

CHALLENGES AND ARCHITECTURAL APPROACHES FOR AUTOMOTIVE PLM IN MULTI-BRAND ORGANISATIONS - A DISCUSSION PAPER

Kehl, Stefan (2); Hesselmann, Carsten (1); Stiefel, Patrick (2); Müller, Jörg P. (1)

1: Clausthal University of Technology; 2: Volkswagen group

ABSTRACT

Today, top-down processes, centralized IT infrastructures, and one-vendor strategies prevail in Product Lifecycle Management (PLM) of large multi-brand Original Equipment Manufacturer (OEM) groups. Given the usually decentralized organisation and structures and processes that emerge from cross-brand collaboration, these centralized approaches are challenging the adaptiveness and performance of the OEM groups.

In this concept paper, we investigate challenges for cross-brand and cross-domain cooperation from the perspective of processes and IT systems. The main contribution of this paper is that we motivate and outline a novel technical architecture approach combining service-orientation with an event-driven software architecture and asynchronous event processing to support users from different brands and domains in their collaboration along the development process. We analyse related work on collaboration models as well as on event processing and discuss our approach before the background of the state of the art. Finally, we summarize our findings and give an outlook to future research venues.

Keywords: Process modelling, Event Processing, Collaboration, Integrated product development, Product Lifecycle Management (PLM)

Contact:

Hesselmann, Carsten
Clausthal University of Technology
Institute of Informatics
Germany
carsten.hesselmann@tu-clausthal.de

Cite this article: Kehl, S., Hesselmann, C., Stiefel, P., Müller, J.P. (2019) 'Challenges and Architectural Approaches for Automotive PLM in Multi-Brand Organisations - A Discussion Paper', in *Proceedings of the 22nd International Conference on Engineering Design (ICED19)*, Delft, The Netherlands, 5-8 August 2019. DOI:10.1017/dsi.2019.115

1 INTRODUCTION

The first IT solutions for storing and managing product data during development (labelled *Product data management* – PDM) evolved in the 1980s. They were largely limited to the documentation of local, function-specific development and design activities. More recently, Product Lifecycle Management (PLM) extended PDM by downstream tasks such as production process planning, aftersales, or customer relationship management (Deuter and Rizzo, 2016).

In contrast to PDM, PLM strategies contain tools and mechanisms to coordinate not just to support various product design roles, but also a large number of heterogeneous roles and actors with a large diversity of capabilities and specializations. These may refer to (i) different stages of the product development process, (ii) diverse disciplines (like mechanics, software etc.) or (iii) individual component suppliers along the value creation chain.

Today, the complexity of products that need to be managed using PLM strategies has reached a very high level (Durão *et al.*, 2016), and is likely to increase in the future (Terzi *et al.*, 2010). Main drivers of complexity are diverse customer demands and resulting variance within products. In fact, the number of theoretically possible product variants exceeds the number of vehicles actually sold (Kees and Seibert, 2010). In order to cope with the large range of products, many specialized actuators are necessary (Hesselmann *et al.*, 2017), which creates the need for additional specialized authoring systems, increasing overall software architecture complexity.

In this paper, we argue that before the background of these trends, centralized and monolithic software architectures are no longer the best choice for supporting PLM processes. One reason for this being that interactions between the heterogeneous, geographically and organizationally dispersed set of actors cannot be fully modelled in traditional sequential business processes, as they potentially have to respond to unforeseen and/ or unknown *events*, i.e. changes on components due to a failure. Thus, development processes are largely non-deterministic, concurrent, and event-driven streams of actions, which in turn are not adequately supported by conventional software architectures such as centralized client-server models (Bruns and Dunkel, 2010, p. 25). Multi-brand Original Equipment Manufacturer (OEM) groups such as Volkswagen group or PSA, where actors spread across different brands, are particularly affected by these problems, and suffer from restricted ability and timeliness in reacting to critical events.

In this discussion paper, we investigate challenges for cross-brand and cross-domain cooperation at Automotive OEMs from a practical point of view, including process challenges (Section 2.1) as well as resulting technology-related challenges Section 2.2 arising at OEMs. In addition, in Section 2.3 we describe observations from industrial practice of attempts to cope the challenges. In Section 3, we analyse related work on collaboration models as well as on Event Processing (EP). Section 4 then outlines a technical approach combining service-orientation with an event-driven software architecture and EP to support users from different domains along the development process. Finally, Section 5 summarizes the findings of this paper and provides an outlook on our future research priorities.

2 PLM CHALLENGES IN CROSS-BRAND AUTOMOTIVE OEMS

The decentralized organizational structure of OEMs in conjunction with largely centralized IT and software / systems architecture creates different challenges on the level of processes as well as on the level of IT systems. In this section, we discuss some important challenges. We start from some process-related observations in cross-brand OEMs in Section 2.1 and identify resulting technology-related challenges affecting the state-of-the-art Service-Oriented Architectures (SOAs) approaches.

2.1 Process-oriented challenges

Today, PLM processes at multi-brand Automotive OEMs can be structured at three interacting levels:

1. The first level describes the different departments / functions in one single brand that participate in a common product development process (PDP).
2. The second level describes organizational structure of the multi-brand OEM group and interactions between the different brands within the group (e.g. Porsche, Audi, or VW in the case of the Volkswagen group).

3. The third level describes the product models and data in different complexities (e.g. functions, systems, hardware, software, hardware assembly sequences, hardware-software-combinations,...), i.e., technically speaking, describes the basic business objects (BOs) that act as the cross-links and synchronization points of the cross-brand PLM interactions.

This overall structure has mostly historically evolved, and is reflected in widely decentralized processes, responsibilities and data arrangements in individual brands. Due to this situation, multi-brand OEMs run into problems when trying to apply conventional PLM introduction scenarios. On the one hand, there are brand specific legacy structures that have been growing over decades in traditional car companies and that cannot be shut down from one day to the next. On the other hand, the overall size of a multi-brand OEM and the resulting complex collaboration sequences between a central group division and its brands creates considerable inertia. Specifically, we identified the following problems at the three levels and derive challenges from them:

At the first level the functions (e.g. requirements engineering, technical product descriptions, design, strak or concept development, or serial phase execution) are isolated across the lifecycle. Each function creates its own output based on a given input of an upstream department and works on their own experiences and product data. This often ends in a data hunting adventure.

Challenge 1: Connect the specialized functions across up- and downstream processes.

On the level of cross-brand collaboration we observed that workflows between two or more brands (or sometimes especially between a brand and its suppliers) do normally not match together (Hesselmann *et al.*, 2017).

Challenge 2: Deal with heterogeneous workflows in order to improve cross-brand collaboration.

For the BO dimension, dependent on which BO has to be shared between the collaboration participants, lack of uniformity, consistency, and timeliness in the content of the BO often leads to misinterpretation and/or high numbers of iterations in individual process steps.

Challenge 3: Improve the comparability and the linkability of business objects.

In the following subsection, we study some technology-oriented challenges and opportunities for addressing the process-related issues discussed so far.

2.2 Technology-oriented challenges

The BOs mentioned above are often created in specialized application systems, such as CAD files (computer-aided design) in a corresponding application. All systems used for a specific field (e. g. PDM) in a company, including possible interfaces to each other, form an *enterprise system landscape*.

Usually enterprise system landscapes are not completely planned out in a top-down fashion; on the contrary, such landscapes have grown historically and the individual systems are usually interconnected pairwise via a multitude of different interfaces. The development and launch of new systems is driven by the respective business areas in a decentralized manner in order to remedy short to medium term problems. This operationally driven software development often means that no central IT strategy is pursued and therefore different systems are often poorly or not at all coordinated.

Especially in multi-brand OEM environments, the absence of such strategies leads to a sprawling diversification regarding highly specialized and domain-specific systems (Kehl *et al.*, 2016). From an overall perspective, each of the systems contains some of the data needed for the complete description of the products being manufactured. In order to be able to determine the current state of the development process at any time, it is necessary to be able to relate these distributed product data and process them together. In abstract terms, the following challenge can thus be formulated.

Challenge 4: Relate product data that is distributed over heterogeneous systems.

A well-known and technology-neutral approach to orchestrate distributed systems and the data they contain are Service-Oriented Architectures (SOAs). However, a drawback with a classic SOA is its strict top-down approach, where a *middleware* service is called from within a business process. Next, the middleware coordinates the calls to the *services provided* by the *distributed systems* (see figure 1). Unfortunately, the classic SOA approach is not very applicable given the characteristics of cross-brand PLM processes as discussed in Section 2.1, because these processes are neither synchronous

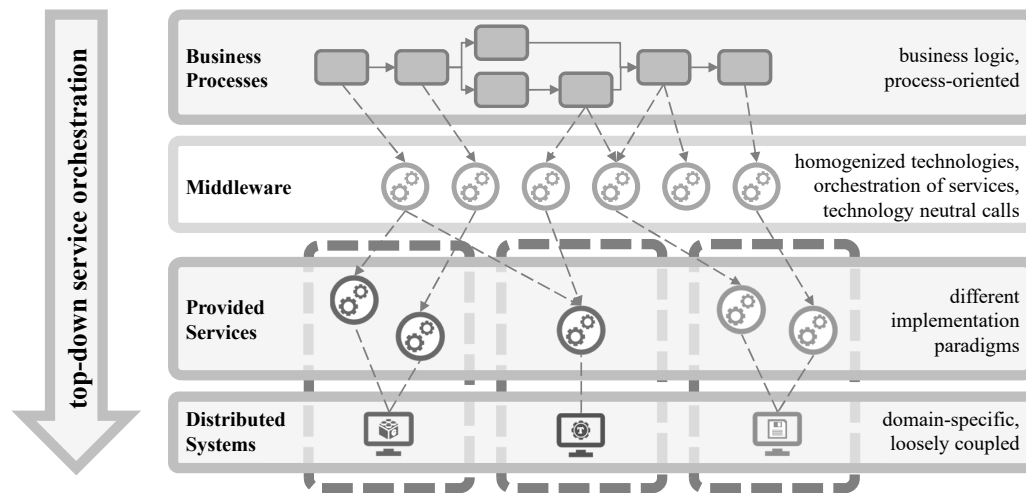


Figure 1. Classic SOA with a process-oriented top-down approach – based on (Kehl, 2019, p. 42)

and modeled completely, nor can there be a central middleware that is aware of all systems and their corresponding services. Each service has to be systemically accessible. This leads to the following challenge:

Challenge 5: Deal with the local anonymity of systems regarding their limited visibility and restricted accessibility.

While collaborating and exchanging data in some degree, brands within a multi-brand OEM are mostly independent from each other, and the brands' IT systems landscapes have emerged historically. In case challenge 6 could be dealt with, it is most likely that the individual systems use different formats of communication and for data exchange. Therefore, we conclude that classic synchronous SOA is not sufficient due to the intransparent quantity and characteristics of the systems used. Thus, we expand challenge 4 as follows:

Challenge 6: Deal with asynchronism of communication and data exchange formats between systems.

2.3 Summary of challenges and observations made in practice

We conclude this subsection by means of three exemplary observations which we have been encountering throughout our research and practical work in multi-brand OEM environments over the past years (Kehl *et al.*, 2015, 2016). In the following we describe observed approaches taken to deal with the challenges we identified in Section 2.1 and 2.2.

The first observation we made is that PLM experts at OEMs often try to define and introduce particularized PLM engineering processes valid for all brands in order to deal with the heterogeneity of processes (challenges 1 and 2). This often results in protracted process analysis and large numbers of iterations. Thus, when introducing a PLM approach in large OEMs, the decentralized character of the PLM processes cannot be mapped efficiently into one single common group-wide process.

A second observation affects the limited usefulness or even applicability of the *frontloading* approach for multi-brand OEMs. Frontloading, e. g. the attempt to speed up the early product development phase by providing more accurate information earlier in the development process, has been advocated in the PLM literature as a silver bullet for optimizing PLM processes (Thomke and Fujimoto, 2000; Belay *et al.*, 2014). Our observation is that PLM experts focus on achieving frontloading by a centralized integration of the data models across the complete left side of the software engineering process model (Groll *et al.*, 2016), from requirements engineering through systems engineering data until to the definition of realizing objects like hard- or software. We conclude that addressing frontloading by introducing a central data backbone for PLM is not a practical solution due to the aforementioned reasons.

Third, we observe that powered by the aforementioned discussions about central processes and data backbone for PLM architectures, often a single-vendor IT strategy is chosen and much resources

are spent on discussing about standard software vs. customizing approaches at OEMs. Thus, PLM introduction activities focus on tools instead of the determination of essential process changes. Figure 2 summarizes these observations (quadrants), sets them in relation to the dimensions “process” (Section 2.1) and “software architecture” (Section 2.2) and situates the challenges addressed in this paper.

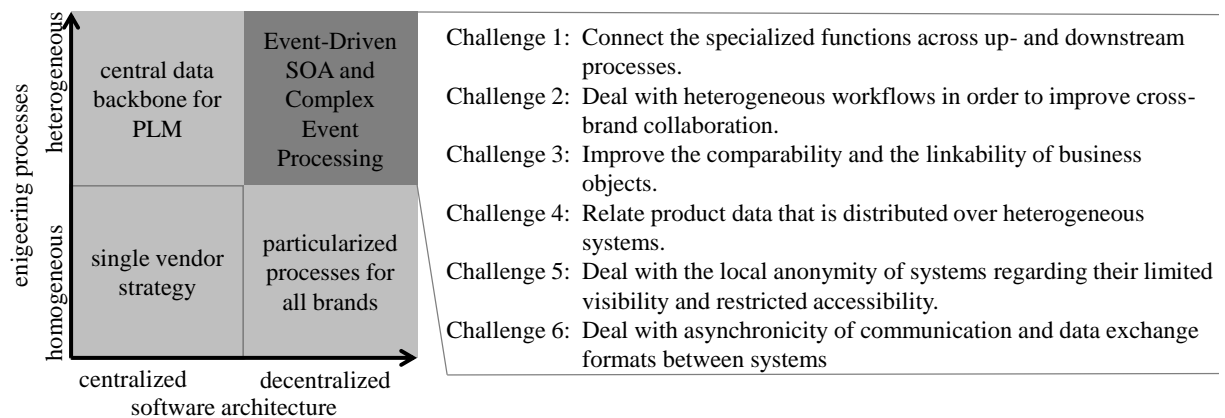


Figure 2. Challenges addressed in this paper in respect to the observations made in practice

We conclude from the discussion in this section that inter-brand and cross-brand collaboration processes at automotive OEMs are mostly insufficiently coordinated. Communication, coordination and co-operation between departments and brands is difficult and can hardly be controlled in a central way. There is a widely spread misconception that central process and system approaches (“one process for all”) will help OEMs to get their PLM integration scenarios managed. This leads us to the following hypothesis:

For an efficient PLM integration, cross-brand OEMs should hold on to the decentralized character of their processes and systems, instead of trying to introduce a central, monolithic process and system approach.¹

In the next section we will discuss publications in the domains of collaboration as well as event processing. Within that discussion we highlight key elements of those publications and their shortcomings.

3 RELATED WORK

There are many publications related to the issues discussed in section 2. A selection of these is covered in the following, starting with related research for collaboration models followed-up with a short overview of recent publications the field of EP.

3.1 Collaboration

Numerous papers have been published dealing with the coordination of actors with different specializations, but in most cases, the focus is limited to different specializations (disciplines) within a single domain (e. g. design). Especially the area of engineering design is well covered (Kehl *et al.*, 2015).

The focus in Concurrent Engineering (CE) is on the parallel execution of steps within the design process in order to achieve the three overall goals: (i) lower costs throughout the entire product lifecycle, (ii) higher quality, and (iii) shorter time to market (Dongre *et al.*, 2017).

In (Chin *et al.*, 2002) the authors propose an approach to improve collaboration of different design-disciplines by deriving multiple discipline-specific views on the same product automatically. Therefore, various discipline-specific submodels are linked using STEP (ISO10303, 2014), whereby several views (so called *domain view models*) can be derived. The coordination of changes in these disciplines and the integration of other domains are not taken into account.

¹ Remind that small and medium enterprises recognize these challenges to a reduced extent but can react and implement process changes more comfortably due to their limited size, while large multi-brand OEMs need to integrate process changes carefully and step-by-step.

Similar applies for the PPO product model (Product-Process-Organization) (Robin and Girard, 2006), which was developed as part of the IPPOP (Integration of Product, Process and Organization for engineering Performance Improvements) research project (Roucoules *et al.*, 2006). PPO is a process-oriented model to support collaboration between designers of various disciplines by resolving conflicts during development (Zheng *et al.*, 2014).

(Stiefel *et al.*, 2012) introduces a decentralized software architecture (based on P2P networks) for collaborative product development across enterprises (or brands). This approach is located in the design domain and focuses on the early and creative phases of the development process. A distributed platform is proposed for forming *ad-hoc collaborations*. Within the scope of such collaborations so-called *initiators* publish specifications as SPARQL-queries (Pan, 2009), on the basis of which *participants* create and submit solutions, which are described by ontologies. In contrast to other approaches, (Stiefel *et al.*, 2012) takes the cross-brand collaboration into account, but focuses on finding and managing loose collaborations, which is not a main contribution of this paper. His work does neither consider data integration, cross-domain collaboration, nor changes to any previously suggested solution.

With special attention to cross-domain collaborations, a SysML-based approach to coordinate activities of different domains during the development of so-called *cybertronic systems* was introduced in (Eigner *et al.*, 2016). The authors propose a modelling framework consisting of several layers that differ regarding the (i) concretion, (ii) level of detail, and (iii) variation. An ontology serves as a common language base to relate the individual layers to each other. Although different domains are considered here, this integration (due to the use of SysML) is limited to the early phases of the development process.

To the best of our knowledge, the only approach that includes the integration of potentially arbitrary domains is presented in (Kehl, 2019), where a cross-domain and component-based product model is proposed that can be extended in a decentralise fashion with domain-specific results. Extensions and subsequent changes of such components are coordinated via an event-based software architecture. In contrast to the approach of this paper, however, the set and type of the events considered is static, which is why they must be centrally defined. This contrasts with the assumptions made in Section 2. In order to meet the requirements derived in section 2, we propose Complex Event Processing (CEP)-based approach in conjunction with such an event-driven architecture (EDA).

3.2 Event processing and complex event processing

CEP is a special form of event stream processing (figure 3a). (CEP Market, 2015) and (Streaming Analytics, 2015) forecast a very significant growth of the CEP market and of the market for streaming analytics in the years up to 2020. This matches (Opieła *et al.*, 2016, p. 19)'s evaluation of CEP technology to have a high potential for the future (figure 3b).

While CEP is an area of active research (see e.g., (Krumeich *et al.*, 2014; Flouris *et al.*, 2016; Soffer *et al.*, 2017; Dayarathna and Perera, 2018)), its standardization concerning definitions, data formats and interfaces is still at an early stage. While current research findings indicate that the event processing industry will continue to have a significant growth over the next decade (Dayarathna and Perera, 2018), it is important to gain industry-related and domain-specific experiences, especially with a larger number of heterogeneous sources of information (Krumeich *et al.*, 2014). Most of the recent studies did not cover issues that arise in largely or any distributed settings, for example in parallel processing or geographically dispersed sites (Flouris *et al.*, 2016). Furthermore standardization of interfaces, event formats and processing languages is an important factor (Krumeich *et al.*, 2014).

Other research presents current use of EP and CEP, in particular. Most of them cover the area of Business Process Management (BPM) or production planning, like (Janiesch *et al.*, 2012; Schlegel *et al.*, 2012; Pielmeier *et al.*, 2017). But there are areas other than BPM where CEP is applied in useful ways. In (Terroso *et al.*, 2015) a CEP approach is used to perceive the context of a vehicle via processing of data from sensors, which are build into the car. The results show high accuracy in detecting one of many itineraries as well as the occupancy of the car. (Shi *et al.*, 2017) present a hierarchical framework with which it is possible to do a real-time mood tracking of microblog streams with CEP. In their work the feasibility as well as effectiveness of the used framework is demonstrated. In contrast to the mentioned works, our study takes multiple event sources into account, including system events as well as human interaction, and does not solely focus on sensor data *or* human input.

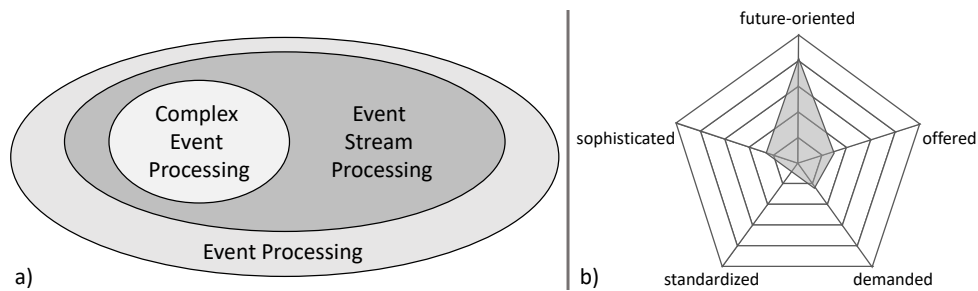


Figure 3. The Event Processing Landscape based on (Dayarathna and Perera, 2018, p. 2) and characteristics of the CEP technology based on (Opiela et al., 2016, p. 19)

4 ARCHITECTURAL APPROACH: EVENT-DRIVEN SOA AND COMPLEX EVENT PROCESSING

Based on the discussion in Section 2 as well as Section 3, we now devise an approach for managing heterogeneous processes and product data on the basis of a loose coupling of heterogeneous systems along the product development phases. In doing so, we aim at maintaining lean processes despite high complexity. In addition, supporting users by automating processes to some extent is necessary to keep management of requirements, changes, and configurations feasible (Hammerschmidt and Hoffmann, 2018). This is especially necessary given the prospect that the amount of product-related data is expected to double every two years for the next decade (Pielmeier et al., 2018). Such loose coupling of systems can be achieved by using an EDA, a software architecture defined by system events which are usually propagated via a publish-subscribe mechanism (figure 4). An event is information about something that happens or is considered to be happening (Luckham, 2002). For our purpose, we follow (zur Muehlen and Shapiro, 2010) who defines events as state changes of objects within the context of a business process.

Pairing process automation with loose coupling of systems based on an EDA does not only demand the capability to handle a large amount of events triggered across a vast system infrastructure, but also a technical solution to combine these events with knowledge about process sequences and handovers across different domains. In addition, this requires dealing with large numbers of application systems, which contain highly diverse data as well as data structures, and which are used for various tasks by different users / roles in varying phases of the product development process.

As a technological vehicle to implement our approach, we use CEP, a software technology first defined by (Luckham, 2002). EDA is an essential concept for CEP (Krumeich et al., 2014) and has proven well-suited for asynchronous work and information flows (Michelson, 2006); dealing with the process asynchronism is a key advantage of using CEP along the product development. Furthermore, CEP systems are capable of handling large amounts of data, enabling processing such data, i. e. events, in near real-time and recognize relevant situations as they occur (Flouris et al., 2016).

Therefore, we second CEP as an approach for dealing with the problems and complexity discussed in Section 2; with this approach we aim at closing the usual gap between process definitions and the data objects which commonly represent intermediate or end results of processes. Note that today, this gap often leads to difficulties in aggregating the current progress of any process or project.

The process-relevant product data is commonly created in an asynchronous fashion as discussed in Section 2.1. Therefore, the data is made available without having direct or immediate interaction with process partners. As a possible result, the further course of a process or sub-process suffers from delay due to lack of information exchange regarding newly created data. However, this is not solely a technical issue of not implementing such an information flow. The fact that not every relevant process partner is known to the data owner or distributor at the time of data creation is an important factor leading to this problem. The publish-subscribe mechanism within a CEP approach can help solving this issue, since relevant process partners can themselves subscribe to the information they are interested in and therefore making the need for the data owner to know every process partner obsolete.

The goal of this approach is to support complex development processes by using CEP, since it is capable of deriving higher level knowledge from aggregating lower-level system events occurring within an EDA (Luckham, 2002; Eckert, 2008). With this we aim at improving three key factors: *process evaluation*, *process continuity*, and *process control*, which are described in more detail hereinafter.

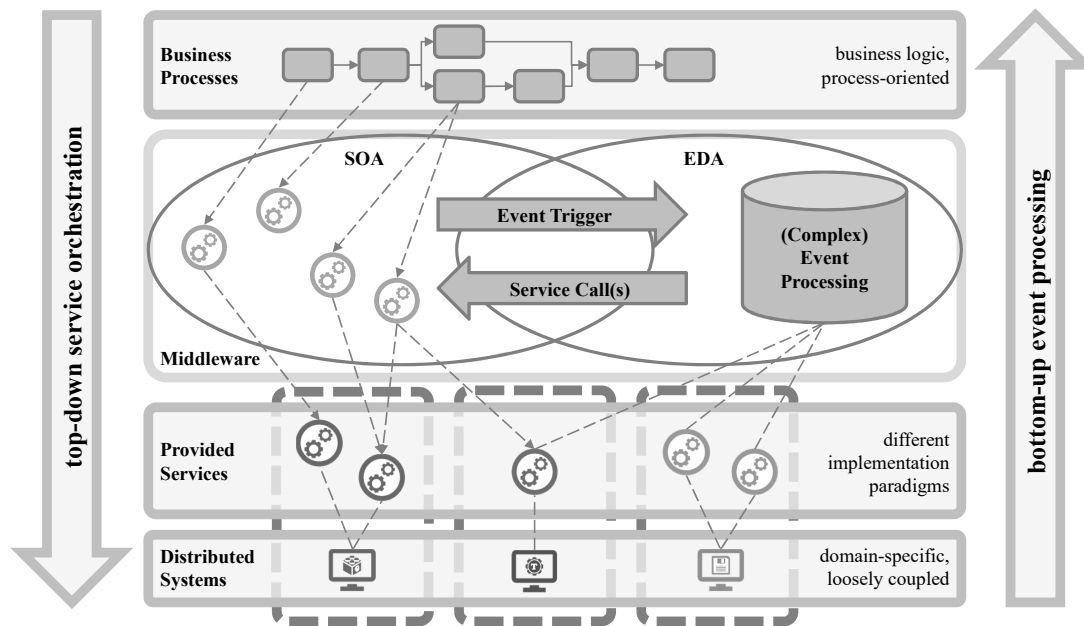


Figure 4. Event-driven SOA to combine a process-oriented top-down and an event-driven bottom-up approach – based on (Kehl, 2019, p. 45)

The *process evaluation* analyses the current status of any process covered as event patterns in a CEP engine, which is the core component of a CEP approach responsible for processing incoming events on the basis of patterns. This is done by evaluating the current status within an event pattern and its complex events. However, doing so requires domain specific knowledge from the involved areas along the development process in order to find a match between the simple events on the lowest level of any pattern and the specifications of product data or changes on those data. Since product data is scattered across a variety of systems it is necessary to connect data from multiple systems via a pattern, in case the overlying process requires this. This is defined using the term *process continuity*. Lastly, the completion of a event pattern will trigger predefined actions. Those can be new events to start the next pattern via notification for the responsible user or an automated specific process step, described as *process control*. Our goal is to improve these three factors with our approach based on CEP technologies and methods. We want to improve the collaboration of users and systems from different domains by developing a complex event processing framework potentially existing of multiple engines exchanging event data collected via an EDA. That includes cross-brand collaboration which is an uncommon aspect in similar research.

5 CONCLUSION AND FUTURE WORK

The decentralized organizational structure of large OEMs, which is unavoidable due to their size and global distribution, poses a number of challenges (Section 2) that many try to tackle by implementing collaboration models *or* event-based approaches (Section 3). In addition, backed by our experience we describe observations from industrial practice of attempts to cope the challenges (Section 2.3). These observations have shown that current approaches in industry try to answer those challenges in a one-dimensional manner (see Figure 2), by supporting heterogeneous engineering processes *or* the decentralization of software systems. Based on this, we advocate a CEP-based approach to address the challenges by supporting both, and present first steps towards such an approach (Section 4).

As next steps on the way to such an approach we see the following research tasks: In order to identify the different abstraction levels of a process required to define event patterns, it is necessary to collect and incorporate expert domain knowledge (Soffer *et al.*, 2017). This knowledge about different steps and iterations to take, about intermediate or end results and process documentation is important in order to derive concepts for a CEP approach based on Use-Cases. This gives the opportunity to identify process steps or iterations which are crucial for the collaboration along development since they i. e. cause a multitude of follow-up processes.

At the same time it is important to find suitable technical solutions to support such processes. For example, the levels of abstraction can be described and visualized with an event model which requires knowledge about CEP and its mechanisms. Furthermore it is important from an overall perspective to evaluate empirically and theoretically different ways of applying a CEP approach in large-scale scenarios. This evaluation requires the formulation of suitable criteria and metrics, and will need to take many factors into account, i. e. creators of product data, format and structure of such data as well as process owners.

REFERENCES

- Belay, A.M., Welo, T. and Helo, P. (2014), "Approaching lean product development using system dynamics: investigating front-load effects", *Advances in Manufacturing*, Vol. 2 No. 2, pp. 130–140.
- Bruns, R. and Dunkel, J. (2010), *Event-Driven Architecture: Softwarearchitektur für ereignisgesteuerte Geschäftsprozesse*, Xpert.press, Springer, Berlin.
- Complex Event Processing (CEP) Market - Global forecast to 2019, Research and Market.
- Chin, K.-S., Zhao, Y. and Mok, C.K. (2002), "STEP-Based Multiview Integrated Product Modelling for Concurrent Engineering", *The International Journal of Advanced Manufacturing Technology*, Vol. 20 No. 12, pp. 896–906.
- Dayarathna, M. and Perera, S., (2018), "Recent Advancements in Event Processing", *ACM Comput. Surv.*, Vol. 51 No. 2, pp. 33:1–33:36.
- Deuter, A. and Rizzo, S. (2016), "A Critical View on PLM/ALM Convergence in Practice and Research", *Procedia Technology*, Vol. 26, pp. 405–412.
- Dongre, A.U., Jha, B.K., Aachat, P.S. and Patil, V.R. (2017), "Concurrent Engineering. A Review", *International Research Journal of Engineering and Technology (IRJET)*, Vol. 4 No. 5, pp. 2766–2770.
- Durão, L.F.C.S., Eichhorn, H., Anderl, R., Schützer, K. and Senzi Zancul, E. de (2015), "Integrated Component Data Model based on UML for Smart Components Lifecycle Management. A Conceptual Approach", in Bouras, A., Eynard, B., Foufou, S. and Thoben, K.-D. (Eds.), *12th IFIP International Conference on Product Lifecycle Management*, Springer International Publishing, Cham, pp. 13–22.
- Eckert, M. (2008), *Complex Event Processing with XChangeEQ: Language Design, Formal Semantics, and Incremental Evaluation for Querying Events*, Dissertation, LMU München: Fakultät für Mathematik, Informatik, Ludwig-Maximilians-Universität München, München.
- Eigner, M., Dickopf, T. and Huwig, C. (2016), "An Interdisciplinary Model-Based Design Approach for Developing Cybertronic Systems". In: *DESIGN 2016 - 14th International Design Conference*. Hrsg. von Marjanovic Dorian, Storga Mario, Pavkovic Neven, Bojetic Nenad und Skec Stanko. DESIGN. 2016, S. pp. 1647–1656.
- Flouris, I., Giatrakos, N., Deligiannakis, A., Garofalakis, M., Kamp, M. and Mock, M., "Issues in complex event processing: Status and prospects in the Big Data era", *Journal of Systems and Software*, Vol. 127, pp. 217–236.
- Groll, M.W. and Heber, D.T. (2016), "E/E-Product Data Management in Consideration of Model-Based Systems Engineering?", in Stjepandić, J., Borsato, M., Peruzzini, M., Wognum, N. and Verhagen, W.J.C. (Eds.), *Transdisciplinary engineering: Crossing boundaries proceedings of the 23rd ISPE Inc. International Conference on Transdisciplinary Engineering*, October 3-7, 2016, Advances in transdisciplinary engineering, IOS Press, Amsterdam, Netherlands.
- Hammerschmidt, C. and Hoffmann, D. (2018), *Product Lifecycle Management: Das PLM von morgen*, Media-Manufaktur GmbH, Pattensen.
- Hesselmann, C., Kehl, S., Stiefel, P. and Müller, J.P. (2017), "Decentralized handling of conflicts in multi-brand engineering change management", in Maier, A., Skec, S., Kim, H., Kokkolaras, M., Oehmen, J., Fadel, G., Salustri, F. and van der Loos, M. (Eds.), *ICED17: 20th-25th August 2017*, University of British Columbia, Vancouver, Canada, Design organisation and management, *Proceedings of the 21th International Conference on Engineering Design*, Design Society, pp. 683–692.
- International Organization for Standardization (2014), *ISO10303-242-2014: Industrial automation systems and integration – Product data representation and exchange – Part 242: Application protocol: Managed model-based 3D engineering No. 10303, 242nd ed.*, Geneva, CH, available at: <https://www.iso.org/standard/57620.html> (accessed 14 September 2017).
- Janiesch, C., Matzner, M. and Müller, O. (2012), "Beyond process monitoring. A proof-of-concept of event-driven business activity management", *Business Process Management Journal*, Vol. 18 No. 4, pp. 625–643.
- Kees, M. and Seibertz, A. (2010), "Compositional Variant Management and its Application in Embedded Software Development", available at:

http://www.prostep.org/fileadmin/user_upload/ProSTEPiViP/Events/Symposium-2010/Presentation_Tag_2/1002_Presentation_Symposium_Seibertz-ProSTEP-IMP_Kees-Ford.pdf (accessed 23 February 2016).

- Kehl, S., Stiefel, P. and Mueller, J.P. (2015), "Changes on changes: towards an agent-based approach for managing complexity in decentralized product development", in Weber, C., Husung, S., Cascini, G., Cantamessa, M. and Marjanovic, D. (Eds.), ICED15: 27th-30th July 2015, Politecnico di Milano, Italy proceedings of ICED15, Design organisation and management, *Proceedings of the 20th International Conference on Engineering Design*, Design Society, pp. 219–228.
- Kehl, S., Hesselmann, C., Stiefel, P. and Müller, J.P. (2016), "Static Product Structures. An Industrial Standard on the Wane", in Harik, R., Rivest, L., Bernard, A., Eynard, B. and Bouras, A. (Eds.), *Product Lifecycle Management for Digital Transformation of Industries: 13th IFIP WG 5.1 International Conference, PLM 2016*, Columbia, SC, USA, July 11-13, 2016, Revised Selected Papers, Springer International Publishing, Cham, pp. 69–78.
- Kehl, S. (2019), *Marken- und domänenübergreifendes Management industrieller Produktdaten*, 1. Auflage 2019, Springer Fachmedien Wiesbaden GmbH; Springer Gabler, Wiesbaden.
- Krumeich, J., Weis, B., Werth, D. and Loos, P., (2014), "Event-Driven Business Process Management - Where are we now?", *Business Process Management Journal*, Vol. 20, No. 4, pp. 615–633
- Luckham, David C. (2002), "The power of events. An introduction to complex event processing in distributed enterprise systems", Reading, Addison-Wesley, Boston Mass.
- Michelson, Brenda M. (2006), "Event-Driven Architecture Overview", Patricia Seybold Group Research Service
- Opiela, N., Hoepner, P. and Weber, M. (2016), *Das ÖFIT-Trendsonar der IT-Sicherheit*, Kompetenzzentrum öffentliche Informationstechnologie, Fraunhofer-Institut für Offene Kommunikationssysteme FOKUS, Berlin.
- Pan, J.Z. (2009), "Resource Description Framework", in Staab, S. and Studer, R. (Eds.), *Handbook on Ontologies*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 71–90.
- Pielmeier, J., Braunreuther, S. and Reinhart, G. (2017), "Modeling Approach for Situational Event-handling within Production Planning and Control Based on Complex Event Processing", *Procedia CIRP*, Vol. 63, pp. 271–276.
- Pielmeier, J., Braunreuther, S. and Reinhart, G. (2018), "Approach for Defining Rules in the Context of Complex Event Processing", *Procedia CIRP*, Vol. 67, pp. 8–12.
- Robin, V. and Girard, P. (2006), "An integrated product-process-organisation model to manage design system", in *The Proceedings of the Multiconference on "Computational Engineering in Systems Applications"*, Beijing, China, 04.10.2006 - 06.10.2006, IEEE, pp. 1287–1293.
- Roucoulès, L., Noel, F., Teissandier, D., Lombard, M., Debarbouille, G., Girard, P., Merlo, C. and Eynard, B. (2006), "IPPOP: an open source collaborative design platform to link product, design process and industrial organisation information". In: *6th International Conference on Integrated Design and Manufacturing in Mechanical Engineering*, IDMME 06. 2006, CDROM.
- Schlegel, T., Vidackovic, K., Dusch, S. and Seiger, R. (2012), "Management of interactive business processes in decentralized service infrastructures through event processing", *Journal of King Saud University - Computer and Information Sciences*, Vol. 24 No. 2, pp. 137–144.
- Shi, S., Jin, D. and Tiong-Thye, G. (2017), "Real-time Public Mood Tracking of Chinese Microblog Streams with Complex Event Processing", *IEEE Access*, Vol. 5, pp. 421–431.
- Soffer, P., Hinze, A., Koschmider, A., Ziekow, H., Di Ciccio, C., Koldehofe, B., Kopp, O., Jacobsen, A., Sürmeli, J. and Song, W. (2017), "From event streams to process models and back: Challenges and opportunities", *Information Systems*.
- Stiefel, P.D., Hausknecht, C. and Müller, J.P. (2012), "Using ontologies to support decentral product development processes", in Fischer, K., Müller, J.P. and Levy, R. (Eds.), *Agent-Based Technologies and Applications for Enterprise Interoperability*, Springer-Verlag, pp. 114–129.
- Streaming Analytics Market - Worldwide Market Forecast and Analysis (2015 - 2020), Research and Markets.
- Terroso-Sáenz, F., Valdés-Vela, M., Campuzano, F., Botia, J.A. and Skarmeta-Gómez, A.F. (2015), "A complex event processing approach to perceive the vehicular context", *Information Fusion*, Vol. 21, pp. 187–209.
- Terzi, S., Bouras, A., Dutta, D., Garetti, M. and Kiritsis, D. (2010), "Product lifecycle management – from its history to its new role", *International Journal of Product Lifecycle Management*, Vol. 4 No. 4, p. 360.
- Thomke, S. and Fujimoto, T. (2000), "The Effect of "Front-Loading" Problem-Solving on Product Development Performance", *Journal of Product Innovation Management*, Vol. 17 No. 2, pp. 128–142.
- Zheng, C., Bricogne, M., Le Duigou, J. and Eynard, B. (2014), "Mechatronic Design Process. A Survey of Product Data Model", *Procedia CIRP*, Vol. 21, pp. 282–287.
- zur Muehlen, M. and Shapiro, R. (2010), "Business Process Analytics", *Handbook on Business Process Management 2: Strategic Alignment, Governance, People and Culture*, Springer, Berlin, pp. 137–157.