INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED13
19-22 AUGUST 2013, SUNGKYUNKWAN UNIVERSITY, SEOUL, KOREA

USING KNOWLEDGE BASED ENGINEERING TO SUPPORT THE DESIGN OF SMART PRODUCTS

Johannes LUETZENBERGER, Patrick KLEIN, Klaus-Dieter THOBEN
University of Bremen, Germany

ABSTRACT

If products are no longer considered as being mere physical devices, it has to be analyzed if KBE (Knowledge based Engineering) approaches can be adapted or if KBE approaches become obsolete for the design of the so called “smart products”.As a sub domain of Knowledge Based Engineering (KBE), Design Automation (DA) builds on the idea of deriving the physical design of a product automatically from codified, product related engineering knowledge. The authors believe that, by paying special attention to the potential interaction of products with different sorts of information and content, DA approaches can even play a major role for the development of smart products. Thus this paper aims to provide a concept for an enhancement of DA. Instead of case based and locally implemented solutions, the concept relies on a central knowledge-based system in order to process the smart layer on top of the geometrical design. The proposed system is be grounded upon an ontology in order to represent the physical and the virtual domain synchronously. Following this approach, different kinds of product development applications can rely on one central knowledge-base.

Keywords: smart products, design automation, KBE, user-centered design, collaborative design, ontologies

Contact:
Prof. Dr. Ing. Klaus-Dieter Thoben
University Bremen
Production Engineering
Bremen
28359
Germany
tho@biba.uni-bremen.de

1 INTRODUCTION

So called Smart Products offer the potential to become more intelligent and better suited to the requirements of the users. They can be defined as entities designed and made for self-organized embedding into different environments, thus providing improved products for user interaction. They offer the possibility to adjust functionalities during the usage phase and customize the product (on item level) and this way provide an added value to the user and at the same time provide new business opportunities to both manufacturers as well as service providers (Mühlhäuser 2008).

Expanding on the idea of Smart Products, current research proposes a user-centred and collaborative design. Such design would focus on the identification of interactions and services to fully reflect user requirements and preferences from early development stages, rather than improvements driven by technology (Hazenberg & Huisman 2012).

On the other hand Knowledge Based Engineering (KBE), has significantly improved product development especially in the domain of repetitive, routine tasks. According to (Skarka 2007) about 80% of the time during the product development phase is dedicated to repetitive tasks. That means, that the large potential of successfully implemented KBE solutions has already been validated by several research and development projects (e.g. for machine-tool design(Nacsa et al. 2005) or metal forging (Kulon, Mynors, et al. 2006)). A KBE solution relies on rules, formulas, constraints and other codified knowledge to “autonomously” derive the physical design of the product from the knowledge-base (e.g. a product shape derived from aero dynamical constraints). In other words KBE itself demands for codified knowledge. The deterministic approach of KBE seems to be in opposition to the user centred/collaborative design.

Even if KBE has already proven its validity, some important questions remain to be answered: Is it possible to support the development of Smart Products if the design is solely processed by knowledge-bases? Is KBE contradicting the paradigm of user-centred design?

2 FUNDAMENTALS OF KNOWLEDGE-BASED ENGINEERING (KBE) AND SMART PRODUCTS

In the following chapter the definition and requirements of Smart Products and the characteristics Knowledge-based Engineering - KBE are briefly introduced in order to analyse limitations of current KBE approaches and identify new requirements for KBE in the context of Smart Product development:

2.1 Smart Products

Smart Products can be defined as entities designed and made for self-organized embedding into different environments in the course of its lifecycle, providing improved simplicity and openness through improved product to user (p2u) and product to product (p2p) interaction (see (Mühlhäuser 2008)). The capabilities to interact can rely on context-awareness, semantic self-description, proactive behaviour, multimodal natural interfaces, Artificial Intelligence planning, and machine learning. A Smart Product is embedded within an environment that provides the intelligence to download process and store information on individual users, their prior interactions with products and the ability to create a context to p2u interaction (Mühlhäuser 2008).

In compliance with the definition above Maass and Varshney characterize Smart Products by several dimensions, which can be interpreted as requirements for Smart Products (Maass & Varshney 2008):

1. **Situatedness**: recognition of situational and community contexts
2. **Personalization**: in terms of tailoring the product according to buyer’s and consumer’s needs and affects
3. **Adaptiveness**: the ability to change product behaviour according to buyer’s and user’s responses and tasks
4. **Pro-activity**: anticipation of user’s plans and intentions
5. **Business-awareness**: consideration of business and legal constraints
6. **Network capability**: the ability to communicate and bundle with other products

Herewith different classes of Smart Products can be realized ranging from customer to product communication in order to support the selection of a perfectly suited product up to the ability to create pleasant experiences along the usage phase.

Different types of Smart Products require different enabling technologies, which are directly influencing the product development process itself (e.g. weather-conditions can be recognized by on-board sensors of a device or alternatively provided by an internet service). As mentioned above, the design has to focus on the identification and interpretation of interactions and services to fully reflect user requirements and wishes from early development stages, rather than improvements driven by technology.

Product development teams become not only responsible for the definition of a digital representation of the product, which enables adaptation to situations and consumers and its respective environment. But they have to identify the interplay between the physical and the virtual world (Meyer et al. 2009). The existence or absence of physical buttons on contemporary mobile devices can serve as a prominent example. The respective design decisions are neither technology/feature driven nor assembly or manufacturing related, but user product interactions are getting into focus. In this context, user centred design has become the driving force for product design (Hazenberg & Huisman 2012).

If the product development will shift its focus more and more to product user interactions in reference to changing environments, it will lead to new requirements for KBE solutions such as being able to model/represent environment characteristics, to set-up relations between interactions and environment, etc..

2.2 Knowledge-based Engineering (KBE)

Nowadays KBE is implemented on many levels in different industries: From simple templates in CAD software to extensive stand-alone software solutions with integration towards other CAx systems, there are many ways of implementing codified knowledge through rule-sets on product design. Within a KBE solution, engineering knowledge is represented in a formal manner and enables the system to automate specific development tasks. Each KBE system provides on the one hand an interface to capture the knowledge in terms of logical rules, algorithms, or constraints, and on the other hand an output module to trigger adjacent CAx systems or/and visualize results (Milton 2008). In this sense, KBE can be seen as the process of gathering, managing, and using engineering knowledge to automate the design process by usage of a KBE system (Prasad 2005). The meaning of “automate” even covers analysis tasks in terms of validation or quality checking, such as compliance to required safety parameters, or ISO standards. Next to time savings a KBE solution can enable a broader variety of detailed design studies of a given master-concept by usage of a rule-based approach for an automated detailing and examination of design variants and in consequence extensively support the optimization of a given (mechanical) design against defined constraints and requirements.

However, currently most KBE-solutions are still very much case based and not grounded in structural frameworks or methodologies (Verhagen & Curran 2010). By an analysis of more than 500 scientific publications in the area of KBE, further limitations of contemporary KBE approaches have been identified (Verhagen et al. 2012):

Many product developers seem to improvise a KBE solution based upon a customized development process and an unstructured problem analysis (Verhagen et al. 2012).

This kind of unstructured approach is followed by contemporary CAD systems. Leading CAD applications provide add-on modules (e.g. (IBM 2009)) for KBE related features. In such modules the KBE intelligence (e.g. a design rule) directly remains inside a CAD-model and is directly stored within the CAD file. Based on a parameterized CAD model, they provide functions like formulas (to create dependencies between parameters), rules and user defined features, allowing the partly reuse of design procedures (IBM 2009). Even if it would be possible to break up this encapsulation, which is given by the proprietary structure of such files, an utilisation of the already implemented design knowledge by other applications would fail, due to a lack of standardization of items, such as Namespaces (e.g. “surface” ; “shape”), Relations (e.g. “if … then …” ; “if ….else”) or Operators & Rules: (e.g. “if … then …” ; “if ….else”).

In addition to the encapsulation, Verhagen, et al. criticize that many KBE-solutions store and represent codified knowledge decoupled from its original context (Verhagen et al. 2012). An adequate documentation is missing and formulas or equations remain unexplained (Kulon, Broomhead, et al. 2006). The cause is often grounded in an unstructured knowledge acquisition process. Without a documentation of the problem in terms of objectives, constraints etc. the traceability of the design process of the implemented solution becomes impossible. Along with the insufficiency of a structured knowledge codification, a lack of knowledge reuse has been identified. Due to missing knowledge - e.g. neglected alternatives for a desired solution – KBE solutions are too often limited to their origin context and thus the reuse of knowledge will be hindered or impossible (Verhagen et al. 2012).

All of those KBE limitations (lack of openness, lack of documentation, lack of knowledge reuse, etc.) may not to be seen as super critical for contemporary solutions in context of KBE. But due to the interdisciplinary nature of Smart Products development (mechanical engineering, informatics, etc.) black box approaches or unstructured codification may become a key hurdle.

Of course it can be argued that Knowledge-based Engineering is still of importance on component or sub-component level (e.g. casings for a device), but the approach of KBE can even play a major role for the overall development of Smart Products in the near future. KBE enhanced models can be seen for instance as an enabler for easy and fast examination of design variants. If appropriate models can be provided, design variants can be of high value in context of user centred design; because it is an established idea to provide users with different kinds of virtual or physical mock-ups (e.g. (Bevan & Curson 1999)).

In order to achieve a support for Smart Products, KBE-systems must be enhanced and adapted, particularly in the sense of paying special attention to the potential interaction of products with different sorts of information and data (especially context-related). This is of course by no means possible, if the KBE intelligence (e.g. a design rule) directly remains inside an encapsulated "engineering"-model (e.g. solely stored in a CAD file), since different domains have to be involved in the modelling process (e.g. to represent a product User interaction in a park, the context in terms of loudness, brightness, needed functions, etc have to be modelled).

3 AN APPROACH TO BASE KNOWLEDGE-BASED ENGINEERING UPON MULTI-DOMAIN KNOWLEDGE

In order to enable KBE for Smart Products the knowledge-based system needs to incorporate knowledge from different domains: To represent the semantics of intelligent functions the system has to capture product as well as context knowledge (e.g. interactions under different environmental conditions). In addition other domain knowledge such as user-interface/ergonomics-related knowledge shall be captured as well (e.g. to identify optimized user interface elements in context of environmental conditions). Hence the approach demands for capturing knowledge of different domains in order to merge it into one integrated model.

The basic idea is to provide a central knowledge-based system on basis of description logics in order to integrate different domains. With respect to the ability to merge knowledge from different domains, ontologies are capable to enhance the base of KBE in terms of multi-domain knowledge. Such knowledge can be generic design knowledge (e.g. ISO-standards) as well as company specific knowledge (e.g. design guidelines). This way the product design is no longer limited to its physical dependencies, but to represent semantics of different domains in one integrated model.

The technology for the underlying IT-layer already exists: formal ontologies expressed in a formal ontology language (e.g. Web Ontology Language – OWL (McGuinness & Van Harmelen 2004)). It is common to use ontologies to add machine-readable meaning to (web-) content. Amongst others the so-called semantic web has become a prominent example. Herewith the idea is to provide the content of the WWW not only on behalf of humans but also according to software. Also other projects show that it is possible to represent knowledge specific to the individual domain (e.g. biology, geology, medicine, etc.).

Even in context of KBE respectively, ontologies have been successfully implemented (Skarka 2007; Ansaldi et al. 2006; Kuhn et al. 2012; Ruschitzka et al. 2010; Fei et al. 2011). Thus proving that it is possible to represent engineering knowledge for a KBE project. Own research activities already show the potential of ontologies to process rules and constraints for KBE (Franke et al. 2010). In this context it should be mentioned, that the "standard" ontology notation is very limited with respect to typical requirements of codified engineering knowledge: features are needed to compare values or parameters and enable simple calculations respectively. Hence it has become common for ontology related KBE approaches to rely on enhancements such as SWRL (Horrocks et al. 2004) or RuleML (RuleML Inc. 2012). In consequence the use of those enhancements makes sense also for the "smart product ontology" approach.

4 SAMPLE SCENARIO FOR MULTI DOMAIN KBE TO SUPPORT SMART PRODUCTS

In the following chapter a sample scenario is provided upon the envisaged approach addressing the development of a smartphone in order to visualize the concept and evaluated against the concepts and dimensions of Smart Products.

4.1 Description of Sample scenario

In assumption of a new kind of product development, which is primarily considering product user interactions in reference to changing environments, instead of a technology driven focus (refer to section 2), the scenario is delimited by users and their typical usage behaviour. In this scenario we differentiate between two individual users (user groups): a business user, who is interested in a smartphone providing a large display for business work and private user (hobby photograph), who is primarily interested in photography. Each user has to be represented with respect to his/hers preferences and characteristics, in order to identify the shape and function of a personally suited smartphone.

In this context an ontology can be set up, which includes a semantic representation of different domains, here the interaction domain and a physical domain:

1. Physical domain: product related dependencies, physical constraints, etc.
2. Interaction domain: representing context, user and usage scenarios interactions etc.

While the interaction domain is capable of representing non-quantified user requirements such as*: "I want a smartphone with excellent picture quality (private user)"*, the physical domain is representing quantifiable technical information such as dimensions and resolution of a camera sensor.

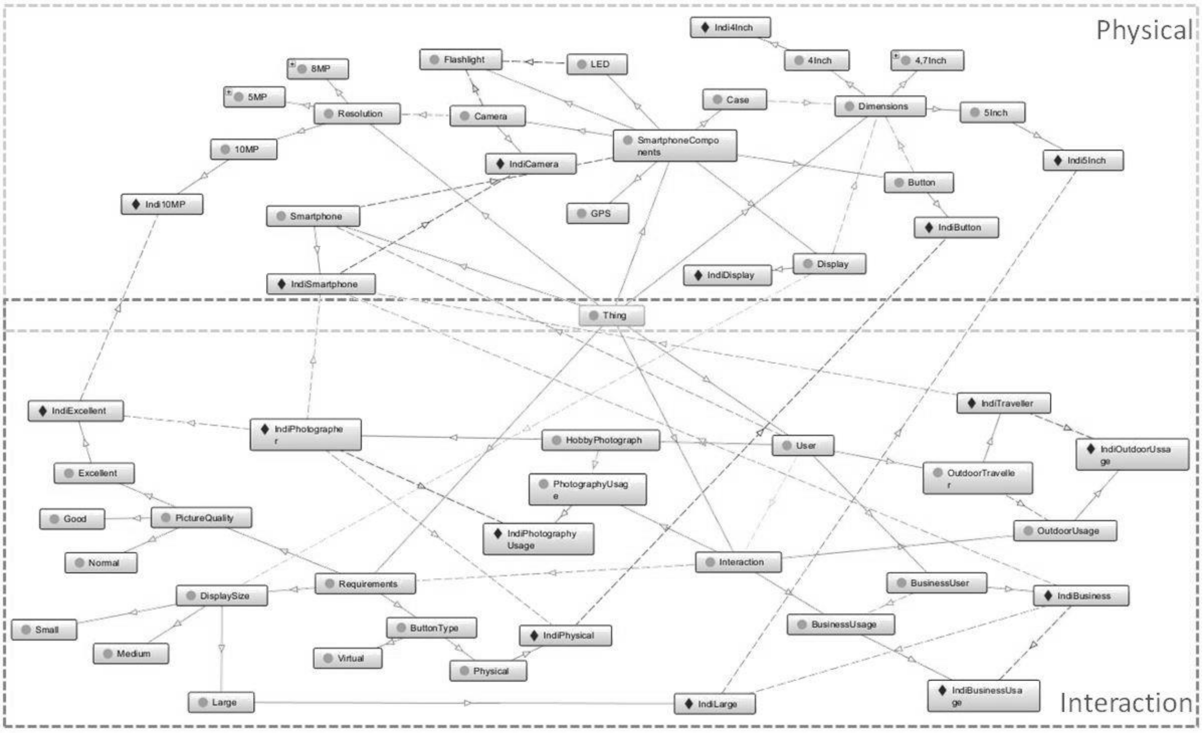

Figure 1: Ontology representing different users, interaction scenarios and smart requirements

Figure 1 illustrates an ontology, which includes both domains (physical and interaction) for the smartphone development. The ontology concepts have been grouped according to both domains. Concepts of the physical domain includes smartphone components, such as *casing, camera sensor, etc.* while the interaction domain is represented by *user, interaction, requirements, etc.*. As common in ontologies the semantics of concepts is provided by relations, such as *Interaction* has*Requirements* or *Requirements isPictureQuality*. This way user requirements can be defined explicitly but

independently from technological constraints (e.g. private user requires excellent picture quality) is modelled as *PictureQuality*, with no concrete resolution in mind.
By definition of relations between the concepts of both domains product related parameters can be linked to user related concepts. For instance the "excellent picture quality" has a semantic relation to the provided resolutions of the camera sensor (Figure 2). In case of availability of new technological components such as a camera sensor with higher resolution, only the technical part of the ontology has to be adapted. The requirement "excellent picture quality" remains untouched. And the concept *resolution* of *camera sensor* will be changed.

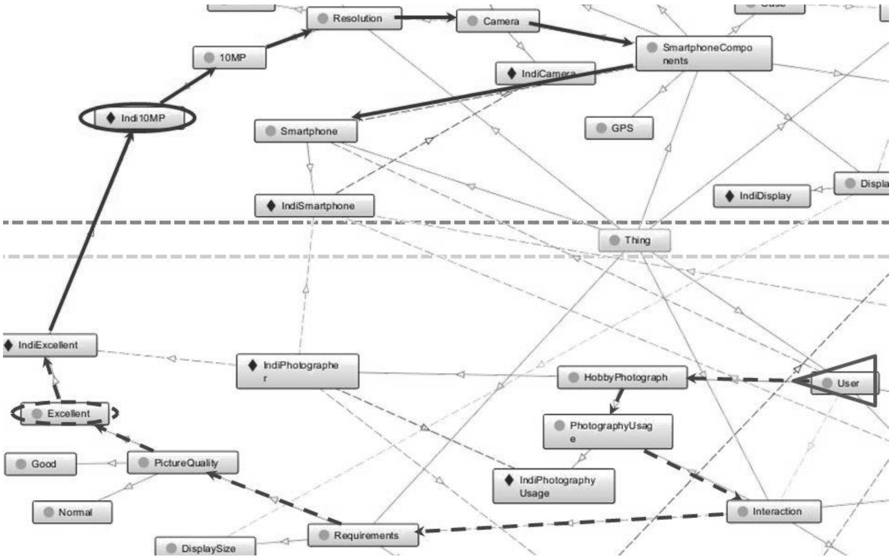

Figure 2: ontology extract: picture quality linkage to camera sensor

Further the shutter release functionality can be represented by a concept called button. As illustrated in Figure 3, a button can be either realized virtual (e.g. as a click-button within the GUI) or physical (e.g. as a push-button). While a virtual shutter release leads to a reduction of the usable space of the display a physical shutter release button has to be placed and integrated in the casing and may influence the overall product size (e.g. thickness of the smartphone). This can be further processed in terms of automated sizing of the geometrical representation of a button dimensions and the casing.

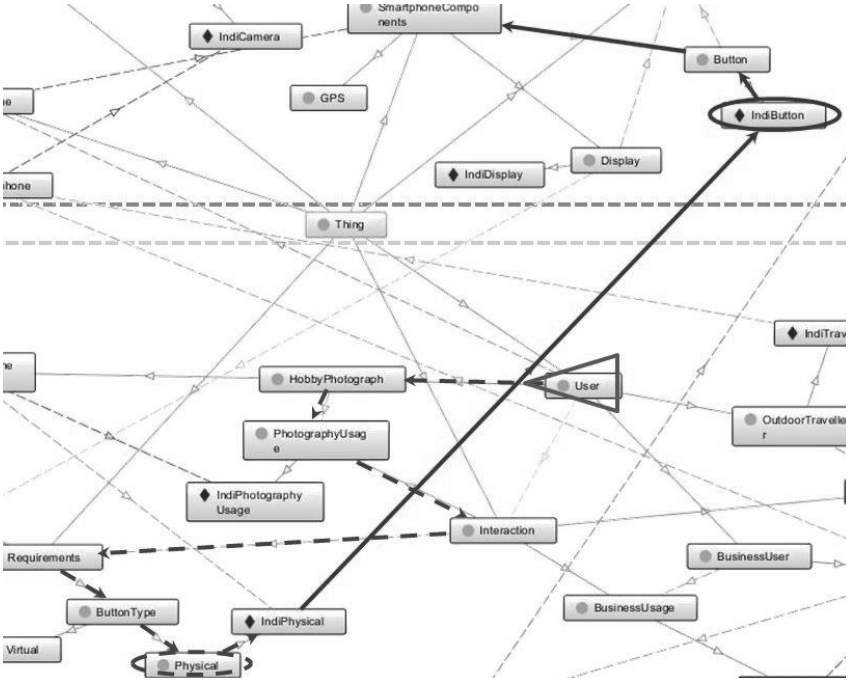

Figure 3: ontology extract: button type linkage to physical button

In this concept the product shape can be referred to a minor set of core parameters (for example “button.diameter”) by usage of the on-board features of contemporary CAD systems. Just the core parameter „button.diameter“ has to be delivered by the knowledge-based system (in our case the ontology). The value of the “button.diameter” may serve as an input for the associated CAD model, such as illustrated below (Figure 4). This way it becomes much easier to handle the Smart Product ontology.

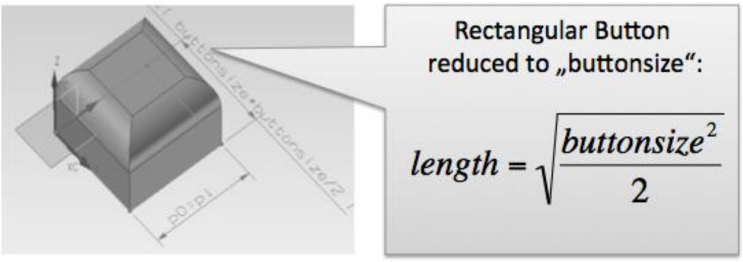

Figure 4: Reduction of geometry related parameters to a core set of parameters

The existence or absence of a physical button leads to different casings and smartphone layouts, rules such as below may complete the scenario:

If button.type = virtual *then* casing.thickness = casing.thickness – *function* (button.diameter)

With respect to the above mentioned easy and fast examination of design variants this multi-domain based approach enables to change the design and appearance of a product quickly just by switching attribute values. If for instance a new camera sensor becomes available only the technological part has to be updated the modelled user requirements (and interactions) remain as they are.

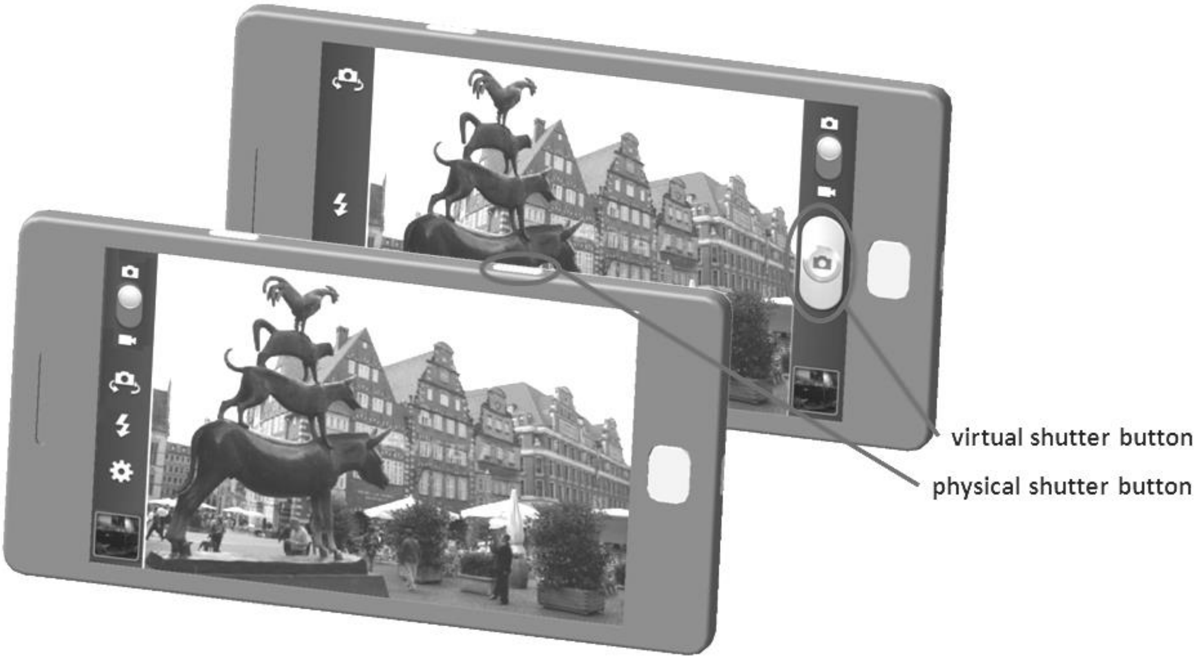

Figure 5: Smartphones with physical and virtual shutter

4.2 Scenario Evaluation

As shown by the initial scenario the description logic of ontology languages allows, that characteristics of a Smart Product can be considered in one integrated knowledge-base to be used in context of KBE. In the following each of the Smart Product “dimensions” is mapped to the capabilities of the multi-domain approach:

- ✓ **Situatedness**: can be represented, since concepts and rules representing and reflecting the situational and community context
- ✓ **Personalization**: can be represented, in terms of modelling personal needs and affects, e.g. by

rules such as: if person.impairment = true then button.type = physical (in any case)

✓ **Adaptiveness**: can be represented, since the ability to process rules in order to change product behaviour according to concepts representing user's responses and tasks

\- **Pro-activity**: is not supported (but it is not in contradiction to the approach)

✓ **Business-awareness**: can be represented, by considering business and legal constraints in the ontology

\- **Network capability: not applicable**

The six dimensions of Smart Products may not only be represented by a single domain such as the interaction domain but they can be represented by other - even already existing - ontologies. The semantic sensor network ontology can be such an example. Its description supports not only the physical structure of a device, but also the processing structure of the sensors. The sensor itself is not limited to a physical object, but can be seen as anything that can estimate or calculate the value of a phenomenon, so a device or computational process or combination could play the role of a sensor (W3C 2012). Since ontologies from different domains can be merged into one, there is a possibility to set up a knowledge-base by using existing ontologies.

As already indicated, using an ontology as a knowledge-base for KBE can improve the engineering processes for Smart Products. But at the same time to model an ontology can become very complex, in particular if a generic ontological representation of a product is envisaged. The ontology needs to be filled with product related knowledge from different domains. Further it has to be reflected, that the target-group is no longer limited to mechanical engineering. To define for instance software features and GUI elements demands for specific expertise not only from the field of product design but informatics, ergonomics etc.

Since the aim and competence of a product developer is not to build a knowledge-base, but to define a product or Smart Product respectively, it remains a critical issue to define concepts for enhancement and adaption of the knowledge-base. The interface to the knowledge-base has to become user-friendlier to ensure an acceptance by the end-user.

While contemporary CAD applications provide an user-friendly KBE interface (rules, constraints and dependencies can be directly modelled via input forms) they may be able to provide an appropriate access to the knowledge-base if the KBE-knowledge could be transferred automatically from CAD into an ontological representation.

As an initial proof-of-concept the linkage between a CAD-system and an ontology has been elaborated and specified as a so called Knowledge Acquisition Wrapper (KAW). As CAD models typically encapsulate design knowledge in proprietary file formats, the KAW prototype has been implemented exemplarily for the commercial (and commonly used) CAD software CATIA V5. However, the conceptual approach of KAW can be easily adapted to similar applications (such as Siemens NX).

Based on a CATIA programming interface KAW is able to gather the whole product-structure even on item level (points, lines etc.). Not only points and lines can be extracted, but also constraints and rules. This allows gathering all information starting from product-name up to rules and linkages between parameters Figure 6. Currently the output is just an XML-file. But as a next step it is planed to update and manage parts of the ontology of the sample scenario in order to prove the vision of a user friendly KBE interface for the ontology.

```
Formula.1: PartBody\Pad.2\Sketch.2\Radius.3\Radius=PartBody\Pad.1\Sketch.1\Radius.1\Radius
Formula.2: PartBody\Pad.2\Sketch.2\Radius.4\Radius=PartBody\Pad.1\Sketch.1\Radius.1\Radius +Ro
```

```
<Product>
<Value>Rohrflansch.1</Value>
<Type>Rohrflansch</Type>
    <Part>
        <Value>Rohrflansch</Value>
            <Formel>
            <1.Parameter>Rohrflansch\PartBody\Pad.2\Sketch.2\Radius.3\Radius<1.Parameter>
            <Operator>=<Operator>
            <Rellen Zahl>PartBody\Pad.1\Sketch.1\Radius.1\Radius </Rellen Zahl>
            </Formel>
            <Formel>
            <1.Parameter>Rohrflansch\PartBody\Pad.2\Sketch.2\Radius.4\Radius<1.Parameter>
            <Operator>=<Operator>
            <Parameter>PartBody\Pad.1\Sketch.1\Radius.1\Radius </Parameter>
            <Operator>+</Operator>
            <Parameter>Rohrdicke </Parameter>
            </Formel>
            <Formel>
            <1.Parameter>Rohrflansch\PartBody\Hole_test\Sketch.3\Radius.9\Radius<1.Parameter>
            <Operator>=<Operator>
            <Parameter>(PartBody\Pad.1\Sketch.1\Radius.1\Radius </Parameter>
```

Figure 6 Prototypical implementation of the Knowledge Acquisition Wrapper (KAW)

5 CONCLUSIONS

According to the findings to date, it can be stated that the development of Smart Products can directly benefit from KBE, if the latter is applied under consideration of specific constraints. However in some contemporary solutions initial hurdles exist, such as encapsulation of engineering knowledge in proprietary files. Nevertheless, it is possible to provide a support to the product development process of Smart Products by KBE, if these hurdles can be solved.

To achieve this, the authors have proposed an architecture, where Smart Product related knowledge is stored in a central knowledge repository and managed by a knowledge-based system. An ontology is proposed to become the core of the underlying IT-infrastructure. As drafted above, ontologies provide the required flexibility to represent classical engineering knowledge and at the same time the "interaction layer" domain. Even better the possibility to use existing (fully elaborated) ontologies as an integral part becomes possible. Consequently, there is no need to reinvent domain specific knowledge.

However the effort and skills, which are needed to model product related knowledge as an ontology, are to some extend challenging for product developers. His/her competencies relates to the design and development and not on formalization of knowledge. In order to avoid demanding new competencies in engineering, it is proposed to make use of already existing CAD KBE interfaces in combination with a KAW module. To enable this linkage the KAW has been implemented on top of a CAD API. It allows extracting KBE knowledge (such as a rule) into a neutral and interoperable format.

Future work will concentrate on the possibility to update and manage ontologies directly from a CAD system to prove the vision of a user-friendly interface for the proposed Smart Products knowledge-base.

It is expected that the ongoing research and standardization activities for ontologies and formal logic languages are going to have large impact on KBE and especially on possibilities to enable KBE in case knowledge from different domains has to be managed. The already noted demand for a "move from black-box applications (proprietary software) to applications with user-friendly knowledge-bases" will be challenged by these developments either.

Such a central knowledge-based repositories may become an integrative part of a PLM strategy, thus being implemented as services (as already proposed by (Fan & Bermell-Garcia 2008)). Grounded on this approach product development applications can not only use stored information (such as parameters and functions) in order to control the mechanical design, but in addition can gain benefits for further knowledge acquisition by reasoning and data mining.

ACKNOWLEDGMENTS

Part of this research has been funded under the EC 7th Framework Programme, in the context of the LinkedDesign project (http://www.linkeddesign.eu). The authors wish to acknowledge the Commission and all the LinkedDesign project partners for a fruitful collaboration.

REFERENCES

Ansaldi, S. et al., 2006. A knowledge-based tool for risk prevention on pressure equipments. *Computer-Aided Design & Applications*, 3(1-4), pp.99–108.

Bevan, N. & Curson, I., 1999. Planning and implementing user-centred design. In *CHI'99 extended abstracts on Human factors in computing systems*. pp. 137–138. Available at: http://dl.acm.org/citation.cfm?id=632800 [Accessed November 22, 2012].

Fan, I.S. & Bermell-Garcia, P., 2008. International standard development for knowledge based engineering services for product lifecycle management. *Concurrent Engineering*, 16(4), pp.271–277.

Fei, G. et al., 2011. A model-driven and knowledge-based methodology for engineering design change management. *Computer-Aided Design and Applications*, 8(3), pp.373–382.

Franke, M. et al., 2010. Ontological semantics of standards and plm repositories in the product development phase. In *Proc. 20th CIRP Design Conference*. pp. 473–482.

Hazenberg, W. & Huisman, M., 2012. *Meta Products: Building the Internet of Things*, Bis Publishers.

Horrocks, I. et al., 2004. SWRL: A semantic web rule language combining OWL and RuleML. *W3C Member submission*, 21, p.79.

IBM, C., 2009. IBM - CATIA Product Synthesis Discipline - All products in this discipline. Available at: http://www-01.ibm.com/software/applications/plm/catiav5/disciplines/prodsynth/products.html [Accessed December 8, 2009].

Kuhn, O. et al., 2012. Framework for the support of knowledge-based engineering template update. *Computers in Industry*, 63(5), pp.423–432.

Kulon, J., Broomhead, P. & Mynors, D., 2006. Applying knowledge-based engineering to traditional manufacturing design. *The International Journal of Advanced Manufacturing Technology*, 30(9), pp.945–951.

Kulon, J., Mynors, D.J. & Broomhead, P., 2006. A knowledge-based engineering design tool for metal forging. *Journal of Materials Processing Technology*, 177(1-3), pp.331–335.

Maass, W. & Varshney, U., 2008. Preface to the Focus Theme Section:'Smart Products'. *Electronic Markets*, 18(3), pp.211–215.

McGuinness, D.L. & Van Harmelen, F., 2004. OWL web ontology language overview. *W3C recommendation*, 10(2004-03), p.10.

Meyer, G.G., Främling, K. & Holmström, J., 2009. Intelligent Products: A survey. *Computers in Industry*, 60(3), pp.137–148.

Milton, N., 2008. *Knowledge technologies*, Monza: Polimetrica.

Mühlhäuser, M., 2008. Smart products: An introduction. *Constructing Ambient Intelligence*, pp.158–164.

Nacsa, J. et al., 2005. Knowledge management support for machine tool designers using expert enablers. *International Journal of Computer Integrated Manufacturing*, 18(7), pp.561–571.

Prasad, B., 2005. What Distinguishes KBE from Automation. Available at: http://legacy.coe.org/newsnet/Jun05/knowledge.cfm [Accessed December 2, 2009].

RuleML Inc., 2012. RuleML Homepage. *RuleML Homepage*. Available at: http://ruleml.org/ [Accessed November 28, 2012].

Ruschitzka, M., Suchodolski, A. & Wróbel, J., 2010. Ontology-Based Approach in Hybrid Engineering Knowledge Representation for Stamping Die Design. In J. Pokojski, S. Fukuda, & J. Salwiński, eds. *New World Situation: New Directions in Concurrent Engineering*. Advanced Concurrent Engineering. Springer London, pp. 227–235. Available at: http://link.springer.com/chapter/10.1007/978-0-85729-024-3_22 [Accessed May 15, 2013].

Skarka, W., 2007. Application of MOKA methodology in generative model creation using CATIA. *Engineering Applications of Artificial Intelligence*, 20(5), pp.677–690.

Verhagen, W.J.C. et al., 2012. A critical review of Knowledge-Based Engineering: An identification of research challenges. *Advanced Engineering Informatics*, 26(1), pp.5–15.

Verhagen, W.J.C. & Curran, R., 2010. Knowledge-Based Engineering Review: Conceptual Foundations and Research Issues. *New World Situation: New Directions in Concurrent Engineering*, pp.267–276.

W3C, 2012. Report Work on the SSN ontology - Semantic Sensor Network Incubator Group. Available at: http://www.w3.org/2005/Incubator/ssn/wiki/Report_Work_on_the_SSN_ontology [Accessed November 23, 2012].