



HAL
open science

A Methodological Approach for the Identification of Context-Specific Reconfiguration Options in the PLM-Context

Philipp Savarino, Michael Abramovici, Jens Christian Göbel

► **To cite this version:**

Philipp Savarino, Michael Abramovici, Jens Christian Göbel. A Methodological Approach for the Identification of Context-Specific Reconfiguration Options in the PLM-Context. 15th IFIP International Conference on Product Lifecycle Management (PLM), Jul 2018, Turin, Italy. pp.389-399, 10.1007/978-3-030-01614-2_36 . hal-02075563

HAL Id: hal-02075563

<https://inria.hal.science/hal-02075563>

Submitted on 21 Mar 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

A methodological approach for the identification of context-specific reconfiguration options in the PLM-context

Philipp Savarino*, Michael Abramovici, Jens Christian Göbel

Chair of Information Technology in Mechanical Engineering (ITM),
Ruhr-University Bochum, Germany
philipp.savarino@itm.rub.de

Abstract. Current changes of traditional products towards intelligent, connected smart products require a fundamental adaption and enhancement of traditional Product Lifecycle Management approaches. In our previous activities we showed that lifecycle management approaches lack suitable methods and IT tools for smart products especially during their use phase. In particular the reconfiguration in the use phase has been addressed, as the characteristic properties of smart products enable a tremendous potential. As an approach virtual product twins were introduced in our previous activities in order to manage both virtual product models as well as product usage data across all engineering domains including environmental data. This contribution describes the core enabler, such as the reconfiguration lifecycle and virtual product twins for the reconfiguration of smart products during their use phase including the necessity of designing the architecture of reconfiguration options. The scope is the introduction of a methodological approach, which allows the identification of context-specific reconfiguration options.

Keywords: Reconfiguration, Smart Products, Product Lifecycle Management, Virtual Twin

1 Introduction

Innovations in Information and Communication Technology (ICT) have led to drastic changes of traditional products towards intelligent smart products. Smart products are defined as intelligent products with the ability to communicate and interact with other smart products and their environment by using internet-based services [1]. Among others, characteristic properties of smart products are their high degree of personalization and automation, autonomic behavior in decision making and their vast number of multidisciplinary components. In the scope of this contribution their capability to react in real-time and their tremendous potential for dynamic reconfiguration especially during their use case is of particular interest. Here, suitable methodological approaches and IT-tools, e.g. for the lifecycle-spanning management of both virtual product instance

models as well as product usage data, are scarcely covered by common Product Lifecycle Management (PLM) approaches [2].

In general it can be distinguished between two types of reconfiguration: type-specific reconfigurations such as common software updates and context-specific reconfigurations like smart services generated individually for each product instance. The paper at hand addresses the context-specific reconfiguration, where every product instance, based on a permanent, individual management across all engineering domains, is confronted with the question, which reconfiguration option can be individually offered. Approaching this challenge, the normative standard ISO 10007 introduced a configuration management model including the definition of configuration units (CU). These configuration units can be considered as entities within an instance configuration that provide a specific end use function. The norm introduced a change management approach, which shares characteristics with a reconfiguration process that are not limited to any lifecycle phase [3]. However, it rather addresses type-specific than context-specific reconfigurations, as it requires otherwise knowledge of all product instance configuration units along the entire lifecycle.

This idea was approached by the idea of virtual product twins, which was originally introduced by the NASA's technology roadmap "Modeling, Simulation, Information Technology & Processing". The virtual twin was defined as an integrated multiphysics, multiscale simulation of a vehicle or system that uses the best available physical models and sensor updates to mirror the life of its corresponding twin [4]. This approach however focused mainly the simulation of the physical twin in order to predict its behavior, instead of using the product instance knowledge for compatibility checks in order to solve reconfiguration option issues, for example.

The IEEE 828 standard for configuration management in systems and software engineering introduced a description of the configuration management process, including the configuration identification process, which is a necessary element for the offering of compatible reconfiguration options. The identification is focusing primarily physical and functional characteristics and a semantic-based description though and excludes to tackle the question on how configuration units can be systemized to allow the description of a product instance's context [5].

An approach for a model-based, consistent management of control unit configurations by using function blocks is defined by the IEC 61499 standard. It focuses rather on the architecture of distributed agile manufacturing systems and thus does not consider the whole lifecycle of a smart product [6]. This standard has been enhanced in several research activities, however by mainly considering the automation of production by solving reconfiguration issues of electronic and software components [7,8].

None of these research activities identified the potential of smart product reconfiguration in the use phase while considering the virtual twin concept as a basis for the reconfiguration process. Additionally, the necessary design of the reconfiguration options architectures has not been in the scope of research activities so far, which is why these issues will be addressed in the following sections.

2 Enabling the reconfiguration lifecycle of smart products

This chapter describes the core enabler for the reconfiguration of smart products during their use phase. It consists of the reconfiguration lifecycle and virtual product twins, which are both integral components of smart products. Apart from that, the methodological approach for the design of reconfiguration options architectures is introduced in order to be able to offer context-specific reconfiguration options.

2.1 The reconfiguration lifecycle of smart products

The characteristics of smart products, as mentioned in chapter 1, lead to a highly expanding set of capabilities regarding the reconfiguration compared to traditional products. Here, the reconfiguration of smart products during the use phase is of particular interest. It enables for example a vast amount of personalization potential, due to the continuous awareness of its own configuration and environment. This continuous reconfigurability during the use phase can be regarded as a reconfiguration lifecycle as shown in figure 1.

The process of configuration starts at the beginning of a smart product's lifecycle. The development of an initial generic variant product configuration in a company's product portfolio establishes the possibility to choose a specific instance configuration, e.g. based on a customer's order (as designed / as ordered). The configuration process is characterized by rules that define the relationship between elements, which can be described as configuration units [3]. Due to variation in the production / manufacturing process, for example related to deviation of production resources, the smart product is built (as built). Parallel to the smart product's lifecycle, the generic variant product configurations available for possible reconfiguration differ, as some variants may have left the product portfolio e.g. for external reasons, such as changes in legislation. This diverse set of reconfigurations is now available in the smart product's use phase. They can be offered during maintenance processes (as maintained) or used to offer a new set of IT-driven functions to the product, e.g. by offering higher degree of personalization for the customer (as reconfigured). Subsequently, the new configuration has to be implemented (as implemented) and reopens a new use / operation phase of the smart product until it reaches its end of life.

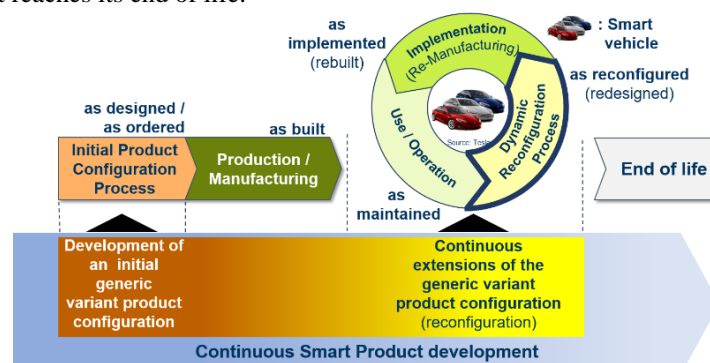


Fig. 1. The reconfiguration lifecycle of a smart product [9].

2.2 Virtual Product Twins

As in the previous chapter introduced, the reconfiguration of smart products during the use phase enables possibilities like a higher degree of personalization. Therefore, a higher attention to the management of both virtual product models from previous lifecycle stages as well as the present models need to be at hand in order to determine the given configuration. Apart from that, especially environmental information can enable reconfiguration offers that can be described as context-specific. A context in this particular angle can be understood as the circumstances that form the setting for an event [10]. For example, a context-specific reconfiguration offer can be described as a triggered event, which is happening based on the smart product's current situation. The possibility to describe a smart product's current state or the surrounding environment is given by product use data. Thus, a holistic approach to describe both virtual product models as well as product use data along the entire lifecycle is shown in figure 2 by considering virtual product twins. In the past, various approaches with different terms addressed similar research questions, for example product avatars where the product-instance is the centre of a lifecycle-spanning management concept [11]. In general, the virtual product twin can be considered as the notion, where data from each stage of the product lifecycle is transformed into information which is made seamlessly available to subsequent stages [12]. An example for the transformation of the data into information can be the use of the data to predict the product's behavior after setting it in its semantic context.

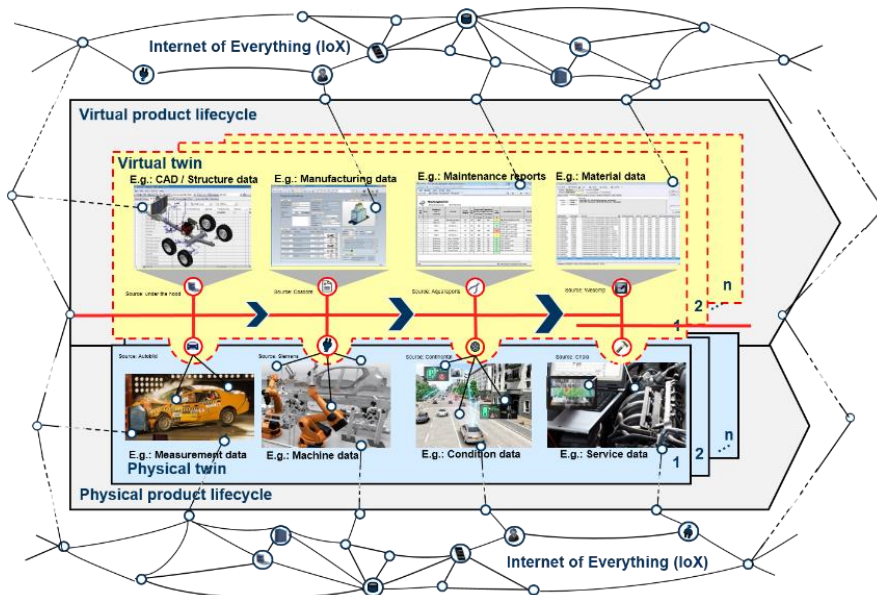


Fig 2. Virtual product twins as a holistic approach for the consideration of virtual product models and product use data along the entire lifecycle [13].

The given approach differentiates between a virtual and a physical lifecycle, because every virtual product twin is related to a physical instance mirroring a physical twin. In

the early phases virtual product models are created in the virtual lifecycle, such as CAD data or structure data based on the configuration process for example. In the following phases, for example manufacturing data, maintenance reports or material data provide information about the current state of the product instance. In the early phases of the physical lifecycle e.g. measurement data can be collected from early prototypes. Subsequently, machine, condition or service data provide knowledge about either the current product instance's own state as well as its environment. As previously described, the data gathered and transformed into information and is then made seamlessly available to subsequent phases. Thus, the amount of information along the entire lifecycle is increasing from every phase as well as from every reconfiguration. This leads to a higher set of context-specific deployment possibilities of the virtual twin especially during the use phase [13].

2.3 Design methodology for reconfiguration options in the scope of context-specific smart product reconfiguration

In order to offer reconfiguration options that can be considered as context-specifically adequate, the architecture of reconfiguration options need to be addressed. Especially for the reason that some reconfiguration options can have a high amount of overlapping configuration units with other reconfiguration options, a systematic approach will be introduced that indicates all necessary components for smart product reconfiguration options. In general, there are two types of configuration units (CU) that can be differentiated: physical configuration units and value-based configuration units. Physical CUs are referring to models describing physical parts such as CAD-data or manufacturing data. The relationship between these physical configuration units is described by considering metadata, which defines the relationship in form of hierarchical levels. Thus, for instance, an assembly between two parts can be described by using metadata that define the structural interdependence between them. Value-based CUs are addressing the context-specific attributes of smart product reconfiguration options. They define a certain value or value-interval of a sensor, for example. In order to successfully offer a reconfiguration option, these values has to be satisfied by the corresponding values of the product's instance sensor. Both physical and value-based CUs can be embedded in a three-layered architecture (figure 3):

- Class list of characteristics group
- Class list of characteristics
- Characteristics

The class list of characteristics group is defining the nature of a reconfiguration option. This classification is nevertheless challenging, as the tremendous amount of interdisciplinary parts in products increase. Therefore, a continuum between mechanical-based and smart reconfiguration options allow a rough classification of a reconfiguration options nature. While mechanical-based reconfiguration option address solely mechanical (physical) configuration units, smart reconfiguration options include highly complex and interdisciplinary configuration units, including software. Additionally, smart reconfiguration options consider value-based configuration options as well.

Class list of characteristics group	Mechanic-based reconfiguration option	↔		Smart reconfiguration option
Class list of characteristics	Reconfiguration option 1	...	Reconfiguration option n	...
Characteristics	Physical CU 1 (PLM) Physical CU 2 (PLM) Physical CU 3 (PLM) ...	Meta-data	Value-based CU 1 (IoT) Value-based CU 2 (IoT) Physical CU 1 (PLM) Physical CU 2 (PLM) ...	Meta-data

Fig. 3. Components of smart product reconfiguration options.

The class list of characteristics describes the reconfiguration option itself. Those are more specifically described by their characteristics located in the third layer, which are simultaneously responsible for the classification of the reconfiguration options nature.

3 Approach for the identification of context-specific reconfiguration options in the PLM-context

In this chapter the architecture and the components for the identification of context-specific reconfiguration options for smart products are outlined. While the first part describes a three-layered structure, the second part identifies the necessary components and their interdependencies.

3.1 Architecture of the methodological approach

A structural systematization of the required components for the identification of context-specific reconfiguration options in the PLM-context shows is shown in figure 4. It consists of three layers each bearing a different character of information considering the given input and output in the form of value-based and physical configuration units (cf. chapter 2.3). Additionally, each layer is differentiating between a physical and a virtual lifecycle, where the physical lifecycle can be regarded as the source of the value-based configuration units (e.g. the value of sensors) and the virtual lifecycle as the source of the physical configuration units (e.g. CAD-data).

The generic layer contains rules that define the relationship between them. This refers for example to configuration units that hold an incompatibility in certain assemblies. The total amount of configuration units as well as its values in the generic layer are discrete as the reconfiguration options are fully defined by the reconfiguration provider. Concrete, all reconfiguration options with its physical and value-based configuration units are determined and are the basis for further consideration for context-specific reconfiguration offers for each product instance.

Part of the instance specific layer are the data of the Product Lifecycle Management (PLM) as well as the Internet of Things (IoT). It contains all product relevant data such as structure, behavior or product usage that can be part of a product instance data for example. Therefore, as it combines all product relevant data from the virtual and the physical lifecycle, it can be considered as the virtual product twin layer. In order to

identify compatible context-specific reconfiguration options, generic reconfiguration options have to be matched with the instance-specific data. Considering the compatibility mainly physical and value-based configuration units including structural information between them have to be checked. Therefore, only a small part of the whole instance-specific layer is relevant for the successful identification of context-specifically matching reconfiguration options.

On the context-specific layer product instance data is constantly updated in order to check the compatibility of the generic reconfiguration options with the product instance models and data. Therefore, the concrete product structure data with its physical configuration units as well as the product usage data in form of value-based configuration units. However, these physical and value-based configuration units are similar to those predetermined by the generic reconfiguration units and congruent to those from the instance-specific layer. Consequently, after a successful compatibility check, a reconfiguration option that is matching to the product instance's context can be identified and offered.

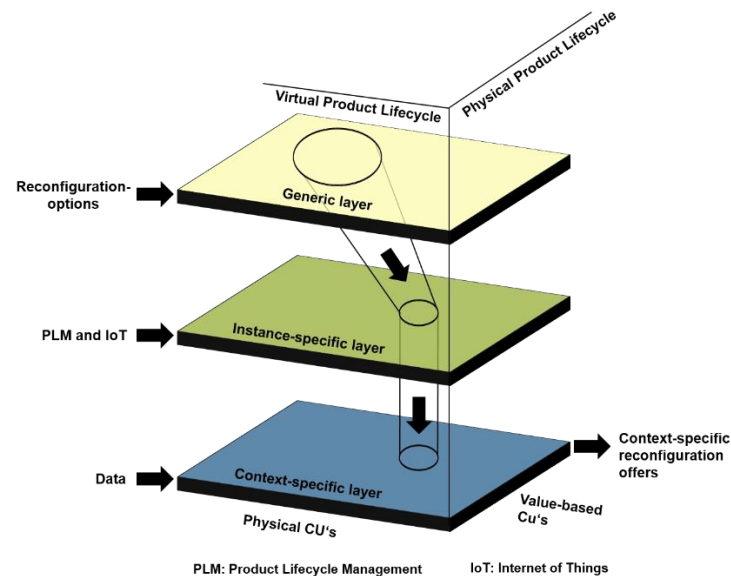


Fig. 4. Structural systematization of the context-specific reconfiguration components in the PLM-context.

3.2 Components of the methodological approach

Necessary components for the offering of reconfiguration options by considering a two-staged methodological approach are shown in figure 5. The approach bases on the generic layer with all possible reconfiguration options that can be offered by a provider. The reconfiguration options derive from the architectural framework as presented in chapter 2.3. Here, the characteristics of each reconfiguration options, namely the physical and value-based configuration units, are of particular interest for the methodological approach. The physical configuration units provide the input for the first stage for

the identification of context-specific reconfiguration options. Therefore, the physical configuration units can be considered as the first objective that has to be successfully met by the physical configuration units of the product instance as part of the product reconfigurator. The aim is to find a general compatibility of the product instance's physical configuration units and its assembly. For example, if there is a necessity for specific configuration units in order to offer a reconfiguration option like an autonomous parking service for a smart vehicle, such as ultrasonic sensors or wheels, those are predefined by the generic reconfiguration options and checked for presence in the product instance. Thus, also the physical configuration units of the product instance have to be considered for this compatibility checks. These physical configuration units are part of the product instance's virtual twin, concrete the part concerning the virtual product lifecycle such as product models related to physical units like CAD data from an engine part. For the first compatibility check are mainly virtual product models relevant, which describe the configuration unit (e.g. CAD-data) including metadata like structural information. Of particular interest are only those configuration units, which are part of the reconfiguration option architectures and contain the physical configuration units of a product instance within its assembly unit. These data can be retrieved from PLM-systems and analyzed in a product reconfigurator tool, e.g. by exporting the configuration unit describing data including its metadata on a predefined temporary basis. Such a product reconfigurator tool is capable of matching determined sets of rules between the physical configuration units in the generic reconfiguration units and those present by the structure models of a product instance.

If the general compatibility between generic reconfiguration option's physical configuration units and those of the product instance have been proven successful, these reconfiguration options are taken into further consideration for context-specific offerings. However, as soon as physical configuration units are replaced at either the formal description of the reconfiguration offers or the product instance, for example due to a product portfolio change (generic) or a maintenance processes (instance), this first stage of compatibility checks have to be repeated. The second stage of the identification of context-specific reconfiguration options is particularly describing the current product instance's condition and its environment. In general, a product's condition and its environment can be determined by analyzing sensor data, for example. Thus, the value-based configuration units of the generic reconfiguration options are the basis for this stage. Concrete, every value-based configuration unit, such as the necessary value interval of a supersonic sensor for a reconfiguration option like an autonomous parking service, is defined in the generic reconfiguration options and is considered as a reference during the compatibility checks with the value-based configuration units (condition and environmental data) of the product instance. The actual condition and environmental data of a product instance can be provided by managing a smart products physical lifecycle data with the help of an internet of things (IoT) platform. However, these data need to be provided for the compatibility check on a constant basis. The shorter the query cycles can be implemented, the better the context of the product on basis of the value-based configuration units can be determined. Consequently, the values of the value-based configuration units have to be matched with those described in the generic reconfiguration options. If the values of value-based configuration units match with

those of the reconfiguration options, context-specific reconfiguration offers can be realized.

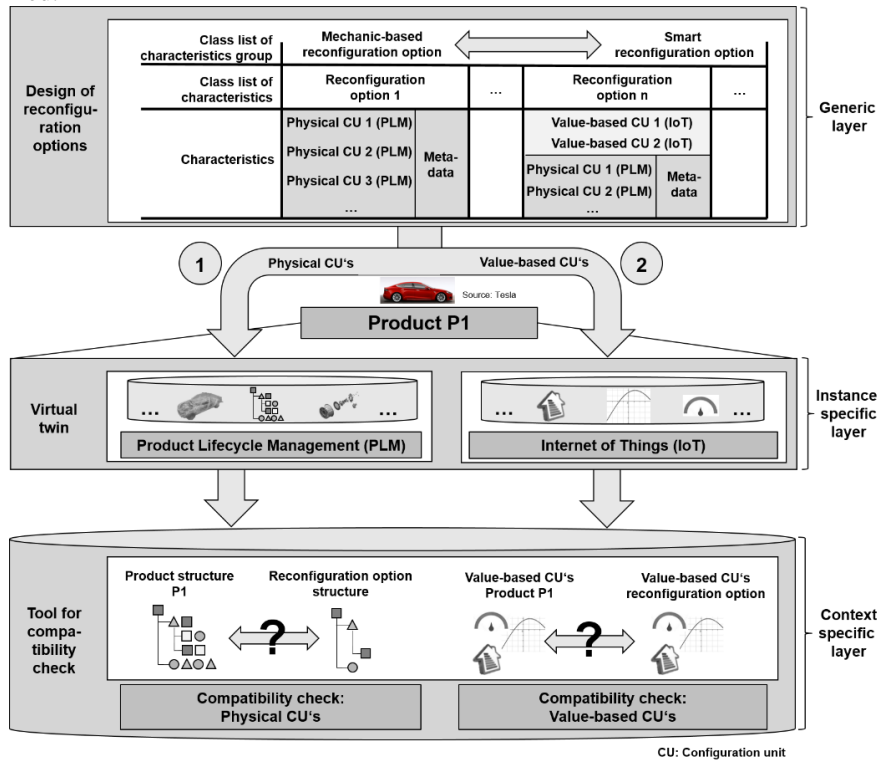


Fig. 5. Approach to identify context-specific reconfiguration options for smart products.

4 Summary and future challenges

Smart products enable a huge potential for reconfiguration scenarios during their use phase. One main driver are their multidisciplinary properties, which reflect their vast amount of multidisciplinary components. This contribution pointed out the main enabler for the reconfiguration of smart products during their use phase, concrete the reconfiguration lifecycle, virtual product twins and the design of reconfiguration option architectures. Here, the reconfiguration lifecycle stressed the potential for reconfiguration of smart products during the use phase by describing the dependencies between the initial configuration process at the product creation process and the continuous reconfiguration process during the use phase. It was shown, how the virtual product twin can serve as a basis for the necessary management of both virtual product models and product use data across all engineering domains along the entire product lifecycle, to support the reconfiguration of smart products during their use phase. Additionally, the architecture of reconfiguration options was introduced, by stressing the characteristics of the configuration units applicable for context-specific reconfiguration offers. Deriving from this point, a three-layered approach was presented with all necessary inputs and

outputs. Finally, this architecture was further detailed by showing how reconfiguration options can be generated context-specifically by considering elements from Product Lifecycle Management as well as Internet of Things. Future challenges will remain in the automation process of identifying context-specific reconfiguration options. This affects especially the compatibility checks, where current physical configuration units and value-based configuration units both from the generic reconfiguration options as well as from the product instance have to be combined.

References

- [1] M. Abramovici, Smart Products, in Lapierrière L, Reinhart G, (Eds) CIRP Encyclopedia of Production Engineering, Springer, Berlin Heidelberg, 2014.
- [2] E. Lutters, F.J.A.M. van Houten, A. Bernard, E. Mermoz, C.S.L. Schutte, Tools and techniques for product design, CIRP Annals - Manufacturing Technology 63 (2) (2014) 607–630.
- [3] DIN ISO 10007, Qualitätsmanagement. Leitfaden für Konfigurationsmanagement (2003).
- [4] NASA Technology Roadmap, Modeling, Simulation, Information Technology & Processing Roadmap Technology Area 11, 2010.
- [5] IEEE standard for configuration management in systems and software engineering, Institute of Electrical and Electronics Engineers, New York, 2012.
- [6] DIN EN 61499, International Electrotechnical Commission 61499-1: Function Block - Part 1 Architecture, 2013.
- [7] A. Zoitl, C. Sunder, I. Terzic, Dynamic reconfiguration of distributed control applications with reconfiguration services based on IEC 61499, in: IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and its Applications: DIS 2006 ; 15 - 16 June 2006, Prague, Czech Republic, Prague, IEEE, Piscataway, NJ, 2006, pp. 109–114.
- [8] S. Olsen, J. Wang, A. Ramirez-Serrano, R.W. Brennan, Contingencies-based reconfiguration of distributed factory automation, Robotics and Computer-Integrated Manufacturing 21 (4-5) (2005) 379–390.
- [9] M. Abramovici, J.C. Göbel, P. Savarino, Reconfiguration of smart products during their use phase based on virtual product twins, CIRP Annals - Manufacturing Technology 66 (1) (2017) 165–168.
- [10] Oxford Dictionaries, Definition of context, available at <https://en.oxforddictionaries.com/definition/context> (accessed on September 5, 2017).
- [11] K.A. Hribernik, L. Rabe, K.D. Thoben, J. Schumacher, The product avatar as a product-instance-centric information management concept, IJPLM 1 (4) (2006) 367.
- [12] R. Rosen, G. von Wichert, G. Lo, K.D. Bettenhausen, About The Importance of Autonomy and Digital Twins for the Future of Manufacturing, IFAC-PapersOnLine 48 (3) (2015) 567–572.
- [13] M. Abramovici, J.C. Göbel, Savarino, P. (Eds.), Gebus, P. (Eds.), Toward Smart Product Lifecycle Management with an integrated Reconfiguration Management: Proceedings of the PLM17-Conference on Product Lifecycle Management (PLM17), Seville, Spain, July 09th - 12th, 2017, 2017.