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

EXTENDING PRODUCT MODELING METHODS FOR INTEGRATED PRODUCT DEVELOPMENT

Martin BONEV, Michael WÖRÖSCH, Dagný HAUKSDÓTTIR, Lars HVAM
Technical University of Denmark, Denmark

ABSTRACT
Despite great efforts within the modeling domain, the majority of methods often address the uncommon design situation of an original product development. However, studies illustrate that development tasks are predominantly related to redesigning, improving, and extending already existing products. Updated design requirements have then to be made explicit and mapped against the existing product architecture. In this paper, existing methods are adapted and extended through linking updated requirements to suitable product models. By combining several established modeling techniques, such as the DSM and PVM methods, in a presented Product Requirement Development model some of the individual drawbacks of each method could be overcome. Based on the UML standard, the model enables the representation of complex hierarchical relationships in a generic product model. At the same time it uses matrix-based models to link and evaluate updated requirements to several levels of the product architecture and to illustrate how these requirements have an upstream (towards stakeholders) and downstream (towards production) effect on the product architecture.

Keywords: product modelling, requirements, integrated product development, product architecture, product variant master

Contact:
Martin Bonev
Technical University of Denmark
Management Engineering
Kgs. Lyngby
2800
Denmark
mbon@dtu.dk

1 INTRODUCTION

1.1 Background

In today's global market competition, manufacturing companies are forced to keep up quickly with a dynamically changing competitive environment. Launching innovative products in accelerating development cycles becomes a crucial competitive advantage (Meyer & Marion, 2012). In order to achieve a high productivity in their product development (PD) process, firms are under pressure to employ suitable tools and methods, which allow an in-depth understanding and managing of knowledge related to the products, processes, but also to the project environment (Cooper & Edgett, 2008). To this end, both researchers and practitioners have put much effort in developing structured approaches on how to make the process of PD more efficient and thereby to reduce the development time and accomplish more successful results. Standardized procedures, methods and notations have been introduced, aiming at improving the management and collaboration of product development projects. Pahl & Beitz (2007) and especially the VDI-Guidelines 2221-2222 e.g. describe a stepwise procedure for product development, starting from identifying the design requirements to modeling the detailed design. The design process is hereby divided into individual steps, which can partly be performed in parallel (Simultaneous Engineering), while keeping a close contact to customers and suppliers. Similarly, Ulrich and Eppinger (2012, p. 2) define product development as a "set of activities beginning with the perception of a market opportunity and ending in the production, sale, and delivery of a product". Traditionally, these phases are performed separately and sequential, except for the detailed design step, which usually includes a number of internal iterations (Unger and Eppinger, 2011). In Concurrent Engineering (CE) all requirements products need to satisfy throughout their life cycle are captured already in the planning and concept phases. Since the majority of the cost is determined at this early stage of the design process (Whitney, 1988), having an overview of the complete lifecycle of a product may reduce all related cost from purchasing to product delivery significantly (Anderson, 2003). Accordingly, the Quality Function Deployment (QFD) and the Design Structure Matrix (DSM) have widely been utilized to identify customers' needs and to link them into the created product architecture (Vezzetti et al., 2011).

1.2 Research Problem and Objectives

Despite the great efforts within the modeling domain, the majority of methods described in academia typically address the uncommon design situation of an original product development of a single product, where the degree of design freedom remains rather high and solutions can be created independently from current product portfolios and product families. At the same time, studies illustrate that 70-90% of the development tasks are related to redesigning, improving, and extending already existing products (Encanação et al., 1990; Ullman, 1997). Existing design specifications are thereby adapted to satisfy new design objectives and constraints (Fowler, 1996). In addition, product development projects are yet increasingly dealing with rising product complexity (Malmquist, 2002). It has therefore become crucial not only to consider internal relations of the product structure (Eppinger et al., 1994; Lindemann et al., 2009), but also to include a number of different business aspects, such as mass customization strategies (Pine, 1993) and the use of commonality and product platforms (Meyer and Lehnerd, 2011).

To overcome these objections, this research attempts to further develop current modeling methods and techniques, to better meet challenges of designers. By considering up-to-date research and trends, the various aspects of an integrated PD, i.e. activities related to market, product and process are discussed (Andreasen and Hein, 1987). Existing methods are adapted and extended through linking updated requirements to suitable product models, capable of illustrating their effect on both the present engineering solutions and on the physical product and process structure.

2 METHODOLOGY

The presented study follows an action research (AR) approach defined by Coughlan & Coghlan (2002). Based on an initial literature review, this paper discusses current challenges and trends of modern PD projects, while particular attention is paid to the established methods and techniques that aim at addressing these challenges. A conceptual model is subsequently proposed, for better integrating upcoming requirements to the product development process. The model is finally tested and verified based on an industrial case. The collaborating partner is a consortium of five Danish companies and five research institutes, focusing on the development, production, and construction of pre-fabricated High Performance Concrete elements. Even though the organization is profit oriented, like most other companies, it has acknowledged the necessity to do upfront research in related areas in order to move the construction industry forward. Thus a rather innovative product development project has been initiated to create modular building components, that are based on platforms and which correspond to today's requirements. The industrial collaboration is realized through a mixed methods research, in particular through a qualitative dominant research with a sequential time order decision.

3 LITERATURE REVIEW

3.1 Knowledge Representation in Collaborative Product Development

In today's PD projects there is a growing communication concern to be handled. As a majority of the projects are being performed by working in teams, who frequently work geographically and temporarily independent from each other, related tasks have to be coordinated (Rodriguez and Al-Ashaab, 2004). An important implication of organizing collaborative product development is to be able to answer the question how a design change in redesign will affect the system, either organizational, product or process related (Tang et al, 2010). Traditionally knowledge about partial design solutions relied on the implicit knowledge and experience of individual design engineers (Suh, 2001). To keep up with the competitive environment, it has become important to make relevant knowledge explicit, thus available and shareable to all the parties involved in the development process. Companies which are able to integrate closely the various perspectives of the technical PD together with the required knowledge management will succeed in creating better products in shorter lead times. Product knowledge should represent the product features, their relation to the product components and the way how the created solution meets the marketing strategy. Process knowledge is about the involved business processes, the responsibilities and their interfaces towards supportive technologies. Eventually, project knowledge specifies the resources available, the functional and non-functional requirements, budgets, targets, milestones, and the like (Ebert and De Man, 2008). The implementation of adequate IT systems, such as Product Life Cycle Management (PLM) systems, hereby facilitates the efficient exchange and sharing of relevant knowledge (Vezzetti et al., 2011).

The discussed research demonstrates how much modern PD projects rely on adequate and explicit knowledge representation. The following sections investigate how this knowledge is outlined by related modeling methods.

3.2 Methods for Analyzing Product Development and Design Activities

3.2.1 Requirements Management

At the heart of any engineering discipline is the interplay between problem and solution domains (Chen et al, 2013). A requirement specifies what the product must do or defines a quality that the product must have (Robertson and Robertson, 2013). Compelling economic arguments justify why an early understanding of stakeholder' requirements lead to systems that better satisfy their expectations (Nuseibeh, 2001). Requirements Management (RM) proposes methods to cope with the requirements at the early phases of the development life-cycle. It presents concepts of identifying, collecting, and allocating "system functions, attributes, interfaces, and verification methods that a system must meet including customer, derived (internal), and specialty engineering needs" (Stevens and Martin, 1995, p.11). On the one hand RM consists of soft processes focusing more on people than products. This characterizes at the requirement elicitation process where requirements are discovered and the main objectives are about understanding stakeholders and discovering needs. When the problem domain is sufficiently well defined, on the other hand harder and more definite modeling techniques can take over (Alexander and Beus-Dukic, 2009). Since detailed descriptions for the requirement specification are typically created in various text based documents of considerable length, it can be difficult to get a sufficient overview of the requirements.

In RM requirements are typically grouped and graded according to their nature, e.g. implied or derived, and the impact the stakeholders have on them (DeFoe, 1993). Investigations on RM challenges have been reported repeatedly over the past years (Juristo et al., 2002). Requirements presentation, as well as incomplete and changing requirements and specifications are thereby seen as a major obstacle that needs to be overcome (Weber and Weisbrod, 2003). The process of moving between the problem world and the solution world is furthermore still not well recognized. Typically the effectiveness of a solution is determined with respect to a defined problem, however, the nature of the problem and its scope could depend on what solutions already exist or what solutions are plausible and cost-effective (Chen et al., 2013). Recent models suggest that instead of doing RM only at the early phases, requirements definition and design are interactive activities, handled simultaneously though the development life-cycle (Nuseibeh, 2001). RM therefore concerns much more than a list of "shall statements". Instead in modern approaches RM issues are engineered, involving tools, modeling, database design, customization with scripts, training, and data handling (Alexander and Beus-Dukic, 2009).

3.2.2 Matrix-Based Modeling Methods

Generally speaking, matrix-based modeling techniques help to classify the product structure, i.e. the relationship between elements. Through Quality Function Deployment (QFD) and the Axiomatic Design (AD) method designers can use a series of inter-domain matrixes (Malmquist, 2002) to transfer the requirements (the voice of customer) into specific product attributes, engineering characteristics, possible design solutions and manufacturing activities (Akao, 1990; Suh, 2001). Both methods provide guidelines for designers to make technical decisions more systematically (Hung et al., 2008; Jin and Lu, 1998), with the objective to design customer satisfaction and quality assurance into the product prior to production (Guinta and Traizler, 1993). Successfully implemented, such modeling methods have e.g. helped to increase competitiveness, lower start-up cost, and shorten design cycles (Kovach et al., 2007; Vallhagen, 1996). Further analytical techniques, such as the Design Structure Matrix (DSM) (Steward, 1981), have been developed to assess, reorganize, and cluster relationships between elements (Eppinger et al., 1994). In order to improve the analytical capabilities, the DSM method has since its introduction been further extended, modified, and integrated into other matrix-based approaches, such as the previously described QFD and AD methods (Guenov and Barker, 2005; Hung et al., 2008). From a solely inter-domain matrix with a limited capability of representing the nature of the relationships, over time the DSM method has increasingly been used on various intra-domain problems, namely in form of a Domain Mapping Matrix (DMM) (Browning, 2012), and in combination with fuzzy logic methods (Ko, 2010). Such DSM tools have been used from reorganizing static and time-based relationships (Browning, 2001) to support planning and scheduling activities (Shi and Blomquist, 2012).

In sum, RM methods – combined with matrix-based modeling techniques – are strong in handling the evaluation of customer driven requirements and a vast amount of static and time-based relations. As long as the relations are described on the same level of abstraction and the information flow goes from the customer domain to the process domain (Suh, 2001), the methods obtain powerful analytical qualities. However, the drawback of such techniques is that they hardly support platform design and product redesign (Malmquist, 2002; Simpson et al., 2010), which is, as previously discussed, a prerequisite for today's product development. The following two sections discuss briefly current approaches within these two domains.

3.2.3 Modeling Methods for Platform-Based Product Development

In mass customization, product specification processes consist of developing the needed specifications to deliver a customer specific product (Hvam et al., 2008). In this area great results have been achieved where customer needs are transformed directly into product designs and production specifications (Pine, 1993). When pursuing mass customization strategies, manufacturers aim at rationalizing their PD through implementing product family architecture based on product platforms (Jiao and Tseng, 1999). In this context, a product platform can be defined as a "set of subsystems and interfaces that form a common structure from which a stream of derivative products can be efficiently developed and produced" (Meyer and Lehnerd, 2011). Companies implementing platform strategies may among other things reuse parts, assemblies, technologies, concepts, and knowledge while simultaneously reducing unwanted complexity and improving their business potential (Andreasen et al., 2001).

While modeling product family architectures, different phases of the product development have to be integrated with the complying business functions. The formulation of a platform model involves considerations from several perspectives, the so called views. In the functional or customer view, the functionality of the product is first determined. The technical or engineering view then reveals how the functionality is provided and what technology has been applied. The physical view consequently describes how the product design is realized by the physical components (Jiao et al., 2007). In addition, to be able to access supply chain considerations, a supplementary representation of possible production layouts (production view) is needed (Mortensen et al., 2008). In order to be able to incorporate the different views of a product, generic modeling notations have to be applied that enable the representation of commonality, alternative variety, and ranges (Jiao and Tseng, 1999; Harlou, 2006). Such a generic modeling approach has for instance been pursued by Harlou (2006). The different perspectives and relationships are modeled with the Product Family Master Plan (PFMP) technique, also referred to as a Product Variant Master (PVM) (Mortensen et al., 2000). The method is based on the product architecture definition by Ulrich (1995), the theory of technical systems by Hubka and Eder (1988), and the theory of domains by Andreasen and Hein (1987). Similar to functional modeling (Jiao and Tseng, 1999), by following the basic principles of object oriented modeling, such as generalization, aggregation and association, the PVM technique uses the Unified Modeling Language (UML) standard to create a comprehensive overview of a product architecture (Hvam et al., 2008). With its additional notation, the method shows its advantages in modeling product platform and family architectures.

However, since relationships between elements are mapped only through direct connections (arrows) and constrains (for configuration), when linking all relations of complex products across the different views, the desired overview can no longer be provided. Hence, in the context of relationship handling, the PVM method does not seem to be capable in replacing the strong analytical techniques of a matrix-based model.

3.2.4 Product Redesign and Product Line Engineering

As discussed previously, development projects are rarely original, but are rather based on already existing products and technologies, which can sometimes be a group of similar products or defined as a product family (Smith, 2012). This means that a part of the development artifacts are new and a part of them already exists. For this type of development to be successful, it is therefore essential to be able to reuse as much as possible of the existing artifacts and to understand the relationship between the artifacts in each process step, e.g. requirements, design solutions, tests and processes (Shirley, 1990). Development projects can furthermore be technical, where new innovative solutions are first introduced for general applications and later to be used in actual products. In the case of internal projects, a common objective is to improve existing product structures and design solutions. From this end it is important to understand the upstream traces regarding how new solutions and designs affect the stakeholders (McGrath and McMillan, 2000).

The software society has addressed this issue by methods of Product Line Engineering (PLE) (Rabiser and Dhungana, 2007). In PLE the development process is split into two activities; (1) domain engineering, where the reusable asset is developed and (2) application engineering, where products are developed from the reusable asset in combination with fulfilling new requirements (Pohl et al., 2005). However, also PLE engineering research has reported that further studies are needed in application requirements engineering and in analyzing the relationship between requirements and the solutions (Rabiser and Dhungana, 2007). To facilitate research in RM and PD based on product families, inspired by the development approach of software, as in PLE, the following section introduces an extended modeling method based on the PVM. The method aims at combining the different techniques into one consistent framework and thus to benefit from their individual advantages.

4 PRODUCT REQUIREMENTS DEVELOPMENT MODEL

4.1 Introducing the PRD Model

When assessing the development task of a physical product from a redesign perspective, separately considered, each of the above described methods reveals a strong weakness in providing the essential overview and insight of requirements coming from different stakeholders and their effect on the product architecture. Supportive methods should be able to describe how the customers' requirements are realized, what engineering solutions have to be used, what is the physical structure of the products, and how are these produced. Since it is in particular important to make visual not only which, but also how parts are related, connected or assembled, hierarchical relationships and attributes have to be considered as well. Consequently, the presented Product Requirements Development (PRD) model builds on the existing capabilities of the PVM technique in mapping the stakeholder´s needs to design solutions. Based on an industrial case, the method addresses both, (1) how complex hierarchical relationships can be mapped and (2) how in turn a resulting product design may affect the stakeholders.

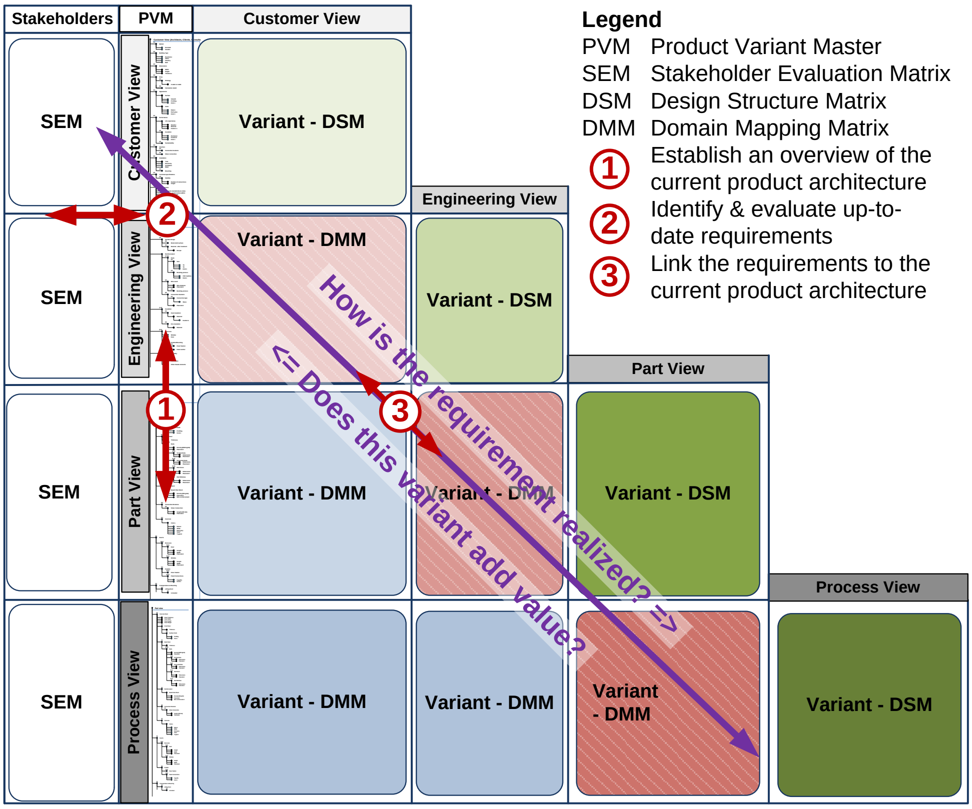

Figure 1 Product Requirements Development Model – Overview

A major difference between the product specification process in mass customization and the development of a new product in a product family is that the first one should fulfill the specific need of a single customer based on available solutions. The latter case needs to consider several stakeholders simultaneously, the impact of new requirements on the product architecture and the effort needed to realize the solutions are unknown. Here, the requirements from each stakeholder have to be evaluated in depth, as they need to be challenged, transformed, and tested by the designers. Since updated requirements have to be set in relation to the current product portfolio, it is eventually inevitable to have suitable models showing the existing product architecture in place. As illustrated in Figure 1, following the notation of the PVM technique, first, if not already available, a generic model of the product family at hand is created. With an additional “Process View”, life cycle considerations related to production, transportation and assembly can be included.

Next, similar to the QFD method, in a second step current stakeholder requirements are identified and directly modeled within the existing hierarchical product architecture of the PVM. As indicated in Figure 1, such requirements can appear in the different perspectives (views) of the model. The most common ones are typically driven by the market and are to be placed within the Customer View of the model. Technology driven requirements on the other hand are mapped in the Engineering View. Besides, requirements coming from other domains can potentially be mapped in the corresponding views. On the left side of the PVM, in the Stakeholder Evaluation Matrixes (SEMs), the requirements are graded and prioritized across the views according to their importance from 1 (low) to 5 (high).

The right-hand side of the PVM displays both, the downstream and upstream impact relationships. Complementary to the DSM and DMM technique, the effect of the requirements on other customer attributes, engineering solutions, physical parts, and processes can be mapped through inter-domain (Variant DSM) and intra-domain matrixes (Variant DMM). The difference to the well-known DSM technique hereby is that each side of the matrix is linked to the PVM structure, and therefore allows a concise expression of hierarchies and relationships, e.g. part-of or kind-of structures and attributes

(Hvam et al., 2008). Alternatively, to link hierarchies, variants and attributes with each other using standard matrix-based modeling methods, for each of the seen “Variant DSMs” or “Variant DMMs” a huge number of DSMs or DMMs is needed. Thus in order to obtain the overview of the resulting changes, at this point integrating the PVM technique with the DSM method appears to be beneficial. Having described the principal makeup of the PDM model, in the following paragraph the model will exemplary be applied on the case study.

4.2 Applying the PRD Model

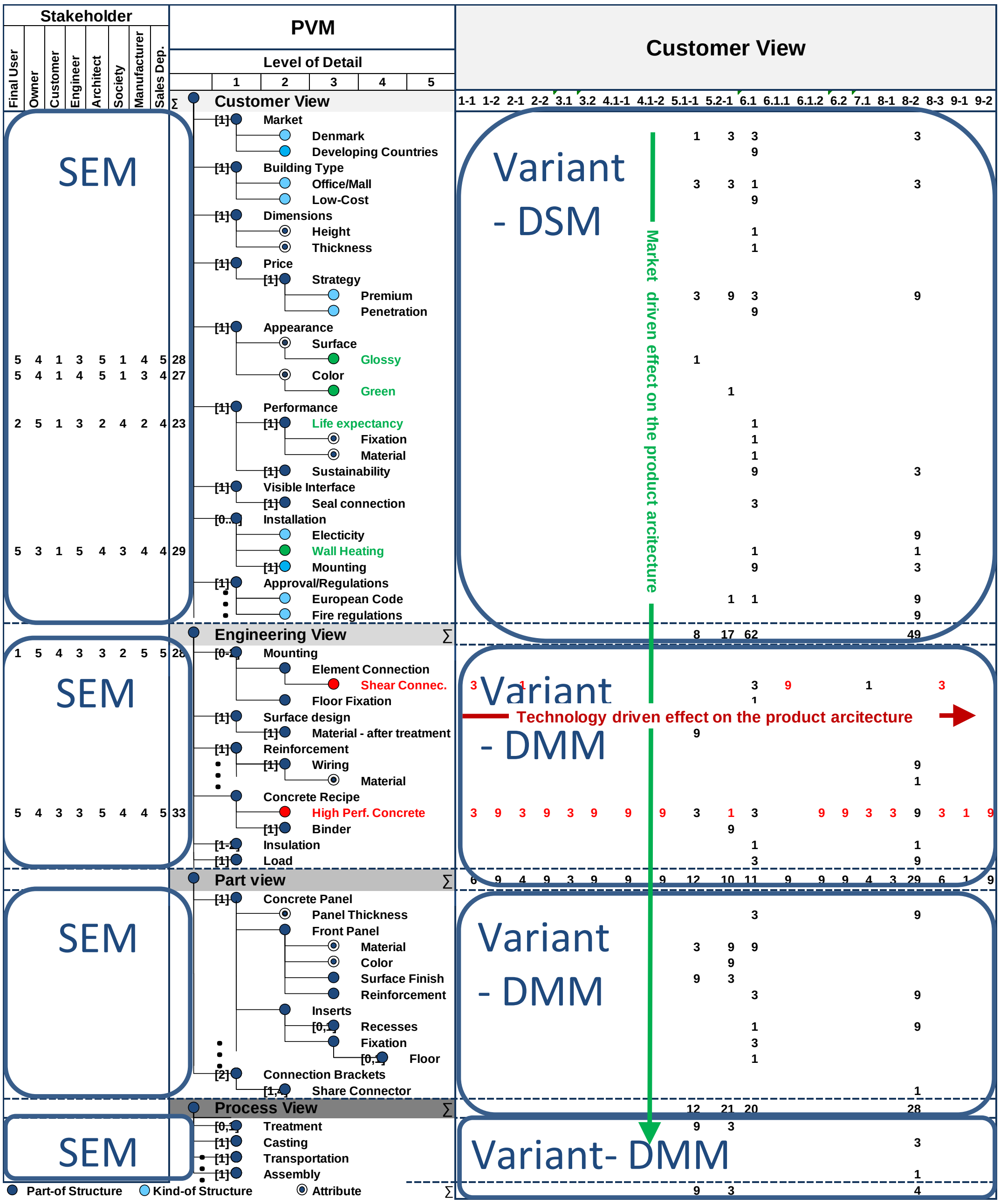

Figure 2 Requirements Evaluation

In the case example first (Step 1) a PVM model of high performance concrete sandwich elements has been created. Figure 2 illustrates a small segment of the entire model, where in Step 2 upcoming requirements were modeled directly into the established PVM. Market driven requirements were illustrated in “green” in the Customer View of the PVM. Here they e.g. concern a new surface and color for the concrete panels, as well as a different heating solution. Besides the requirements from the market, in technological development projects, requirements could also be triggered by the used technical solution as indicated by the “red words” on the engineering level (Engineering View). With the use of the different colors, change requests in the model could quickly be retrieved. Next, on the left-hand side of the PVM the stakeholders of the project were mapped into the described SEMs. In order to formally prioritize their preferences for all new requirements, their individual assessment was aggregated to the sum at the right-hand side of each SEM. Since in the case study all stakeholders had the same relative importance, no other proportional weighting for prioritizing the requirements was needed. It should be noted that in other cases different prioritizing strategies may exist. In some projects stakeholders may either have a greater voting right than others or other rather strategic aspects might be more important. Either way, at the end of this step arising requirements should be given a relative priority.

In Step 3, as illustrated on the right-hand side of the PVM, the impact of the requirements was modeled according to the fuzzy logic model. By grading the strength of the relationships with numbers (1, 3, and 9) (Ko, 2010), again it was possible to formalize how strong the effect of each requirement is on the current product architecture. Rather than only showing if there is a relationship at all, a higher number indicated a stronger effect. Equivalent to the active and passive sum of a matrix (Lindemann et al., 2009), for each Variant DSM or DMM, the total impact of each requirement was calculated at the bottom as the sum of the individual relationships. However, in order to obtain the overview, Figure 2 shows only partly the downstream effects of the requirements. For example, the impact on the stakeholders from the new “High Performance Concrete” (HPC) is depicted through the PVM structure of model. It has both a relatively high priority in the SEM and strongly affects the entire product architecture. “Life expectancy” on the other hand has been less prioritized by the stakeholders. Even though it has a significant effect in the Variant DSM in the Customer View, downstream traces (shown through the Variant DMMs) are less impaired. Another example shows how even more detailed requirements, such as the new “shear connection” can directly be shown within the model. Since “shear connection” is a part-of the mounting group, its indirect effect on a higher level of detail can be seen. In relation to the other requirements, it had a moderate priority from the stakeholders. But since it is not directly visible to the end users and affects a rather limited number of physical components, its impact on the remaining architecture is narrowed. All in all, by integrating the different modeling methods, this method shows how requirements have been graded by the stakeholders (upstream effects) and how they in turn affect the product architecture (downstream effects).

5 CONCLUSION AND FURTHER WORK

Product models, capable of representing how updated customer requirements affect the product lifecycle, enable designers to preserve the overview of the current product architecture, to better coordinate upcoming development activities, and moreover to plan and to calculate alternative solutions. By making use of established product modeling methods, such as the UML-based PVM, this paper contributes to an integrated PD process, which aims at better responding to the requirements of modern product development. Through the integration of several modeling techniques, the presented PRD model overcomes some of their individual drawbacks, e.g. the representation of hierarchical levels, product variants and attributes, while still being able to visualize correlations. Therefore, with the right integration, the PRD model expands the individual modeling possibilities. In sum, it (1) enables the representation of complex hierarchical relationships in a generic product model, (2) links and evaluates updated requirements to several levels of the product architecture, and (3) illustrates how these requirements have an upstream (towards stakeholders) and downstream (towards production) effect on the product architecture. However, in order to address all subsequent aspects of the PD process and therewith to explore the full potential of the model, further research needs to be done. It would for instance be interesting to investigate how matrix-based analysis methods, such as partitioning, could be solved with the Variant-DSMs and – DMMs of the model. Here, future research could for instance focus on what impact structural improvement of the product, through e.g. modularization, could have on the entire product architecture as well as on new requirements.

REFERENCES

Akao, Y. (1990) *Quality Function Deployment, QFD - Integrating Customer Requirements into Product Design.* Portland, Productivity Press.

Alexander, I., and Beus-Dukic, L. (2009) *Discovering requirements: How to specify products and services*. Chichester, Wiley.
Anderson, D. M. (2003) *Design for manufacturability & concurrent engineering: how to design for low cost, design in high quality, design for lean manufacture, and design quickly for fast production*. Cambria, CIM Press.
Andreasen, M. M., and Hein, L. (1987) *Integrated product development.* Bedford, IFS Publications.
Andreasen, M. M., McAloone, T., and Mortensen, N. H. (2001) *Multi-Product Development: Platforms and Modularisation.* Lyngby, Technical University of Denmark.
Bowning, T. R. (2001) Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions. *IEEE Transactions on Engineering Management*, Vol. 48, No.3, pp. 292-306.
Chen, L., Ali Babar, M., and Nuseibeh, B. (2013) Characterizing Architecturally Significant Requirements, *IEEE Computer Society*, Vo. 30, No. 2., pp.38-45.
Cooper, R.G., and Edgett, S. J. (2008) Maximizing productivity in product innovation. *Research Technology Management*, Vol. 51, No. 2, pp. 47-58.
Coughlan, P., and Coghlan, D. (2002) Action research for operations management. *International Journal of Operations & Production Management*, Vol. 22, No. 2, pp. 220-240.
Defoe, J. C. (1993) *An Identification of Pragmatic Principles - Final Report*, Seattle, International Council on System Engineering.
Ebert, C., and Man, J. D. (2008) Effectively utilizing project, product and process knowledge. *Information and Software Technology,* Vol. 50, No. 6, pp. 579-594.
Encanação, J. L., Lindner, R., and Schlechtendahl, E. G. (1990), *Computer Aided Design: Fundamentals and System Architectures*. Berlin, Springer.
Eppinger, S. D., Whitney, D. E., Smith, R. P., and Gebala, D. A. (1994) A model-based method for organizing tasks in product development. *Research in Engineering Design*, Vol. 6, No 1, pp. 1-13.
Fowler, J. E. (1996) Variant design for mechanical artifacts: A state-of-the-art survey. *Engineering with Computers*, Vol. 12, pp. 1-15.
Guenov, M. D., and Barker, S. G. (2005) Application of Axiomatic Design and Design Structure Matrix to the Decomposition of Engineering Systems. *Systems Engineering*, Vol. 8, No. 1, pp. 29-40.
Guinta, L. R., and Praizler N. C. (1993) *The QFD book: the team approach to solving problems and satisfying customers through quality function deployment*. New York, Amacom
Harlou, H, (2006) *Developing product families based on architectures – Contribution to a theory of product families*. Lyngby, Technical University of Denmark.
Hubka, V. and Eder, W. E. (1988) *Theory of Technical Systems*. Berlin, Springer.
Hung, H. F., Kao, H. P., and Juang, Y. S. (2008) An integrated information system for product design planning. *Expert Systems with Applications*, Vol. 35, No. 2, pp. 338-349.
Hvam, L., Mortensen, N. H., and Riis, J. (2008) *Product customization.* Berlin, Springer.
Jiao, J., and Tseng, M. M. (1999) A methodology of developing product family architecture for mass customization. *Journal of Intelligent Manufacturing*, Vol. 10, No 1, pp. 3-20.
Jiao, R. Jianxin, Simpson, T., and Siddique, Z. (2007) Product family design and platform-based product development: a state-of-the-art review. *Journal of Intelligent Manufacturing,* Vol. 18, No. 1, pp. 5-29.
Jin, Y., and Lu, S. (1988) Toward a Better Understanding of Engineering Design Models, in Grabaowski, H., Rude,S., and Grein, G. *Universal Design Theory*. Aachen, Shaker Verlag, pp. 71-86.
Juristo, N., Moreno, A. M., and Silva, A. (2002) Is the European Industry Moving toward Solving Requirements Engineering Problems? *IEEE Software*, Vol. 19, pp. 70-77.
Ko, Y. T. (2010) A dynamic planning method for new product development management. *Journal of the Chinese Institute of Industrial Engineers*, Vol. 27, No 2, pp. 103-120.
Kovach, J., Fredendall, L. D., and Cho, B. R. (2007) The interconnectedness among auxiliary benefits and supporting practices within the Quality Function Deployment process. *International Journal Of Six Sigma And Competitive Advantage*, Vol. 3, No. 2, pp. 137-155.
Lindemann, U., Maurer, M., and Braun, T. (2009) *Structural complexity management an approach for the field of product design*. Berlin, Springer.
Malmquist, J. (2002) A Classification on Matrix based Methods for Product Modeling. *Proceedings of DESIGN 2002*, Dubrovnik.

McGrath, R. G., and MacMillan, I. C. (2000) *Assessing technology projects using real options reasoning.* Philadelphia, Snider Entrepreneurial Center.
Meyer, M. H., and Marion, T. J. (2012) Preserving the integrity of knowledge and information in R&D. *Business Horizons*, Vol. 56, No.1, 51-61.
Meyer, M. H., and Lehnerd, A. P. (2011) *The Power of Product Platforms.* New York, Free Press.
Steward, D. V. (1981) The Design Structure System: A Method for Managing the Design of Complex Systems. *IEEE Transactions on Engineering Management*, Vol. 28, No. 3, pp. 71-74.
Mortensen, N. H., Yu, B., Skovgaard, H. J., and Harlou, U. (2000) Conceptual modeling of product families in configuration projects. *Proceedings Of Product Models*, Linköping.
Mortensen, N. H., Pedersen, R., Kvist, M., and Hvam, L. (2008) Modeling and visualising modular product architectures for mass customization. *International Journal Of Mass Customisation*, Vol. 2, No. 3/4, pp. 216-239.
N.N. VDI-Richtlinie 2221 (1993) *Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte*. VDI-Handbuch Konstruktion, Berlin, Beuth Verlag.
N.N. VDI-Richtlinie 2222 Blatt1 (1997) *Methodisches Entwickeln von Lösungsprinzipien*. VDI-Handbuch Konstruktion, Berlin, Beuth Verlag.
Nuseibeh, B. (2001) Weaving Together Requirements and Architectures, *IEEE Computer*, Vol. 34, No. 3, pp. 115–119.
Pahl, G., and Beitz, W. (2007) *Engineering design: a systematic approach*. New York, Springer.
Pine, B. J. (1993) *Mass customization: the new frontier in business competition*. Boston, Harvard Business School Press.
Pohl, K., Böckle, G., and van der Linden, F. J. (2005) *Software Product Line Engineering: Foundations, Principles, and Techniques*. New York, Springer.
Robertson, S., and Robertson, J. (2013) *Mastering the requirements process: Getting requirements right.* New Jersey, Addison-Wesley.
Rodriguez, K., and Al-Ashaab, A. (2005) Knowledge web-based system architecture for collaborative product development. *Computers in Industry*, Vol. 56, No. 1, pp. 125-140.
Shi, Q., and Blomquist, T (2012) A new approach for project scheduling using fuzzy dependency structure matrix. *International Journal of Project Management*, Vol. 30, No. 4, pp. 503-510.
Shirley, G. V. (1990) Models for managing the redesign and manufacture of product sets. *Journal of Manufacturing and Operations Management*, Vol. 3, No. 2, pp. 85-104.
Simpson, T. W., Siddique, Z., Jiao, R. J. (2010) *Product Platform and Product Family Design: Methods and Applications*. Boston, Springer.
Stevens, R., and Martin, J. (1995) What is Requirements Management? *Proceedings of the Fifth Annual International Symposium of the INCOSE*, Volume 2, pp. 13-18.
Suh, N. P. (2001) *Axiomatic design advances and applications*. New York, Oxford University Press.
Tang, D., Zhu, R., Tang, J. Xu, R. and He, R. (2010) Product design knowledge management based on design structure matrix. *Advanced Engineering Informatics*, Vol. 24, No. 2, pp. 159-166.
Ullman, D. G. (1997) *The Mechanical Design Process*. Singapore, McGraw-Hill Book Co.
Ulrich, K. T. (1995) The Role of Product Architecture in the Manufacturing Firm. *Research Policy*, Vol. 24, No. 3, pp. 419-440.
Ulrich, K. T., and Eppinger, S. D. (2012) *Product design and development*. New York, McGraw-Hill Higher Education.
Unger D., and Eppinger S. D. (2011) Improving product development process design: A method for managing information flows, risks, and iterations. *Journal of Engineering Design*, Vol. 22, No. 10, pp. 689-699.
Vallhagen, J. (1996) *An Axiomatic Approach to Integrated Product and Process Development*. Göteborg, Chalmers University of Technology.
Vezzetti, E., Moos, S., and Kretli, S. (2011) A product lifecycle management methodology for supporting knowledge reuse in the consumer packaged goods domain. *Computer-Aided Design*, Vol. 43, No. 12, pp. 1902-1911.
Weber, M., Weisbrod, J. (2003) Requirements Engineering in Automotive Development: Experiences and Challenges. *IEEE Software*, Vol. 20, No. 1, pp. 16-24.
Whitney, D. E. (1988) Manufacturing by Design. *Harvard Business Review*, Vol. 07/08 1988, pp. 83-91.