Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

Procedia CIRP 47 (2016) 376 – 381

www.elsevier.com/locate/procedia

Product-Service Systems across Life Cycle

Improving Product-Service Systems by Exploiting Information From The Usage Phase. A Case Study.

Johannes Lützenberger[a], Patrick Klein[a], Karl Hribernik[a], Klaus-Dieter Thoben[b]

[a]*BIBA - Bremer Institut für Produktion und Logistik GmbH at the University of Bremen, Hochschulring 20, 28359 Bremen, Germany*
[b]*Faculty of Production Engineering, University of Bremen, Badgasteiner Straße 1, 28359 Bremen, Germany*

* Corresponding author. Tel.: +49-421-218 50126; fax: +49-421-218 50126. *E-mail address:* lue@biba.uni-bremen.de

Abstract

Nowadays the relevance of Product-Service-Systems (PSS) is increasing. Providing customers with products and supporting services suiting the customer expectations becomes a key-factor for being successful in the market. Contemporarily, a huge amount of data such as usage-data from sensors or Product Embedded Information Devices as well as customer feedback from social media, forums or blogs is already available for manufacturers of PSS. These data sources provide valuable knowledge about the customer's usage of products and their expectations and complaints.
The successful exploitation of PUI - Product Usage Information (such as sensor data or user feedback) becomes a key success factor for future product developments as they aid the development of PSS directly deriving from costumer requirements. But an efficient use of such knowledge requires the setup of PUI related analysis, filters and the identification of dependencies between PUI and design parameters or component attributes. Thus, transferring the usage information to design requirements is currently the major challenge in the process of developing successful PSS.
Currently, there is a lack of research for a systematic transfer of PUI into product design requirements. A basic pre-condition for a knowledge transfer is a formal and neutral representation of both sides, usage data on the one hand and design parameters on the other. An approach for such a neutral representation is KbeML, which is specified as a formal extension of the established SysML standard, enabling a linkage of PUI and formalized KBE models.
This paper provides a case study regarding the connection of PUI and KBE models in the branch of White Goods. The information gathered from sensors embedded in washing machines will be considered in order to retrieve improved design requirements for next generation washing machines. In a first step the product structure of a washing machine will be represented in a formal and neutral manner by using KbeML. This way the washing machine is formally described in terms of an assembly structure, and broken down into subsystems and eventually individual parts, which are defined by their relevant core parameters.
In addition, the derived sensor data will be formalized as a SysML extension. The linkage between both sides (product structure data and PUI) can be achieved by mapping design parameters directly to parameters and values provided by sensors. To enable an analysis of these parameters the modelling language will provide statistical elements (e.g. median) allowing the extraction of critical values (information) from data streams. This way a comprehensive modelling environment can be provided to corresponding stakeholders, supporting an effective and efficient application of usage data for the development of new or the improvement of existing PSS.

Peer-review under responsibility of the scientific committee of the 8th Product-Service Systems across Life Cycle

Keywords: Product Service Systems Design, PSS Design, KBE, Knowledge Based Engineering, Product Usage Information

1. Introduction

Providing customers with products and supporting services suiting to customer expectations has become a key-factor for being successful in the market. Vice versa, customers are getting used to have a more direct influence on new product generations. The increasingly accepted possibilities to provide feedback via social media (forums, blogs, help-desks, etc.) and

Peer-review under responsibility of the scientific committee of the 8th Product-Service Systems across Life Cycle
doi:10.1016/j.procir.2016.03.064

direct interaction with product-related services raise customer expectations to have an influence on the development of product successors, updates and/or product related services. The latter relates to the concept of product service systems (PSS) and its upcoming research challenges[1].

In parallel, the incorporation of sensors, PEIDs, and similar embedded IT components, contributes to an increasing amount of Product Usage Information (PUI), which becomes accessible for the development teams. This emerging information source is supported by the growing digitization of products. Technologies covered under the umbrella of internet of things, and cyber-physical systems respectively, support the possibility to collect such data (and increase the amount of data as well). The process of collecting, storing and analyzing data from customers and end-users of products with the goal to discover new needs or identify changes in usage patterns is called informatization [2]. Informatization is done to enhance existing products or service offers or the related service-level agreements (SLAs) back to the customer. Opresink et al. [2] systematize this process as an information feedback loop beginning with collecting and storing data from customers, analyzing it to create data about them, and providing information about new service offerings back to the customer, after which the loop is repeated. In this sense the presented approach can be perceived as a contribution to methodologically support such an information feedback loop.

Even if a huge amount of usage-related data is already available, the successful exploitation of such valuable PUI is often not used efficiently to improve future product generations. One major challenge can be seen in the challenge to link usage related data to product-related parameters in a way, that the corresponding design or product behavior will be influenced directly (in terms of an adapted tangible product) or indirectly (in terms its service components). Accordingly, a scientific classification of the presented findings relates mainly to product-oriented PSS[1], but the inherent extensibility of the approach does not exclude an applicability in context of industrial PSS (IPS²) [3], since it is in any case a major pre-condition to exploit successfully the amount of product-related usage information (PUI). An adequate filtering of provided data, as well as a structured approach to relate its output to product related and service parameters is required. Against this background, a modelling approach for a formal representation of the connection between PUI and Knowledge Based Engineering (KBE) KBE. KBE can be seen as the process of gathering, managing and using engineering knowledge to automate the design process by software support [4].

2. Background – specification of KbeML

As outlined in the following the approach proposes an enhancement of KbeML [5] to allow a formal and neutral representation of usage data on the one hand and design knowledge on the other. Originally, KbeML itself has been developed to overcome current shortcomings in the context of KBE, as being identified by different studies (e.g [6], [7]). The elements allow users to model a product structure of a given product using elements such as assemblies (CADAssemblies) or components (CADComponents) as well as their constituting parameters (CADParameters). (KbeML suggests the prefix “CAD” for a all design related elements. However the proposed specification is aiming to support the whole spectrum of CAx tools [5]).

Besides the representation of a product decomposition, KbeML is capable of representing equations and rules. Equations are represented in a neutral way with their own variables and constants. Variables can be linked to the predefined parameters of components (defined as part of the product-structure). This way, the computation base for KBE parameters of components can be codified and formally represented. The separation into parameters of equations and parameters of components allows to re-use equations and rules for different contexts if modelled in KbeML [5].

Technically, KbeML is defined as an extension of the modelling languages SysML [8] and UML [9]. SysML itself, which has become an established standard for systems engineering [10], has been defined as a so-called extension (of UML). One of the key advantages of the UML "standard family" lies in its inherent capability to be extendable. The needed meta-model layer is provided by default in the specification of UML itself. Hence, an adaption to domain specific needs (such as systems engineering or KBE) can be performed without contradicting the existing specification.

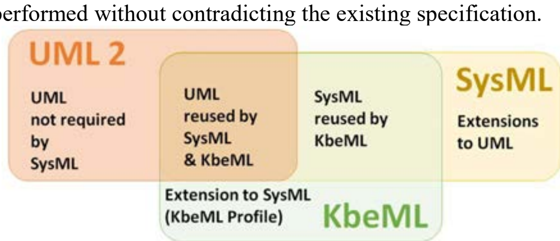


Figure 1: Relation between UML, SysML and KbeML[2]

The Venn diagram (Figure 1) visualizes the relationship between UML, SysML and KbeML. The region marked “Extensions to SysML (KbeML Profile)” indicates that new modelling constructs have been defined that have no counterparts in SysML or UML. The elements have been realized by defining stereotypes of existing SysML elements (e.g. Constraint Blocks). As outlined in section 3.2 a very similar approach is used for the so-called statistical elements to be used context of PUI analysis.

Since KbeML is grounded on the UML & SysML it can benefit from and be combined with other parts of the “Standard family”. As already documented in [5], a product development process for example can be represented with SysML behavioural elements, which are related to BPMN (ref. to [11]). Accordingly, the new elements, which are proposed below, are not isolated in sense of a new modelling language, but can be used in conjunction with other diagrams and modelling element (e.g. a requirements diagram to collect customer needs or activity diagrams to model the development process of PSS, etc.).

3. Enhancement of KbeML to represent usage information

So far, KbeML is focusing on KBE. Accordingly, it includes the representation of formalized engineering knowledge and the representation of the product itself, defined by its structure as a set of components and their constituting design parameters.

In order to represent usage information in addition to KBE knowledge it is planned to extend the current KbeML specification with new elements representing PUI and thus allowing to create linkages between design-related parameters and usage related parameters. The rationale behind these new elements as well as their formal definition is described in the following.

3.1. PUI and why there is a need for statistical elements

All data which is gathered within the usage phase of a product and relates to the usability product itself can become valuable input for the design of its successor. However, the acquired usage data, which usually comes from different sources, such as social media (blogs, forums, etc.) product embedded information devices or integrated sensors is usually referring to product instances, e.g. a review from a customer. In addition, the data is usually collected over time-periods e.g. a recording temperature values of a washing machine in household 1.

In contrast product design (except make-to-order) usually refers to a product class, e.g. a product line, or series and the required input parameters are usually represented by single values and not as a function over time.

Thus, in order to use the collected data within the design phase the acquired data sets need to be analyzed and filtered sufficiently. Afterwards the outcome of such a filtering process can be mapped to the design parameters of the product or to be more precise to the affected component. Established statistical functions, can be used for such a filtering (e.g. [12]). Some samples are provided in Table 1 along with their potential role in the given washing machine use case.

Table 1: Identified Statistical Functions (Extract)

Name	Description	Role in the Usage Scenario
Minimum	Minimum value of a range of data	The minimum weight of the load of a washing machine
Arithmetic mean	Value showing the central tendency of a range of data	On average the washing machine was loaded with a weight of 5.78kg
Mode	Most common value in the data set	Most of the washes were carried out with a load of 5.6kg.
Interval	An interval represents a range of data.	6,4 kg is the difference (range) between the maximum and minimum loading weight
Variance	A variance is the squared deviation from the mean.	4,6 kg is the average square deviation of the distance of the loading weight from the mean loading weight
Frequency	Absolute: An absolute value of the frequency	In 2014 the washing machine was 120 times used.

In reference to these functions new elements will be proposed for KbeML, which represent each a statistical function. These will allow the identification of critical values from data sets or data streams. In a second step the output values can be mapped to those parameters which are constituting the components and this way transferred into input-parameters for a KBE solution. For the definition of statistical elements, the extension mechanism of the SysML/UML standard family has been used.

3.2. Formal Definition of Statistical Elements

The statistical elements for KbeML will be defined as stereotypes of a SysML "constraint block". The formal UML/SysML extension is outlined in Figure 2.

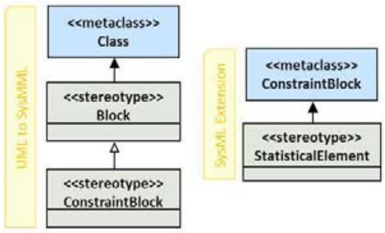


Figure 2: SysML Extension for Statistical Element

In the following the calculation of the arithmetic mean serves as an example. The formal representation of the arithmetic mean in alignment to the SysML notation is shown in Figure 3. Even not necessary for this sample, the modelling of the statistic element *arithmetic mean* covers two internal constraint blocks, each representing a basic calculation (Sum & division) to visualize that complex equations can be composed by simple base calculations in order to reduce complexity in diagrams. In a usage context, the arithmetic mean element will be visualized in a simplified manner (e.g. illustrated in Figure 7), since there is often no need to visualize the internal structure of an element.

In line with KbeML green is used as a unique color to identify the statistical elements in SysML/KbeML diagrams.

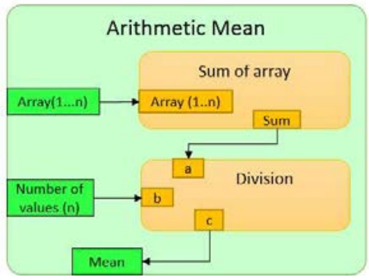


Figure 3: New element arithmetic mean based on SysML constraint block

With the "arithmetic mean" element an average value of a range of values can be calculated. E.g. if embedded sensors of a washing machine provide a range of water consumption values during several washes an average water consumption of all processed washes can be calculated using the equation of the arithmetic mean. Accordingly, a track of usage data such as water consumption can be transformed into a single value.

In order to process the calculation of the arithmetic mean the required input parameters are defined explicitly in the constraint block (Figure 3- small blocks in dark green). In a separate step, these parameters can be linked to external parameters, which can represent given data, such as usage data (*PUIParameters*), or design parameters (*CADParameters*), etc. In the provided example (below) the PUI parameters will represent data, derived from the sensors, which are embedded in the washing machine.

The separation between the constraint block parameters and the external parameters allows to define and reuse constraint blocks in different models, since the representation of knowledge has become independent from a single context [5].

Technically the modelled statistical elements can become classes of a "filter library" for data acquisition (implemented e.g. in JAVA). This library can be used by corresponding software modules on a business logic layer. Since SysML/KbeML are based upon an open format (XMi standard), an automated processing of the modelled filter processing becomes possible.

4. Sample Scenario - Identification of appropriate Bearings based on usage data

In order to illustrate the applicability of the KbeML (product structure, equations and PUI and statistical elements) and the proposed enhancement in terms of a statistic elements library to handle PUI, the design layout of washing machine bearings is used as an example.

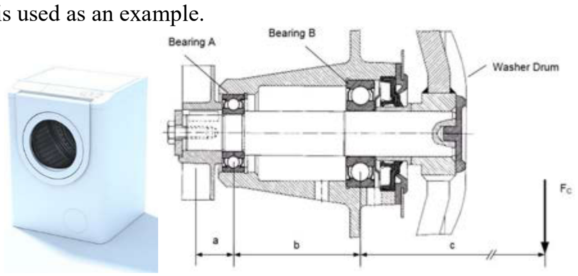


Figure 4: Washing Machine and tech. drawing of the drum mounted on a shaft with Bearings

As most of the products, also a washing machine is an assembly consisting of several components and sub-assemblies. Besides components and sub-assemblies such as the casing, heating elements or water filters, the washer drum and the main shaft with appropriate bearings are constituting components of the product washing machine (Figure 4). The components itself are defined by design parameters such as diameter, length or distance of shaft shoulders. In the given example, it is envisaged to base the identification of appropriate bearings on usage data derived from embedded sensors, providing data for Spin value, Load weight, etc. In concrete the weight of the eccentric load during washings as well as the spin value (number of revolutions) are required for the calculation of bearing lifetime. Therefore, a range of values derived from several washings has to be considered, which leads to integration of the statistical element *arithmetic mean* (refer to Table 1).

In Figure 5 an extract of the KbeML model of the "washing machine" is provided. The washing machine itself can be visualized as *CADAssembly* of several CADComponents, such as shaft, washer drum & bearings.

Geometrical parameters of *CADComponents* (e.g. drum diameter; or shaft length) are defined for each component.

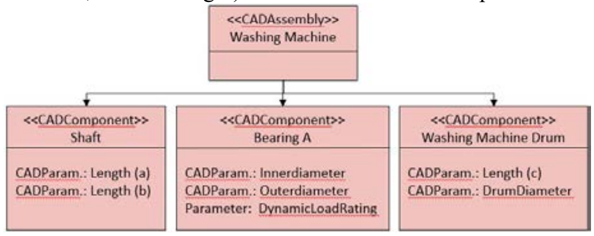


Figure 5: Formal representation of the product structure in KbeML (Extract)

While most CADParameters can be defined with concrete values (e.g. DrumDiameter = 500mm) the *CADParameter* OuterDiameter of the *CADComponent* Bearing A is defined without a value, since this parameter will be processed along with the selection of the bearing. In other words, it is one output of the formalized calculations.

4.1. Representation of the calculations using PUI

The calculation of the lifetime of bearings of a washing machine depends on the external forces acting on the bearings. The main force is the dynamic centrifugal force (F_C in Figure 4) during the spin cycle of the washing process. A simplified calculation process is presented in the following.

The equations are represented by the KbeML element *CADEquation* (which is based upon a SysML constraint block [3]). While the equation itself is encapsulated within the block, all parameters needed for the calculation (e.g. mass or diameter) are represented by mini-blocks (dark cyan). As outlined in Figure 6 connectors visualize the mapping between parameters serving as input parameters and equation related parameters (This way a reuse of equations in different contexts becomes possible).

As input for calculating the *CADEquation* centrifugal force the diameter of the washer drum is needed, which is part of the *CADComponent* washer drum in the product structure representation (Figure 5). Accordingly, the washing drum diameter is represented as a CADParameter (Figure 6).

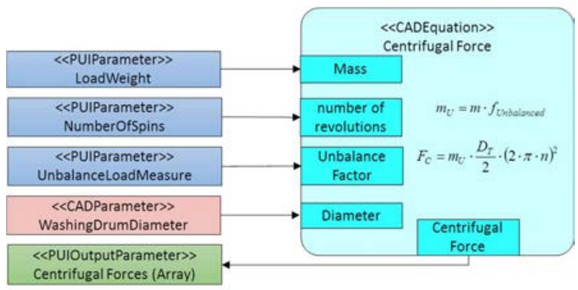


Figure 6: Formal representation of Centrifugal Force incl. input parameters

Additionally, the load weight, the number of spins and the unbalance measure are required and thus visualized explicitly each as a parameter. In our sample, corresponding data is derived from sensors and thus the Prefix PUI is used. Since the calculation will be processed for data sets from a range of washings, all *PUIParameter* are taken into account as arrays.

Based on the *CADParameter* and the *PUIParameters* the calculation of the centrifugal force can be processed. The result of this calculation is the *PUIOutputParameter* centrifugal force, which is of type array, since data from a range of washings has been used as input.

4.2. Formal Representation of the Bearings Constant calculation

In order to calculate the forces acting on the bearings the average of the array of centrifugal forces is calculated at first. Thus, the statistical element arithmetic mean has been linked to the *PUIOutputParameter* Centrifugal Forces and the *PUIParameter* Number of Measures (Figure 7):

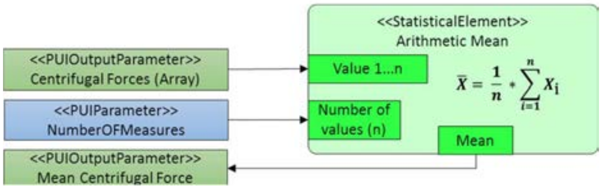


Figure 7: Calculation of Arithmetic Mean of Centrifugal Forces

The arithmetic mean of the centrifugal forces becomes the input for calculating the bearing load. Typically, the forces, which are acting on bearings, are identified using a free body diagram, which is transferred into mechanical equations based upon the geometrical dependencies.

Without detailing the moment-balance system it can be stated that in addition to the unbalanced force the *CADParameters* length (a), length (b), length (c) are required to have the system fully determined (Figure 8).

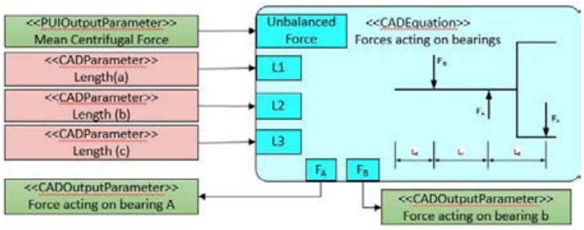


Figure 8: Calculation of Forces acting on bearings modelled in KbeML

The *OutputParameter "Force acting on bearing A"* is one major input for calculating the lifetime of bearings (represented as bearing load P in the equation – Figure 9). In addition, the lifetime equation requires *P, n, C,* L_h as parameters (thus visualized each by a rectangle). The dynamic load rating C is treated as an Output-Parameter, since this parameter enables the selection of an appropriate bearing in standard part catalogues. The lifetime L_h usually relates to product requirements or a specification sheet but can also be related to an average lifetime of past products. The number of spins *n* is also related to PUI data sets using the *Statistical Element* Arithmetic Mean.

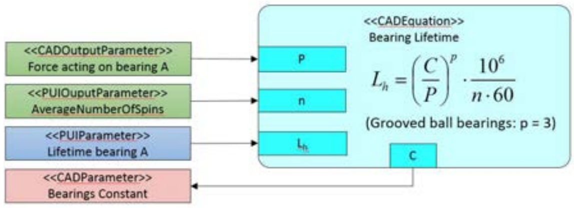


Figure 9: Representation of Calculation of Bearing Lifetime

As outlined in Figure 10 all *CADEquations* can be linked to each other resulting in a formal representation of the calculation process needed to identify the bearings constant, which can be used to select automatically an appropriate bearing in a standard parts catalogue.

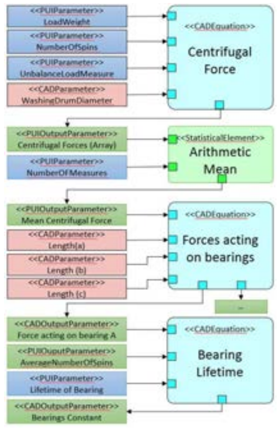


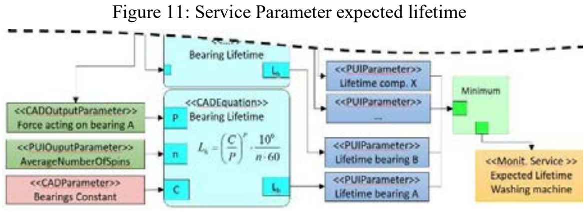
Figure 10: Design process for selecting the bearings based on PUI (Schema)

4.3. Feeding service-related parameters

Based on the sensor data from the washing machine and the user feedback not only the product can be adapted to consumer needs but also services can be created/supported. The monitoring of washing machine lifetime can be provided e.g. as part of such a service. Such a continuous monitoring may enable complex services for the user to individually adapt the programs of his/hers product and increase the overall lifetime.

In **Fehler! Verweisquelle konnte nicht gefunden werden.** the service related lifetime parameter has been linked to the KbeML model. The *StatisticalElement minimum* is used to provide the overall lifetime of the washing machine:

Figure 11: Service Parameter expected lifetime



Even though only a single service related parameter is modelled so far, the use case indicates capabilities to support a co-design [13] of product and services.

4.4. Modelling alternatives

In section 4.1, the approach for calculating the centrifugal force handles an array of centrifugal forces. The *StatisticalElement* Arithmetic Mean is used only once to define the calculation of centrifugal force. However, this is not the only correct solution. In Figure 12 an alternative approach has been visualized using KbeML (& *StatisticalElements*). The alternative modeling foresees to provide an "*Arithmetic Mean*" for each of the PUI parameters first and map only the OutputParameters to the *CADEquation* "Calculation of the Centrifugal" force.

The output values of the centrifugal force are diverging from each other, which indicates that the way how PUI is provided to the design phase is of notable importance.

The differences increase if the statistical element itself is exchanged by an alternative. If for example, the arithmetic mean is replaced by the *StatisticalElement "maximum value*", a significant impact on the bearing constant value can be noted.

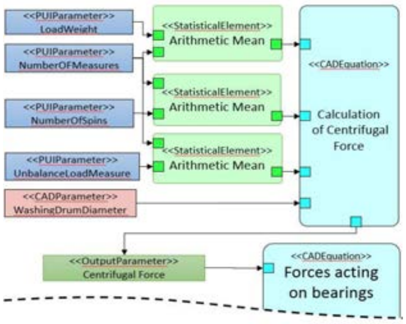


Figure 12: Alternative usage of PUI for the given design process

KbeML approaches support traceability and enables knowledge exchange, since the modelled process and the way, how usage information is analyzed, is visualized in formal diagrams based upon the established standards of SysML and UML.

5. Conclusion and Outlook

An efficient use of Product Usage Information (such as sensor data or user feedback) requires the identification of dependencies between PUI and design parameters or component attributes. Thus, transferring the usage information to design requirements is currently one major challenge in the process of developing successful PSS. In addition, the existing solutions, which are exploiting usage information often lack of transparency and documentation. In this sense, the effort for incorporating a feedback-loop for usage data into PSS design is contradicting the enormous potential for future design optimization.

Against this background, the development of a methodological approach for modelling and representation of product usage information in context of product design has been proposed providing support for an analysis of usage related data sets and their linkage to product design parameters on the one hand and to enable an exchange and reuse of such codified knowledge on the other hand.

The conducted research focused upon the extension of KbeML, specified as an extension to SysML and approaching to support Knowledge Based Engineering. KbeML elements have been enhanced with so-called *StatisticalElements* enabling the identification of meaningful (or critical) values from data sets derived from sensors. The formal approach yet appears a bit cumbersome at a first glance, but as outlined in the sample scenario the PUI information can be linked to product & service parameters in different ways. Hence, the way of mapping PUI influences the calculations and the formal representation may help to improve the tangible product- and intangible service-design sufficiently, since traceability will be ensured.

As a next step, research will continue in activities to push KbeML into the process for a formal standardization under the umbrella of OMG. In addition, the activities will concentrate upon the evaluation of modelled design alternatives and validating the positive influence of (the enhanced) KbeML on PSS design.

Acknowledgements

This research is part of the H2020 Framework Program, in the context of the FALCON project (Grant No. 636868).

References

[1] T. S. Baines, H. W. Lightfoot, S. Evans, et al., *Proc. Inst. Mech. Eng. Part B J. Eng. Manuf.*, vol. 221, no. 10, pp. 1543–1552, Oct. 2007.

[2] D. Opresnik, M. Hirsch, C. Zanetti, and M. Taisch, 'Information–the hidden value of servitization', in *Advances in production management systems. Sustainable production and service supply chains*, Springer, 2013, pp. 49–56.

[3] H. Meier, R. Roy, and G. Seliger, 'Industrial Product-Service Systems—IPS2', *CIRP Ann. - Manuf. Technol.*, vol. 59, no. 2, pp. 607–627, 2010.

[4] B. Prasad, 'What distinguishes KBE from automation', *COE NewsNet*, 2005.

[5] P. (1) Klein, J. (2) Luetzenberger, and K.-D. (1) Thoben, 'A PROPOSAL FOR KNOWLEDGE FORMALIZATION IN PRODUCT DEVELOPMENT PROCESSES', *80-10 Proc. 20th Int. Conf. Eng. Des. ICED 15 Vol 10 Des. Inf. Knowl. Manag. Milan Italy 27-300715*, 2015.

[6] W. J. Verhagen and R. Curran, 'Knowledge-based engineering review: conceptual foundations and research issues', in *New World Situation: New Directions in Concurrent Engineering*, Springer, 2010, pp. 267–276.

[7] W. J. Verhagen, P. Bermell-Garcia, R. E. van Dijk, and R. Curran, 'A critical review of Knowledge-Based Engineering: An identification of research challenges', *Adv. Eng. Inform.*, vol. 26, no. 1, pp. 5–15, 2012.

[8] OMG, 'SysML'. [Online]. Available: http://www.omg.org/spec/SysML/. [Accessed: 18-Mar-2016].

[9] OMG, 'UML', 2011. [Online]. Available: http://www.omg.org/spec/UML/. [Accessed: 24-Oct-2013].

[10] M. Bone and R. Cloutier, 'The Current State of Model Based Systems Engineering: Results from the OMG^{TM} SysML Request for Information 2009', in *Proceedings of the 8th Conference on Systems Engineering Research*, 2010.

[11] N. Q. Bảo, 'A proposal for a method to translate BPMN model into UML activity diagram', in *13th International Conference on Business Information Systems*, 2010.

[12] T. H. Wonnacott and R. J. Wonnacott, *Introductory statistics*, vol. 19690. Wiley New York, 1972.

[13] C. Durugbo, 'Strategic framework for industrial product-service co-design: findings from the microsystems industry', *Int. J. Prod. Res.*, vol. 52, no. 10, pp. 2881–2900, May 2014.