INTERNATIONAL DESIGN CONFERENCE - DESIGN 2010
Dubrovnik - Croatia, May 17 - 20, 2010.

COMBINING PROCESS MODEL AND SEMANTIC WIKI

A. Albers, B. Ebel and C. Sauter

Keywords: semantic, wiki, iPeM, process, documentation

1. Introduction

Increasing product complexity, global markets and shorter product life cycles are only a few reasons why the development of new products is a challenging task. A lot of knowledge is needed for and generated in product development processes. Knowledge management is becoming more important as a success factor for product development. However many knowledge management approaches and tools seem to have failed to fully fulfil the needs of designers and companies so far as most approaches could not reach successful implementation and long-term active participation of users in industrial practice of product development [Schütt 2003]. In this paper an approach is presented which suggests a combination of the integrated product engineering model (iPeM) and semantic wikis for supporting knowledge management in product development. The goal is to combine the advantages of semantic wikis regarding easy documentation, communication and structuring with a generic model for product engineering as a universal structure for documenting and accessing wiki contents. An implementation of this approach in an industrial context is presented and discussed. The remainder of this paper is structured as follows: The motivation and problem description are presented in chapter 2. Chapter 3 gives a short overview of some of the most relevant literature for this work including the integrated product engineering model (iPeM). Chapter 4 presents an approach for the support of product development process documentation and knowledge management with iPeM and semantic wikis. An example for implementation and validation of this approach in an industrial design project is presented in chapter 5. The results and experience from this implementation are discussed in chapter 6, followed by a short summary and an outlook on future work.

2. Motivation

Developing new products successfully and efficiently is a prerequisite for economic success in today's markets. The market and competition in which companies need to succeed have changed significantly since the middle of the 20th century. In order to survive in such conditions, companies need to be able to offer unique advantages to their customers which distinguish them from their competitors. Innovation is needed and only continuous innovation allows companies long term survival in dynamic globalized markets. Innovation is the successful implementation of an invention (a new product) in a market. Developing new and often complex products requires a lot of knowledge but generates a lot of knowledge as well. Being able to provide the necessary knowledge for a new development and the ability to efficiently apply the knowledge in a company is becoming an increasingly important successor factor. According to Klein, three main categories of design knowledge can be distinguished in the field of product development: (1) general domain knowledge, (2) case-specific object level knowledge, and (3) problem solving & control knowledge [Klein 1998]. General domain knowledge is relevant for several product development projects in a company. Even if products of a company are not very complex, the amount of general domain knowledge relevant for developing a product is often large. Even if all needed knowledge is stored somewhere in a company, it can be difficult to apply this knowledge if it cannot be accessed easily. Domain knowledge is changing more slowly than case specific knowledge, but it still needs to be complemented, structured and updated. The chance of reusing general domain knowledge from a core field of a company is higher than reusing case-specific knowledge. So the effort for coding and making general domain knowledge available for others may be slightly higher than for process specific knowledge. For accessing the general domain knowledge it is ideal if all of it is available in an explicit form. Large amounts of explicit knowledge need to be structured so that users can efficiently find what they are looking for. These knowledge structures can be either prescribed top down by the system or defined bottom-up by the users. The top-down approach helps clarity, consistency and easier navigation through a knowledge base. On the other hand top-down structures are normally less flexible and may restrict users when they add new knowledge and need to integrate it into an existing structure. The other possibility is to let users create structure freely in a more "bottom up" way. That means users define classification and relations freely for knowledge when they add it to the knowledge base. This may be faster and more convenient for contributors but there is a risk for inconsistency in the structure. Ideally knowledge processes support both types of structuring in a combined approach which is efficient for users when they are accessing knowledge and when they are adding knowledge to a domain knowledge base. The domain knowledge of the designer engineers can be assumed to change little in comparison to case-specific knowledge. So it seems reasonable to be distinguished and organized these two types of knowledge differently.

Case-specific knowledge can be all the knowledge that is needed for and generated in a particular development project. This knowledge is important for goal orientation, process transparency and the efficiency of the process, e.g. by avoiding rework or duplication of work and by allowing efficient communication in a design project.

Problem solving & control knowledge helps to plan processes and to define actions when the planned process deviates from the plan, as it is normally the case in product development projects [Albers 2010]. Design methods, problem solving processes, best practices for processes and reference process models are different forms of this knowledge. If all processes are described with a common language it becomes easier to learn from ongoing or past projects and to apply it to future projects. So ideally a good universal process model of product development processes allows descriptive as well as prescriptive process modeling. The influence of the process is relatively high on case-specific object level knowledge and problem solving & control knowledge, but significantly less for general domain knowledge. So it is important in which process context case-specific knowledge and problem solving knowledge are used or generated, but not for general domain knowledge. Therefore it seems appropriate to consider the process for structuring knowledge types (2) and as the case may be (3) but not for (1). Therefore, different structures are needed to allow users easy and efficient access to the different types of knowledge. Domain ontologies can be used for (1), a process ontology based on an universal process meta model is suitable for (2) and a combination of both for (3).

Wikis have become popular tools in knowledge management and semantic wikis combine the advantages for collaboration of social software with those of semantic software for structuring and adding machine-interpretable meaning. Process models provide structure for development projects and help to properly document them. So combining semantic wikis with process models could support knowledge management in product development.

3. Literature Review

Knowledge management in product development is a difficult task as it is a dynamic and iterative process which often involves ambiguity und uncertainty. Attempts to capture all knowledge relevant for a product in one KM-Tool have not succeeded [Schütt 2003]. These attempts often fail because they are not subjectively efficient for the designers. Using such a system often requires additional effort which does not justify the benefit from using such a system. Knowledge management tools should be easy to use and it should be possible to flexibly and efficiently integrate them into daily work. As new knowledge is continuously created in a product development process, it should be integrated into a knowledge base. Designers decide what is important and should be documented and what not. If the barrier for adding is high, they add less and most of it stays tacit knowledge, being more difficult to share und reuse. If the effort of using the KM system is felt to be higher than the benefits, even after an introductory phase, the system will not be used regularly and then is not an interesting source for others, so a vicious circle begins and the tool fails. If the barrier for sharing is low, the chances of documentation and reuse rise. Wikis have a low barrier for sharing knowledge as they are easy to use and knowledge can be instantly shared over a network. Process models can provide ontologies for structuring knowledge in a semantic wiki. The potential of the combination of semantic wikis and process ontology for knowledge management in product development has not been sufficiently explored.

3.1 Wiki

Wikis are software systems, which allow users to easily generate, publish and edit web pages, i.e. open content management systems. They are one way of enabling computer-supported cooperative work (CSCW). The first wiki was implemented by Ward Cunningham [Cunningham and Leuf 2001] (WikiWkikWeb) in order to easily exchange information for software development projects. His intention was to create “the simplest online database that could possibly work”. Wiki is a Hawaiian word for “fast” and expresses the intention to easily document, share and access information. Two main elements of a wiki-system are the wiki pages and the wiki engine. The wiki pages are created and edited by users. The wiki engine is the software system, which provides the functionalities for viewing, editing and publishing the wiki pages on a network. Today more than 100 different types of wikis (“wiki engines”) exist. Most of them are freely available and open source, some companies offer professional support for open source wikis, which is often a prerequisite for software in industrial applications. Besides the open source wikis there is also a growing number of commercial wiki software from large software companies like Microsoft, IBM or Atlassian. Wikis have become increasingly popular in recent years mainly due to the following advantages: Easy collaboration and formation of opinion, easy documentation and editing, easy cross-linking, simple structuring, full text search and often free and open source.

Wikis are currently being used in a wide range of private, public and commercial applications [Hinkelmann and Wache 2009]. The most popular and well-known is the online encyclopedia Wikipedia. Wikis have also become popular in education at universities and schools [Walthall et al. 2009]. Wikis have also been suggested for and used in product development.

3.2 Semantic Wiki

A Wiki is not automatically structured. Structure needs to be added by users. Semantic wikis provide the same functionality as normal wikis. Additionally they offer the possibility to enhance the contents of a wiki with metadata and thereby structure it [Völkl et al. 2006]. Depending on the type of wiki, the metadata is either derived from an ontology or defines an ontology. One advantage of using a semantic wiki is that wiki content can be retrieved more easily by queries not only on the text but also on metadata. Metadata can easily be included in the contents of a wiki page with a simple syntax or interactive fields and minimal additional effort for the contributor.

Two different types of semantic wikis have evolved in the last years. The one type uses a “bottom up” approach for structuring the wiki contents (e.g. “SemanticMediaWiki”). The types for the wiki content and the relations in the metadata are freely defined by the users while adding the wiki content. So the ontology slowly emerges out of the user input and grows continuously. This approach is flexible, easily expandable, causes less effort for the users but has the risk of redundancy, ambiguities and requires more “wiki gardening”. The other type of semantic wiki uses a top down approach (e.g. Semantic Wiki for Sharepoint”). Ontologies are centrally deployed and users have to “attach” the wiki content to the predefined ontology. This approach is less flexible but has a reduced risk of redundancy and inconsistency.

3.3 Process models

Process models are important for describing, planning and managing product development projects. Ideally a model for design processes supports all these three tasks and provides on the one hand a common language to describe models of different processes and to relate them so that it becomes

easier to compare them, to align the thinking about processes and to give a universal thought pattern for modeling design processes. Processes can be modeled as a set of interrelated activities. Using sets of generic activities as a basis for modeling processes allows a consistent and coherent description and makes it easier to achieve a common understanding of a process in a group [Sim and Duffy 2003].
Various development process models have been defined, often with different emphases. A good overview of design process models can be found in [Browning and Ramaseh 2007] and [Wynn 2007].
Process models can not only serve as the support for process description, planning an execution. They can also provide a structure for knowledge used and generated during a design process. If the process is also used as one structure for knowledge, than the reuse of knowledge from previous processes can be simplified. Not only the problem or system architecture but also the process could be used to specify the context for which knowledge is needed.
Organizational learning and knowledge management are one purpose of product development process modeling but further research is still needed to determine how process models are best used and integrated in the design process to support knowledge management [Browning and Ramaseh 2007].

3.4 Integrated Product Engineering Model (iPeM)

The integrated product engineering model (iPeM) is a comprehensive model for product engineering [Albers 2010]. It was developed to describe any specific engineering process from a descriptive point of view in a meta model to a prescriptive formulation for application. The meta-model describes product engineering as a system that consists of three main sub-systems: system of objectives, system of objects and operation system. The operation system generates objectives and transforms these objectives into a system of objects. The system of objectives describes all relevant objectives, their interdependencies and boundary conditions. The system of objects contains intermediate results of the engineering process, i.e. drawings, models, prototypes and also the actual marketable product.
In general, product development can be understood as problem solving. In the iPeM, activities are structured along two dimensions: the activities of product engineering from project planning to analysis of decommission and the generic activities of problem solving from situation analysis to recapitulation and learning. This matrix spans a generic set of activities for modeling product engineering processes and is also referred to as activity matrix (see figure 1).

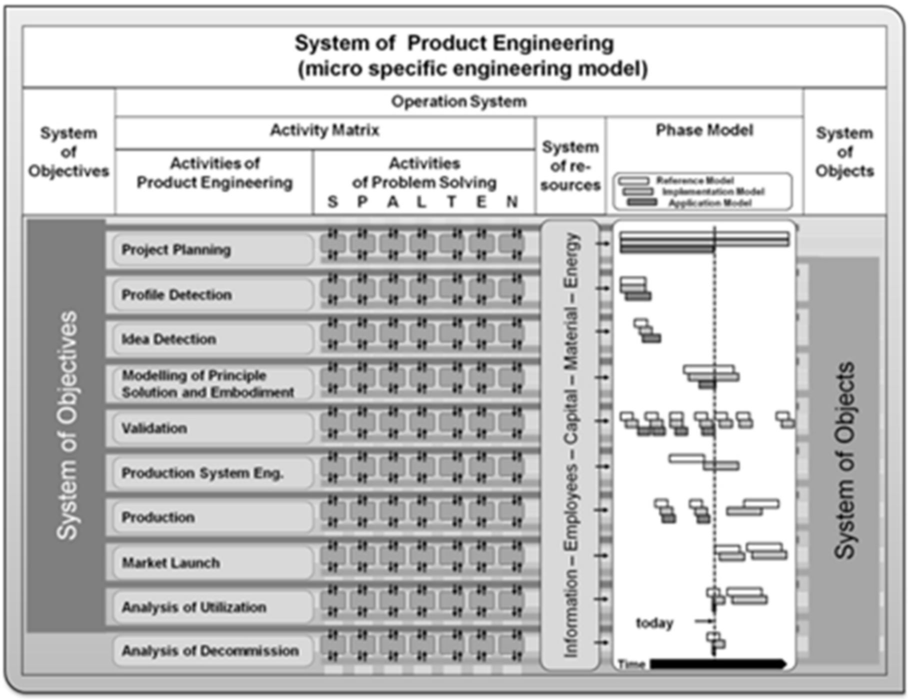

Figure 1. The integrated product engineering model [Albers 2010]

4. Approach

For today's product developing companies knowledge means competitiveness. As the process of product development is one of the crucial drains and sources of knowledge for such companies, a methodology is needed to accumulate and distribute the needed, respectively generated knowledge.

Therefore, one of the central points of this approach is the accumulation and distribution of knowledge in a product development process. An isolated approach that focuses only on this aspect is doomed to fail as the expected benefit would not outperform a folder system - neither paper-based nor digital. Rather, the attention has to be directed to the nature of a product development process as a project, i.e. the aspects of a determined duration and the uniqueness of the activity. That means, a successful product development process needs the company's general knowledge base but generates non-generic knowledge as well.

For project purposes the accumulation of this non-generic, specific knowledge is one key for success. Because of the determined duration of each project, the generated knowledge is quite often a combination of the accumulated information and the ability of the involved persons to turn that information into knowledge. For company purposes the accumulated non-generic, specific information is of minor priority. As there is always a part of this specific information that can be easily turned into generic knowledge. This generic knowledge unconditionally needs to be mirrored back into the company's knowledge base.

Supporting drains and sources of knowledge in the context of a product developing company accordingly means:

- establishing a company's knowledge base that distributes knowledge to projects
- establishing a reservoir that accumulates the project's specific information
- establishing the possibility of mirroring back the project's generic knowledge
- avoiding the common interfaces between the three preceding aspects.

The integration of the company's and project's dimension into one approach and the choice of a flat hierarchy seem to be the only possibility to satisfy those four requirements. Therefore, a flexible, highly networked system is needed to realize these requirements. This could for example be a wiki-system (Figure 2).

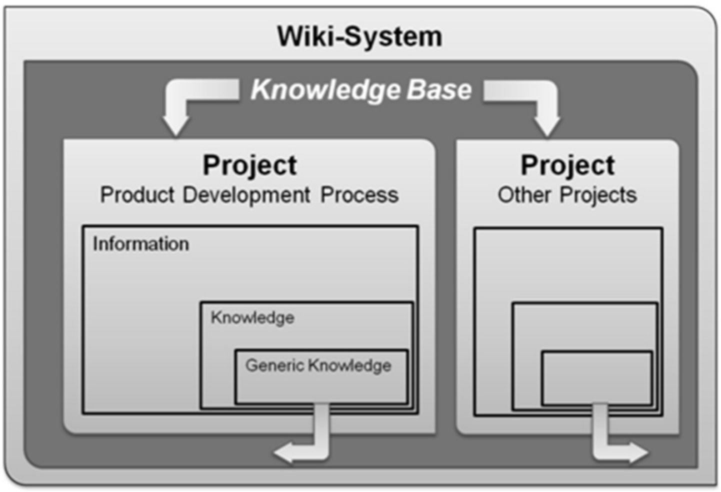

Figure 2. General approach

As mentioned before, the intention of this approach is to accumulate and distribute knowledge. Neither of these two central activities can be executed actively by a wiki-system, but the system can support the user's intention passively.

The accumulation and distribution of knowledge by a wiki system is per se well supported by several characteristics, e.g. easy editable, cross-linked pages, clear arranged page content, powerful search function et cetera. Having in mind a system, that manages the company's whole explicit knowledge, a solid approach has to be established.

4.1 Knowledge Base

It's quite easy to put some knowledge into a knowledge base – often it is not that easy to find the needed and basically available knowledge, particularly if the plenty of explicit knowledge at least almost correlates to the entire knowledge of the whole company. The available explicit knowledge needs to be structured in such a way, that the specific content corresponds to the structural elements. An ontology is needed.

With the help of an ontology each fragment of knowledge can be classified and more easily identified. To achieve this basic demand, the ontology needs to model basic aspects of the product and the involved domains. The benefit of such an usage of the ontology emerges with a huge amount of explicit knowledge available. But there are further more benefits, using an ontology - even if the amount of knowledge is still sparse. Linking knowledge explicitly or connecting problem/solution-patterns are only two of them.

To implement an ontology into the wiki-system, a semantic wiki is necessary.

4.2 Project Documentation

Being aware of a project's objectives, status, reasons for decision and the alternative solutions is one of the most important factors of success. Both, effectiveness and efficiency of a project depends on this factor. If the project is a small one, i.e. with a duration of a few months, only a few persons of the same or the bordering domains are involved, with a non-critical product complexity and so on, the human's cognitive ability is more or less sufficient to handle the amount of information. However, most of the today's projects in a product developing company are not small in this sense. Therefore, a project documentation is needed that helps the involved persons to keep track of the project's progress:

- Establishing a consistent and networked system of objectives
- Laying open the project's state including a retrospective/chronological aspect (process)
- Being robust against changes of the objectives in particular against those, that lead to iterations

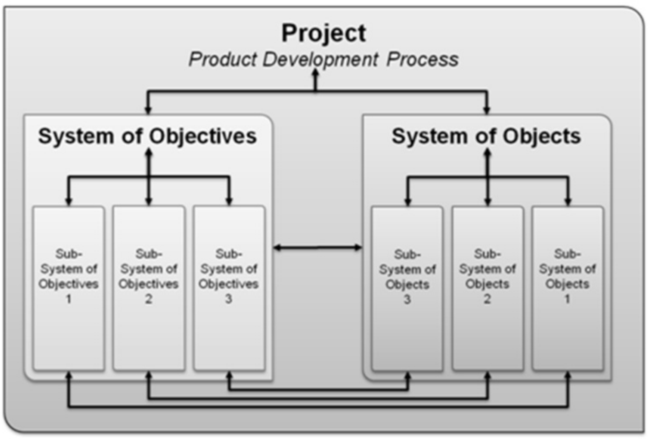

Figure 3. General project documentation

Figure 3 shows the general design of the project documentation according to the integrated product design model (iPeM). The clear separation of the objectives and the state of a project helps to develop a consistent target system and to keep track of the project's state. A cross-linking between the project(-overview) and its systems of objectives and -objects as well as a linking between the two systems combines those three aspect to one consistent approach.

This first element of the project documentation allows a purposeful development process. The second element needs to lay open the project's state. Every project of product development fallows a certain process, so a meta-model of the product development process allows the structured documentation of the system of objects' evolution. By the use of the activity matrix as a part of the iPeM, the documentation of the system of objects takes place by tracking the single activities that lead to the corresponding evolution (see figure 4).

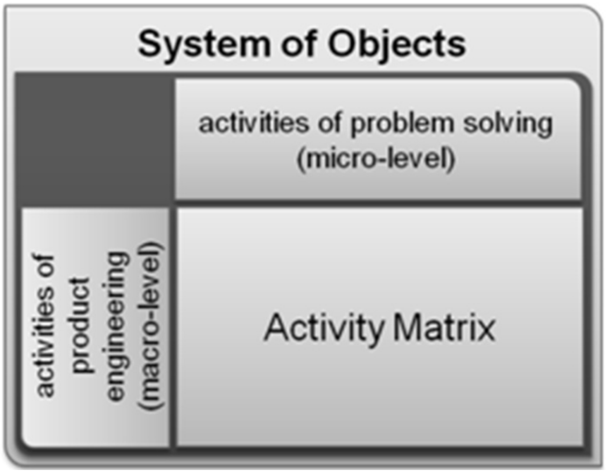

Figure 4. Usage of the activity matrix

With the evolution of the system of objects a second possibility of structuring this system gains in importance: the product's functional structure. With this third element of the project documentation a functional-based subdivision of the necessary activities is realized. The implementation of hierarchic systems of objects calls for corresponding sub-systems of objectives, whereas each of those sub-systems contains the specific, adapted objectives for each particular sub-system of objects. To maintain the consistence of the approach, the added sub-systems need to be integrated into the networked links (see figure 3).

The fourth element of the project documentation is a procedure to handle iterations in a product development process. An iteration becomes necessary if the system of objectives expands and is becoming inconsistent. The range of the iteration varies from a simple adaption of a single variable to the elimination of a whole product concept or -idea. In either case the reason of the iteration and the affected part of the former system of objectives and -objects need to be preserved to comprehend the new alignment. Therefore the affected data of the system of objectives and the affected activities of the system of objects are hidden with the ambition of creating a fictive linear-chronological development process (see figure 5). Finally, the outsourced data needs to get linked to the corresponding system of objectives and -objects to achieve the consistence of both, the systems and the documentation.

With the use of the existing ontology, extended by the aspects of the iPeM (system of objectives, system of objects and activity matrix) and the functional structure of the system of objects, an enhanced access can be realized.

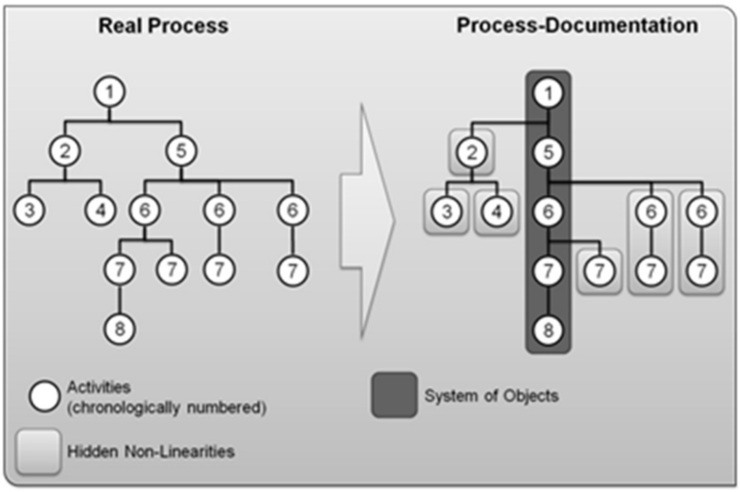

Figure 5. Documentation of iterations

5. Implementation and Validation in Industrial Environment

The implementation and validation of the expounded approach took place in cooperation with one of the leading solution providers for the print media industry.

To receive a valid statement of the practicability and usability of the approach the implemented wiki-system was used during a complex predevelopment project that was embedded into the research and development division of the cooperating company. The project was accompanied from the beginning of the early stages, this means that the market demand was already identified and essentially quantified, to the ending of the final design including a digital mock-up. Because of the already mentioned complexity and the limited period of time (six months) at least three engineers were involved at the same time. Those mechanical engineers, that formed the core-team of the project, were temporarily supported by several experts of the bordering domains (e.g. electrical engineering, control engineering, and informatics). Therefore not only the product itself was characterized by complexity but the product development process as well.

The concrete implementation of the approach described in chapter 4 considering these terms and conditions was realized by a MediaWiki-engine. By the use of this choice the plenty advantages of the well known Wikipedia encyclopedia in terms of accumulation and distribution of knowledge are intrinsically available. In order to enhance these advantages the possibility of adding Extensions on the basic MediaWiki was used. The integration of the SemanticMediaWiki- and the Halo-Extension allowed the easy use of semantic annotations and thereby an enhanced distribution of knowledge. The use of semantic annotations was generally constricted by a lean ontology that provides the basic elements of the approach such as macro- and micro-activities as well as those elements that were related to the product itself (top-down planning). A continuous, regularized extension of the ontology was realized by the users of the semantic MediaWiki throughout the whole use of the wiki-system (bottom-up planning). Beyond the implementation of an ontology, the initial use of the wiki-system at the beginning of the product development process was improved by other preliminary work, e.g. implementing the basic structure, including some basic knowledge, or integrating the semantic MediaWiki into the company's corporate design. The final preliminary activity was aimed at an instruction of the semantic MediaWiki's structure and its main functionalities (see figure 6).

Over the course of the project the wiki-system was mainly used by the three mechanical engineers, who formed the core of the team. All these three engineers were responsible for the content of the wiki, whereas only one of them administrated the system.

6. Discussion and Conclusion

The clear separations of objectives and objects helped to develop the thing right as well as the right thing. The explicit linking structure allowed a comfortable and efficient change of perspective between the objectives and the progress of the project respectively the latest version of the system of objects. Furthermore, the cross-linking of the hierarchic, functional structure helped to avoid interface problems between the systems and their corresponding sub-/super-systems (see figure 6).

By the realization of the knowledge base and the project documentation (in this case a product development process) within one (wiki-) system, a smooth and easy exchange of information respectively knowledge was provided. This approach provides a generic and transferable structure for knowledge created and needed in product engineering. The model used for structuring knowledge is flexible enough to handle also unpredictable or unplanned aspects of processes.

Overall, four main problems were identified during the project:

1. The acceptance/comprehension of iPeM as a mental model and its containing aspects such as system of objectives/objects and the activity matrix.
2. The partly existing ambiguousness
    a. of the system of objectives and the system of objects (e.g. a relevant insight that was generated in the system of objects becomes part of the system of objectives),
    b. of the question, when and under which circumstances a system needs to get divided into several sub-systems.
    c. under which circumstances the activity matrix is too coarsely grained and needs to be refined further.

3. The answer to the question which percentage of an engineer's daily labor time is needed and available for an adequate documentation respectively transformation of project information into the knowledge base.
4. The organizational culture needs to accept and live for the 'sharing knowledge' approach.

The first two problems can be dealt by this approach in combination with the intelligence and the incentive of its users. The flexibility of this approach, which brings up those two problems, seems to be the only way to keep such a system simple and consistent. Therefore, the intelligence and incentive of the system's user is an imperative part of the solution.

The second two problems need to be answered by the approach as well as the organization in which it shall be implemented. Only with the help of a solid strategy of introduction in combination with the organization's will to advocate and claim this approach, success becomes possible.

Another aspect which needs to be clarified through further research is the scalability of the approach.

Figure 6. Project and corresponding systems

As iPeM is a meta-model for product development processes and its elements can be used to model different processes with a set of identical elements, the scalability to multi-project and larger project environments should be possible without serious problems. For structuring very large projects, the activities of product engineering can be further refined with sub-activities for a particular organizational environment. For larger knowledge bases the benefits of additional structuring of the contents through an ontology become even more useful for the users when retrieving knowledge. A prerequisite for good usability of a large knowledge base is a carefully designed and validated ontology. Scaling this approach to larger and multi-site projects requires similar levels of familiarity with the iPeM model from the people involved and therefore requires systematic training and education on the job.

7. Summary and Future Work

Efficient knowledge management is becoming increasingly important for product development and Wikis have become popular as support tools for knowledge management, also in product development. Semantic wikis allow to structure wiki contents with machine-readable metadata which can be used to retrieve wiki content in a more specific and user-friendly way. Semantic wikis try to combine the advantages of social software with those of semantic software systems. Models of product development can provide a suitable ontology for structuring content of a semantic wiki in product development. An approach was presented for structuring semantic wiki contents with the integrated product engineering model iPeM. A pilot implementation of this approach in an industrial development environment was presented using SemanticMediaWiki. The pilot implementation showed the principle feasibility but also that further refinements of the approach and changes in processes, organizational culture and education could improve the benefits of the approach. Fields for future work are for example studies with larger groups and longer duration and possible ways to combine the advantages of "bottom up" and "top down" strategies for structuring wiki content.

References

Albers A., "Five Hypotheses about Engineering Processes and their Consequences" in Tools and Methods of Competitive Engineering TMCE 2010 Ancona, Italien, 2010.(accepted for publication)

Browning T. R. and Ramaseh R. V., "A Survey of Activity Network-Based Process Models for Managing Product Development Projects", Production and Operations Management, 16, 2007, pp. 217-240.

Cunningham W. and Leuf B., "The Wiki Way. Quick Collaboration on the Web", Addison-Wesley Longman, Amsterdam, 2001.

Hinkelmann K. and Wache H., "Proceedings of WM2009-CWeb2.0 5th Conference on Professional Knowledge Management - Corporate Web 2.0". K. Hinkelmann and H. Wache, Solothurn, Switzerland,2009, p.^pp.

Klein R., "A Knowledge Level Theory of Design and Engineering ", in Globalization of manufacturing in the digital communications era of the 21st century: innovation, agility, and the virtual enterprise, Kluwer Academic Publishers, Dordrecht, 1998.

Schütt P., "Die dritte Generation des Wissensmanagements. (the third generation of knowledge management)", KM-Journal 11, 2003, pp. 1-7.

Sim S. K. and Duffy A. H. B., "Towards an ontology of generic engineering design activities", Research in Engineering Design, 14, 2003, pp. 200-223.

Völkl M., Krötzsch M., Vrandecic D., Haller H., and Studer R., "Semantic Wikipedia" in IW3C2 Edinburgh, Scotland, 2006.

Walthall C., Sauter C., Deigendesch T., Albers A., and Ramani K., "Survey Of Wikis As Design Support Tool" in International Conference on Engineering Design, ICED '09 Stanford, CA, USA, 2009.

Wynn D. C., "Model-based approaches to support process improvement in complex product development". Ph.D. thesis, University of Cambridge, 2007.

Prof. Dr.-Ing. Dr. h.c. Albert Albers
Karlsruhe Institute of Technology (KIT), Institute for Product Development (IPEK)
Kaiserstr. 10, D-76131 Karlsruhe
Telephone: +49 721 608-2371
Email: albers@ipek.uni-karlsruhe.de
URL: http://www.ipek.uni-karlsruhe.de