 and 6, Steps 3a, 5a and 6a are implemented for Supplier 3. Step 4 is analog to the scenario described above.

This simple scenario shows already the three major Cloud manufacturing principles as mentioned in Section I: (i) JD-Company leases and releases manufacturing assets directly and without any prior commitment. (ii) JD-Company is able to scale the leased assets up or down if the order situation changes. (iii) JD-Company pays only if a particular product has been delivered.

VI. Conclusion

Cloud manufacturing is a recent advancement of the idea to model manufacturing business processes based on virtualized services. So far, the basic approach to Cloud manufacturing has been defined, but there is a lack of software frameworks supporting it. In this paper, we have discussed which software functionalities need to be provided in order to realize Cloud manufacturing process landscapes. Based on this discussion, we have conceptualized an according software framework.

Within the example use case scenario, we have shown how such a software frameworks supports typical Cloud manufacturing situations, with a focus on process enactment. So far, the focus of our work was on the basic BPMS and elasticity properties, while other aspects – most importantly the integration of data sources – have not been covered yet.

Even though we restrict the focus of the paper at hand to BPMS and Cloud controller functionalities, we want to highlight related research questions, which in our opinion are of primary importance and need to be solved in order to realize Cloud manufacturing. The following list is by no means exhaustive, but represents our impression of the most relevant research challenges at the time of writing this paper:

(i) *Process Modeling and Service Descriptions:* As a foundation for process modeling but also monitoring (see below), it is necessary to describe virtualized manufacturing assets and manufacturing services using a rich semantic model including their functional, non-functional, and technical capabilities. Hence, to establish Cloud manufacturing, it is necessary to build new standards or extend existing standards, e.g., the *Standard for the Exchange of Product model data* (STEP), which is an existing ISO standard for the exchange of manufacturing information [43]. Also, work from related fields, e.g., material descriptions or ontologies for inter-enterprise collaboration or supply chain modeling [44]–[46], needs to be taken into account.

(ii) *Service Marketplaces:* While the idea of service marketplaces is not new, existing marketplaces usually do not offer the means to seamlessly integrate offered services into business processes [47], [48]. To decrease integration efforts, marketplaces should actually provide middleware functionalities realizing interoperability between heterogeneous software systems, e.g., "traditional" BPMS or MES. This is a very important aspect to allow the envisioned end-to-end integration of leased services and their enactment and monitoring. Furthermore, there are aspects like trust (see below) or even financial strength of potential suppliers, which need to be regarded; these aspects have been omitted in work on service marketplaces so far.

(iii) *Process Monitoring:* For the monitoring of process instances, it is first necessary to integrate data from technologically heterogeneous data sources, which is a distinct research topic on its own. Once data is available, it should be automatically integrated in process monitoring and analyzed. Not every piece of monitoring data is significant, and therefore, it is necessary to identify relevant events and assign them to process steps [49]–[51].

(iv) *Trust and Data Security:* Trust, privacy and data security issues are among the most-often named key challenges in Cloud computing and BPM [52]–[54]. The Cloud manufacturing concept (or even the more general idea of a Manufacturing Business Web) can only succeed in practical settings if appropriate mechanisms ensure data security and establish trust between partners.

In our own future research, we will primarily focus on process enactment aspects. Most importantly, Cloud manufacturing can only be realized if supply chain partners are able to get real-time data from technologically heterogeneous sources and this data is easily exploitable. Hence, it is necessary to realize "Manufacturing Data-as-a-Service", i.e., despite different data formats and technologies applied, process-related data should be automatically integrated and exploited.

ACKNOWLEDGMENTS

This work is supported in part by TU Vienna research funds, by the Austrian Science Fund (FWF): P23313-N23, and by the Commission of the European Union within the ADVENTURE FP7-ICT project (Grant agreement no. 285220).

REFERENCES

[1] S. Dowlatshahi and Q. Cao, "The relationships among virtual enterprise, information technology, and business performance in agile manufacturing: An industry perspective," *European Journal of Operational Research*, vol. 174, no. 2, pp. 835–860, 2006.

[2] X. Xu, "From cloud computing to cloud manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 28, pp. 75–86, 2012.

[3] D. Zuehlke, "SmartFactory – Towards a factory-of-things," *Annual Reviews in Control*, vol. 34, pp. 129–138, 2010.

[4] E. A. Lee, "Cyber Physical Systems: Design Challenges," in *11th IEEE Symposium on Object Oriented Real-Time Distributed Computing (ISORC'08)*. IEEE, 2008, pp. 363–369.

[5] A. Fisher, C. A. Jacobson, E. A. Lee, R. M. Murray, A. L. Sangiovanni-Vincentelli, and E. Scholte, "Industrial Cyber-Physical Systems – iCyPhy," in *Fourth International Conference on Complex Systems Design & Management (CSD&M 2013)*. Springer, 2013, pp. 21–37.

[6] R. Breu, S. Dustdar, J. Eder, C. Huemer, G. Kappel, J. Köpke, P. Langer, J. Mangler, J. Mendling, G. Neumann, S. Rinderle-Ma, S. Schulte, S. Sobernig, and B. Weber, "Towards Living Inter-Organizational Processes," in *15th IEEE Conference on Business Informatics (CBI 2013)*. IEEE, 2013, pp. 363–366.

[7] A. Norta, P. Grefen, and N. C. Narendra, "A reference architecture for managing dynamic inter-organizational business processes," *Data & Knowledge Engineering*, vol. 91, pp. 52–89, 2014.

[8] D. T. Matt and E. Rauch, "Design of a Network of Scalable Modular Manufacturing Systems to Support Geographically Distributed Production of Mass Customized Goods," *Procedia CIRP*, vol. 12, pp. 438–443, 2013.

[9] S. Schulte, D. Schuller, R. Steinmetz, and S. Abels, "Plug-and-Play Virtual Factories," *IEEE Internet Computing Magazine*, vol. 16, no. 5, pp. 78–82, 2012.

[10] Y. Y. Yusuf, M. Sarhadi, and A. Gunasekaran, "Agile manufacturing: The drivers, concepts and attributes," *International Journal of Production Economics*, vol. 62, no. 1, pp. 33–43, 1999.

[11] R. Hans, S. Zöller, S. Abels, A. Miede, and R. Steinmetz, "Enabling Collaboration in Virtual Manufacturing Enterprises with Cloud Computing," in *19th Americas Conference on Information Systems (AMCIS 2013)*. Association for Information Systems (AIS), 2013.

[12] C. A. Marín, P. Vrba, P. Leitão, L. Mönch, G. V. Lioudakis, D. Kazanskaia, V. I. Chepegin, L. Liu, J. Barbosa, A. Tsarev, P. Skobelev, and N. Mehandjiev, "Application of intelligent service bus in a ramp-up production context," in *Industrial Track of the Conference on Advanced Information Systems Engineering 2013 (CAiSE'13)*, ser. CEUR Workshop Proceedings, vol. 1017. CEUR-WS.org, 2013, pp. 33–40.

[13] P. Spiess, S. Karnouskos, D. Guinard, D. Savio, O. Baecker, L. M. S. de Souza, and V. Trifa, "SOA-Based Integration of the Internet of Things in Enterprise Services," in *7th IEEE International Conference on Web Services (ICWS 2009)*. IEEE, 2009, pp. 968–975.

[14] P. Mell and T. Grance, *The NIST Definition of Cloud Computing*. Recommendations of the National Institute of Standards and Technology, 2011.

[15] D. Wu, M. J. Greer, D. W. Rosen, and D. Schaefer, "Cloud manufacturing: Strategic vision and state-of-the-art," *Journal of Manufacturing Systems*, vol. 32, no. 4, pp. 564–579, 2013.

[16] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility," *Future Generation Computing Systems*, vol. 25, no. 6, pp. 599–616, 2009.

[17] X. V. Wang and X. W. Xu, "An interoperable solution for Cloud manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 29, pp. 232–247, 2013.

[18] F. Tao, Y. Laili, L. Xu, and L. Zhang, "FC-PACO-RM: A Parallel Method for Service Composition Optimal-Selection in Cloud Manufacturing System," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 2023–2033, 2013.

[19] Y. Laili, F. Tao, L. Zhang, and B. R. Sarker, "A study of optimal allocation of computing resources in cloud manufacturing systems," *International Journal of Advanced Manufacturing Technology*, vol. 63, no. 5–8, pp. 671–690, 2012.

[20] S. Zöller, R. Hans, D. Schuller, U. Lampe, and R. Steinmetz, "Integrating Smart Objects as Data Basis in Virtual Enterprise Collaborations," in *Enterprise Interoperability VI – Interoperability for Agility, Resilience and Plasticity of Collaborations*. Springer, 2014, pp. 297–305.

[21] F. Jammes and H. Smit, "Service-oriented paradigms in industrial automation," *IEEE Transactions on Industrial Informatics*, vol. 1, no. 1, pp. 62–70, 2005.

[22] W. He and L. Xu, "Integration of Distributed Enterprise Applications: A Survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 35–42, 2014.

[23] U. Rauschecker, M. Meier, R. Muckenhirn, A. Yip, A. Jagadeesan, and J. Corney, "Cloud-Based Manufacturing-as-a-Service Environment for Customized Products," in *eChallenges e-2011*. IEEE, 2011, pp. 195–204.

[24] O. F. Valilai and M. Houshmand, "A collaborative and integrated platform to support distributed manufacturing system using a service-oriented approach based on cloud computing paradigm," *Robotics and Computer-Integrated Manufacturing*, vol. 29, pp. 110–127, 2013.

[25] I. Mezgar and U. Rauschecker, "The challenge of networked enterprises for cloud computing interoperability," *Computers in Industry*, vol. 65, no. 4, pp. 657–674, 2014.

[26] M. Khalfallah, N. Figay, C. F. D. Silva, and P. G. (forthcoming), "A cloud-based platform to ensure interoperability in aerospace industry," *Journal of Intelligent Manufacturing*, vol. NN, no. NN, 2014.

[27] G. Stuermer, J. Mangler, and E. Schikuta, "Building a Modular Service Oriented Workflow Engine," in *2nd IEEE International Conference on Service Oriented Computing and Applications (SOCA 2009)*. IEEE, 2009, pp. 1–4.

[28] S. Dustdar, Y. Guo, B. Satzger, and H. L. Truong, "Principles of Elastic Processes," *IEEE Internet Computing*, vol. 15, no. 5, pp. 66–71, 2011.

[29] M. Xu, L.-Z. Cui, H. Wang, and Y. Bi, "A Multiple QoS Constrained Scheduling Strategy of Multiple Workflows for Cloud Computing," in *IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA 2009)*. IEEE, 2009, pp. 629–634.

[30] K. Bessai, S. Youcef, A. Oulamara, and C. Godart, "Bi-criteria strategies for business processes scheduling in cloud environments with fairness metrics," in *IEEE 7th International Conference on Research Challenges in Information Science (RCIS 2013)*. IEEE, 2013, pp. 1–10.

[31] E. Juhnke, T. Dörnemann, D. Bock, and B. Freisleben, "Multi-objective Scheduling of BPEL Workflows in Geographically Distributed Clouds," in *4th International Conference on Cloud Computing (CLOUD 2011)*. IEEE, 2011, pp. 412–419.

[32] C. Janiesch, I. Weber, J. Kuhlenkamp, and M. Menzel, "Optimizing the Performance of Automated Business Processes Executed on Virtualized Infrastructure," in *47th Hawaii International Conference on System Sciences (HICSS 2014)*. IEEE, 2014, pp. 3818–3826.

[33] S. Euting, C. Janiesch, R. Fischer, S. Tai, and I. Weber, "Scalable Business Process Execution in the Cloud," in *2014 IEEE International Conference on Cloud Engineering (IC2E 2014)*. IEEE, 2014, pp. 175–184.

[34] S. Schulte, P. Hoenisch, S. Venugopal, and S. Dustdar, "Introducing the Vienna Platform for Elastic Processes," in *Performance Assessment and Auditing in Service Computing Workshop (PAASC 2012) at 10th International Conference on Service Oriented Computing (ICSOC 2012)*, ser. LNCS, vol. 7759. Springer, 2013, pp. 179–190.

[35] P. Hoenisch, S. Schulte, S. Dustdar, and S. Venugopal, "Self-Adaptive Resource Allocation for Elastic Process Execution," in *6th International Conference on Cloud Computing (CLOUD 2013)*. IEEE, 2013, pp. 220–227.

[36] P. Hoenisch, S. Schulte, and S. Dustdar, "Workflow Scheduling and Resource Allocation for Cloud-based Execution of Elastic Processes," in *6th IEEE International Conference on Service Oriented Computing and Applications (SOCA 2013)*. IEEE, 2013, pp. 1–8.

[37] S. Schulte, D. Schuller, P. Hoenisch, U. Lampe, S. Dustdar, and R. Steinmetz, "Cost-Driven Optimization of Cloud Resource Allocation for Elastic Processes," *International Journal of Cloud Computing*, vol. 1, no. 2, pp. 1–14, 2013.

[38] M. Weske, *Business Process Management: Concepts, Languages, Architectures*, 2nd ed. Springer, 2012.

[39] M. Taisch, B. Stahl, and G. Tavola, "ICT in Manufacturing: Trends and Challenges for 2020 – An European View," in *IEEE 10th International Conference on Industrial Informatics (INDIN 2012)*. IEEE, 2012, pp. 941–946.

[40] S. Schulte, U. Lampe, M. Klusch, and R. Steinmetz, "COV4SWS.KOM: Information Quality-aware Matchmaking for Semantic Services," in *9th Extended Semantic Web Conference (ESWC 2012)*, ser. LNCS, vol. 7295. Springer, 2012, pp. 499–513.

[41] M. L. Pinedo, *Scheduling – Algorithms, and Systems*, 4th ed. Springer, New York, 2012.

[42] W. M. P. van der Aalst, A. H. M. ter Hofstede, B. Kiepuszewski, and A. P. Barros, "Workflow Patterns," *Distributed and Parallel Databases*, vol. 14, no. 1, pp. 5–51, 2003.

[43] M. J. Pratt, "Introduction to ISO 10303–the STEP standard for product data exchange," *Journal of Computing and Information Science in Engineering*, vol. 1, no. 1, pp. 102–103, 2001.

[44] T. Rahmani, S. Bougain, and D. Gerhard, "Ontology-based integration of heterogeneous material data resources in Product Lifecycle Management," in *IEEE International Conference on Systems, Man, and Cybernetics (SMC 2013)*. IEEE, 2013, pp. 4589–4594.

[45] H. Lin and J. Harding, "A manufacturing system engineering ontology model on the semantic web for inter-enterprise collaboration," *Computers in Industry*, vol. 58, pp. 428–437, 2007.

[46] T. Grubic, I. Veza, and B. Bilic, "Integrating process and ontology to support supply chain modelling," *International Journal of Computer Integrated Manufacturing*, vol. 24, no. 9, pp. 847–863, 2011.

[47] J. Cardoso, K. Voigt, and M. Winkler, "Service Engineering for the Internet of Services," in *10th International Conference on Enterprise Information Systems (ICEIS 2008)*, ser. LNBIP, vol. 19. Springer, 2008, pp. 15–27.

[48] D. Oberle, N. Bhatti, S. Brockmans, M. Niemann, and C. Janiesch, "Countering Service Information Challenges in the Internet of Services," *Business & Information Systems Engineering*, vol. 1, no. 5, pp. 370–390, 2009.

[49] S. Zöller, M. Wachtel, F. Knapp, and R. Steinmetz, "Going All the Way – Detecting and Transmitting Events with Wireless Sensor Networks in Logistics," in *38th Annual IEEE Conference on Local Computer Networks (LCN 2013), Workshop Proceedings*. IEEE, 2013, pp. 39–47.

[50] N. Herzberg, A. Meyer, and M. Weske, "An Event Processing Platform for Business Process Management," in *17th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2013)*. IEEE, 2013, pp. 107–116.

[51] C. Janiesch, M. Matzner, and O. Müller, "Beyond process monitoring: a proof-of-concept of event-driven business activity management," *Business Process Management Journal*, vol. 18, no. 4, pp. 625–643, 2012.

[52] R. Moreno-Vozmediano, R. S. Montero, and I. M. Llorente, "Key challenges in cloud computing: Enabling the future internet of services," *IEEE Internet Computing*, vol. 17, no. 4, pp. 18–25, 2013.

[53] T. H. Noor, Q. Z. Sheng, S. Zeadally, and J. Yu, "Trust Management of Services in Cloud Environments: Obstacles and Solutions," *ACM Computing Surveys*, vol. 46, no. 1, p. 12, 2013.

[54] A. Miede, G. Simsek, S. Schulte, D. F. Abawi, J. Eckert, and R. Steinmetz, "Revealing Business Relationships – Eavesdropping Cross-organizational Collaboration in the Internet of Services," in *10. Internationale Tagung Wirtschaftsinformatik (WI 2011)*, 2011, p. 7.