Technical University of Denmark

DTU

An agile documentation system for highly engineered, complex product configuration systems

Shafiee, Sara; Hvam, Lars; Kristjansdottir, Katrin

Published in:
Proceedings of the 22nd EurOMA Conference

Publication date:
2015

Document Version
Peer reviewed version

An agile documentation system for highly engineered, complex product configuration systems

Sara Shafiee, sashaf@dtu.dk, sara@topsoe.dk
Industrial PhD Student, Department of Management Engineering, Technical University of Denmark, 2800 Kgs.Lyngby, Denmark

Lars Hvam
Professor, Centre for Product Modelling (CPM), Department of Management Engineering, Technical University of Denmark, 2800 Kgs.Lyngby, Denmark

Katrin Kristjansdottir
PhD Student, Department of Management Engineering, Technical University of Denmark, 2800 Kgs.Lyngby, Denmark

Abstract

When modelling and maintaining a product configuration system (PCS) there is a need for a complete and updated documentation of the system. This research work focuses on a framework for documenting PCSs based on structures, data and constraints already implemented in the PCS. Looking at previously suggested PCS documentation, systems focus on creating the documentation separated from PCS. Documenting the PCS separately request extra time and resources. We suggest building the PCS models and then extracting the structure, data and constraints for documentation from PCS. This makes the task of documentation easier and less time consuming.

Keywords: Product configuration systems (PCS), Documentation System

Introduction

A PCS can be defined as a product-oriented expert system, which allows users to specify products by selecting components and properties under the restriction of valid combinations (Mittal & Frayman, 1989; Franke, 1998). Due to the complexities resulting from the large number of customized products, an increased attention has been paid to the order fulfilment process (Zhang, et al., 2010). Studies have revealed that in the companies using PCS and not facilitating with documentation system they are not able to develop their configurators and they have had to abandon or rebuild their configurators (Haug, et al., 2009). When modelling product families for product configuration, Duffy suggests four basic representations of the products as shown in Figure 1 below:

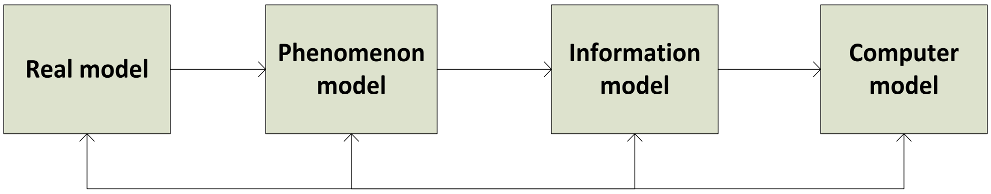


Figure 1 - Moving from the real world to an IT system (Duffy & Andreasen, 1995)

The information model is used to represent the formal IT representation of the model and is usually based on UML notation (Kruchten, 1998); but the information model could not be used for communication with domain experts. The phenomenon model is created only for the communication with domain experts and other employees that do not have IT expertise. The phenomenon model visualizes the structure of product families and demonstrates the model available in the PCS (Hvam, et al., 2008).

Problem statement

Updating both a phenomenon model of the products and the IT model in PCS requires significant time and resources. Therefore, avoiding duplicated information is another challenge for developing this documentation system framework (Selic, 2009). Therefore we need to avoid redundancy and extra works by using an automated documentation system. For this we will investigate if it is possible to generate the documentation (the phenomenon model) based on structures, data and constraints implemented in the PCS. In more detail we would like to investigate the requirements for documenting the chosen phenomenon model in the literature, the adaptation of the phenomenon models to the available structures in PCS, possible ways for extracting the phenomenon model from PCS, and evaluate the suggested framework in several projects.

Research method

In accordance with the overall objective, the research has been structured into two phases. The first one focused on the development of the framework and the second one focused on testing.

Phase 1 – Framework development

For this study the phenomenon model is made based on the available techniques of Product Variant Master (PVM) and Class, Responsibilities, and Collaboration (CRC) cards as the explicit phenomenon models (Hvam et al., 2008). This is due to the experiences of the research team and the company and methodologies and requirements for documentation in software projects (Briand, 2003). Gathering the list of requirements for our explicit phenomenon model based on the previous literature and current requirements; and the next phase is the investigation how to use commercial PCS for generating phenomenon models.

The framework was developed by researchers with an applied research background in modelling products, product architecture and product configuration, software development, combining traditional domains of mechanical engineering with product configuration and software development.

Phase 2 – Testing the framework

The test phase aimed to prove the possibility of data extraction from PCS and efficiency in this type of documentation. The purpose of the test was not to determine if the framework would work in all situations, but only to undertake a first empirical test and obtain input for the further development of the procedure. The role of the researchers was to define which data to extract from the PCS and setup the structure of the documentation system in the file sharing system.

Literature study

The keywords used in the literature study include: “modelling techniques”, “mass customization”, “product configuration”, “IT systems”, “UML”, “configuration systems structure”, “knowledge management”, and “content management systems”. Additionally, the list of references of each article is used to identify related bibliography, as well as the names of the researchers in the recognized research groups within this field.

Product Modelling for PCS

The Unified Modelling Language (UML) is a general-purpose visual modelling language that is designed to specify, visualize, construct and document a software system (Mekhilef, et al., 2003) and to encourage designers to formalize their implicit knowledge, to make the knowledge extraction easier. Figure 2 depicts a high-level UML state chart for models (Ambler, 2002). In this state chart a temporary model is created to communicate and make it permanent when it is completely clear for everyone; it is creating value and everybody willing to document it.

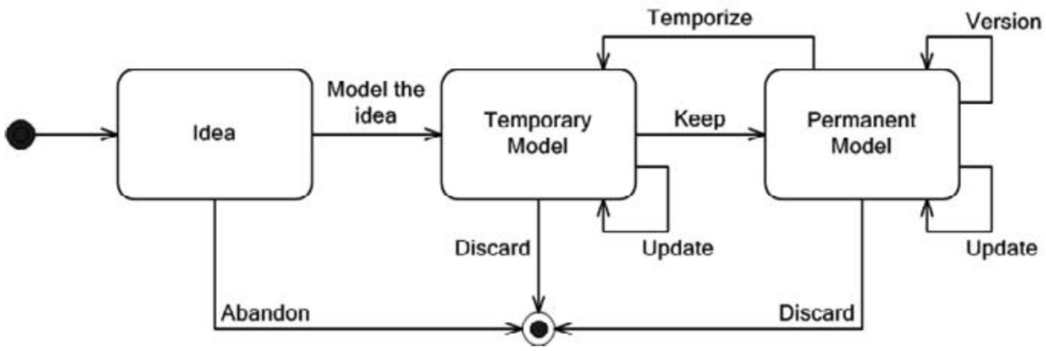


Figure 2 – A UML state chart that depicts the lifecycle of an agile model **(Ambler, 2002)**

There are different modelling techniques for the configuration projects available in the previous researches (Haug, 2007; Aldanondo, et al., 2000; Chao & Chen, 2001; Margo & Torasso, 2003; Tseng, et al., 2005; Jinsong, et al., 2005; Yang, et al., 2009). The Product Variant Master (PVM), class diagram and the CRC Cards which are extracting from UML (Unified modelling language) has been used for this project.

Product Variant Master

The PVM presented by Hvam (Hvam, et al., 2008) aims at representing product knowledge in a structured format; it shares high similarity to the Product Family Master Plan (PFMP) that is used developing ”product families”, based on the architecture presented by Harlou, (Harlou, 2006). For visualizing and facilitating product knowledge, the PVM has proven successful in several cases.

CRC Cards

The CRC cards were proposed by Cunningham (Beck & Cunningham, 1989). Originally the CRC cards were developed as a way of teaching object oriented thinking. Hvam et al. have later presented several revised definitions of the CRC cards to be used in product configuration projects (Hvam, 2006).

Commercial configuration systems structure

As this research framework is limited to pre-existing configuration framework, we have to define the general structure from commercial PCSs. Friedrich and et al. describe the modern configuration framework as the systems which have to provide mechanisms that abstract from the underlying technical representation as far as possible in the modelling phase (Friedrich, et al., 2014). Felfernig et al. explain a UML based method for configuration development as a graphical notation (Felfernig, et al., 2000). Hvam et al. discuss a list of requirements for selecting a PCS and they are considering using tables and tree structures in the configurator formation (Hvam, et al., 2008).

Documentation Requirement for PCS

Having a proper requirements specification is vital for documentation of PCSs (Hvam, et al., 2005). In general the requirement for a structured documentation of the configurator models is increasing along with the number of completed models and users. Majority of requirements are extracted according to the expectation from an IT documentation system and UML framework and mentioned as general requirements from the PCSs. Some of the documentation system requirements that needs to be considered for the documentation system are listed in Table 1.

Table 1 – requirements for PCS documentation

Requirements	**Description**	**General/ Specific**
Easy to use and easy to construct	The product exports must know this tool as user friendly software to use and it is important that documentation can be easily done (Elgh, 2011); Hvam, et al., 2005).	General
Enter the changes in one place	Avoiding making errors in updating the configurator and system, the updates should apply only in configurator and the other system should receive the changes automatically (Ambler, 2002).	General
Version comparing	Allow historical comparisons between different versions of the documentation (Rask, 1998).	General
Access limitation	There is not possible for the users to make changes except the person who is responsible (to keep the configurator update with the system)	General
Access management	A database over user groups allowing them different rights to access and edit the model (Hansen, 2010).	General
Integration	Integration with other documentation systems in the company and Existing systems (e.g. ERP) will inform the responsible by an email each time that there is a change in the documents (version or name) (Hvam, et al., 2005).	Specific
Model history overview	Management of changes made to the model and the ability to revert to a previous version (Hvam, et al., 2005).	Specific

Hyperlinks	Allows the user to link to external references such as drawings and documentation (Hvam, et al., 2005).	Specific
Model tree structure	New primary view with an unchangeable tree structure (Haug, 2010).	Specific
Database Management System (DBMS) access	A DBMS module handles all communication to internal as well as external databases by using standardized communication protocols. This ensures adaptability to various database systems (Hvam, et al., 2005)	General

Content Management Systems

A content management system (CMS) is a computer application that allows publishing, editing and modifying the content as well as maintenance (Rockley, et al., 2003). Most CMS include Web-based publishing, format management, revision control (version control), indexing, search, and retrieval. A CMS usually improves communication while reducing the costs (Powel & Gill, 2003).

Suggested framework for the documentation system

In the suggested framework for the documentation of PCSs the documentation of the product model is generated and maintained from the PCS. The main idea of this framework is extracted from literature part which is UML based lifecycle (Ambler, 2002); and it visualizes the creation of a temporary model as PVMs and CRC cards at first and then makes it permanent when it is completely clear. As shown in Figure 3, in the suggested framework the initial phenomenon model is made based on the modelling techniques selected (in this case using PVM and CRC-cards). When the initial version of the PCS is established, the PVM, class diagram and CRC-cards are generated from the PCS in an adapted version. The initial model is archived, and from this point forward all future versions of the PCS are documented based on PVM's and CRC-cards generated from the PCS. For future versions of the PCS only these models are being updated and used.

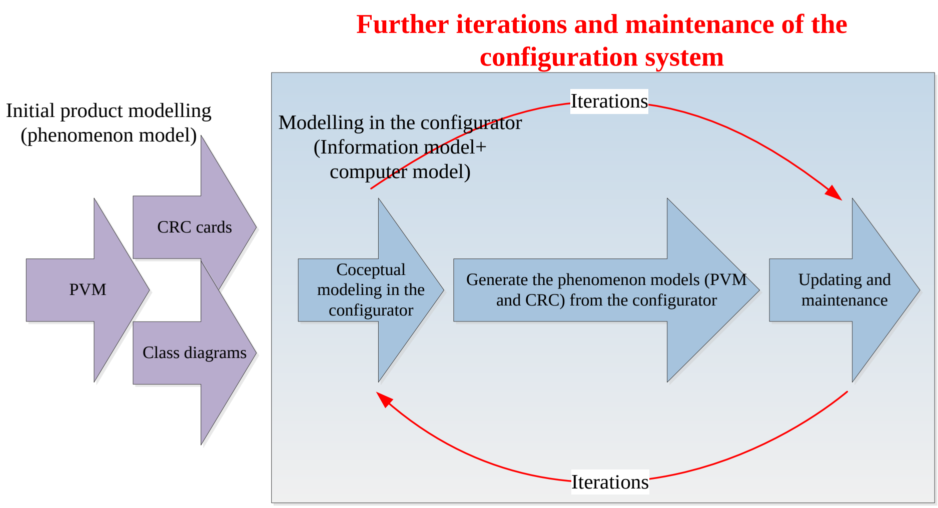


Figure 3 – the framework for PCS documentation

It is expected that the configurator at the company includes the PVM and also CRC card structure in a similar way which is usable for the documentation system as in the Figure

4. These expected structures developed according to observation and evaluation of different successful main commercial configurators on the market. The product components are in tree structure format and the constraints and rules are shown in the IT language in the configurator. The effort is to transfer and translate all this information into the documentation system and generate the PVMs and CRC cards.

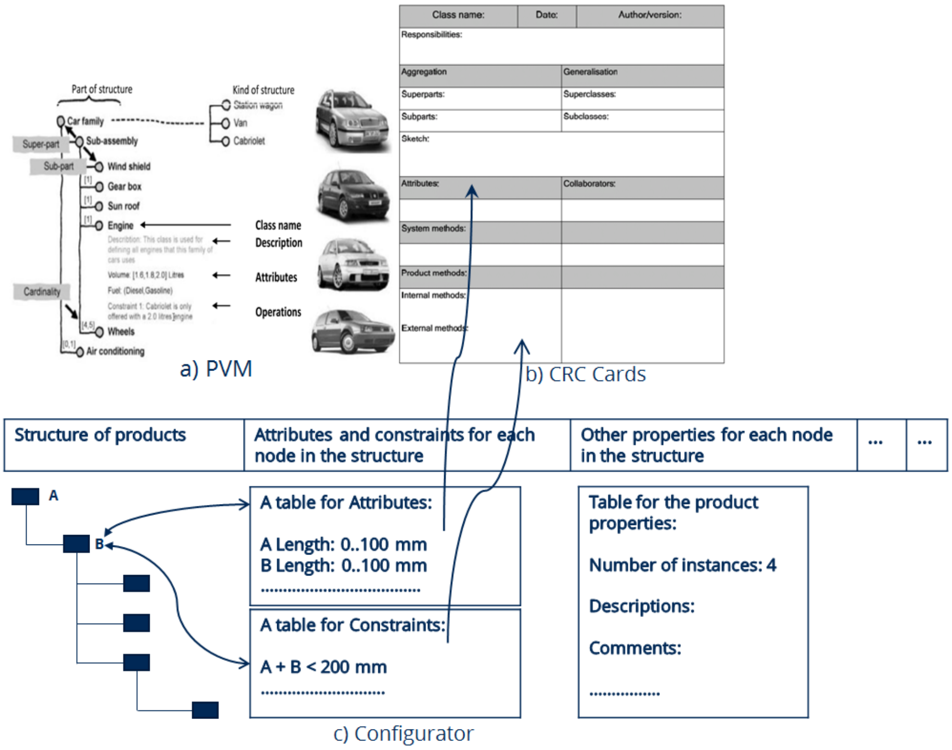


Figure 4 – PVM and CRC Cards structure (a,b) (Hvam, et al., 2008). Documentation from commercial PCS in the form of PVMs and CRC cards (c)

There are PVM and CRC structures inside the available PCS according to the expectation and discussion in the literature (Figure 4). It is not possible to see constraints and variants inside the tree structure but they are all available in the CRC cards on the right hand side (Tiihonen, et al., 2013). There are drawbacks in PCS structure such as: lack of explanations, unavailability of figures, and no product hierarchy; but it is still very efficient to use PCS as a database (Friedrich, et al., 2014). Generating the information from configurator, it is possible to have the figures which are still coming from configurator. It is possible to manage and add all the explanations and additional information to the configurator (not as the PCS information but just for the documentation system use) and transfer it to the documentation system.

Case study

The proposed documentation framework has been applied in a real context to assess its functionality. The case company, Haldor Topsoe A/S, is an international company specialized in the production of heterogeneous catalysts and in the design of process plants based on catalytic processes. There are three different configuration projects as the testing units on production stage inside the company which has been using for testing the mentioned documentation system.

Stakeholders' Requirements

Table 2 illustrates the list of specific requirements from the main stakeholders at the company. The system responds to almost all of the requests from the stakeholders.

Table 2 – Stakeholders' requirements

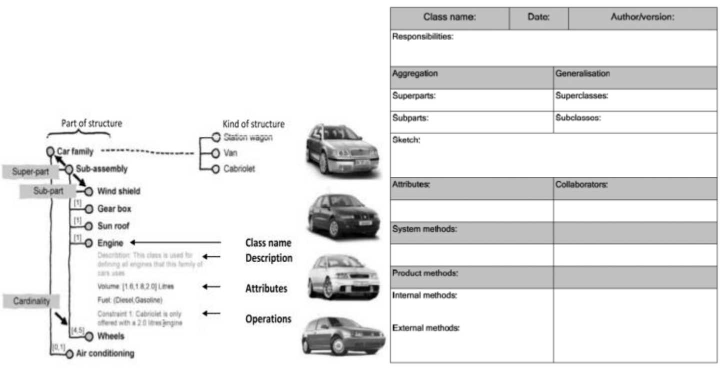
Requirements	Stakeholders	General/Specific
Agile documentation with the less possible time and resource to update and maintain	Steering committee	General
Avoid redundant requirements	Steering committee	General
Having a proper communication platform for the domain experts	Domain experts	Specific
Model of the products	Domain experts	Specific
A web based user interface	Domain experts and configuration team	General
Model structure and CRC Cards	Domain experts and configuration team	Specific
To ease understanding of the model	Configuration team	Specific
Add explanations to the model (tags)	Configuration team	Specific
Using Hyperlinks to other data bases	Domain experts	General
To be able to hide and show the desired information	Domain experts	Specific

Documentation Tool for Configuration System

The steps for setting up Microsoft SharePoint as a documentation system for the configuration projects are as following:

- SharePoint Team Site will be created, user administration will be managed in SharePoint standard security based on user roles,
- Request Tracking List will be implemented of the Discussion Type in SharePoint to allow users to communicate about the requests directly in the list,
- The entire Configurator Documentation System will be implemented in client side using JavaScript technology.

As illustrated in Figure 5 and according to the discussions, for extracting the data from the PCS and using it as database, the structure should be at the level of our expectation and compatible with our modelling techniques. Figure 6 demonstrates the documentation system at the company which is receiving all the information from the configurator and through XML file. There is the similarity between the three figures and all showing the tree PVM structure of the product as well as the information included in the CRC cards.



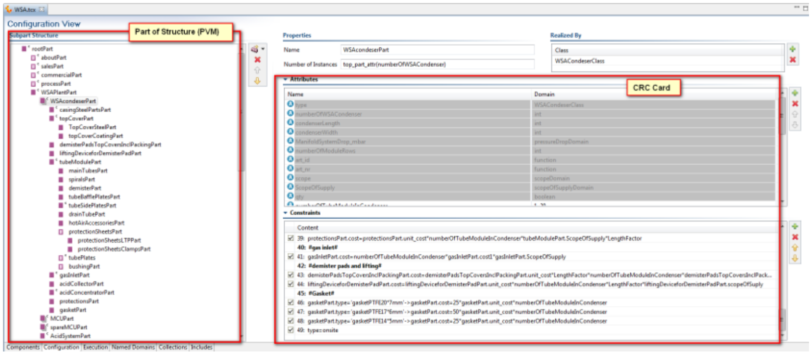


Figure 5 – PVM and CRC Cards framework in the configurator (Tacton)

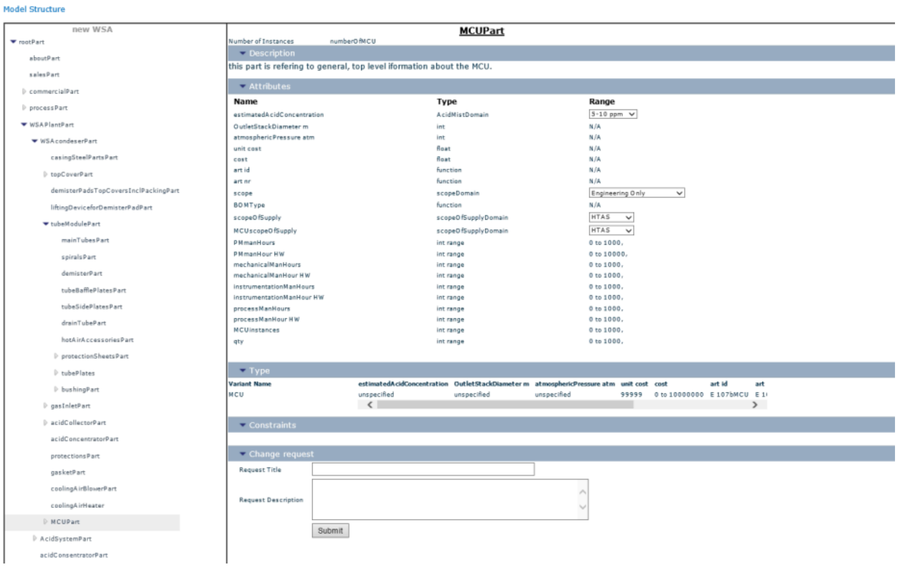


Figure 6 – PVM and CRC inside the documentation system

Discussions

The system evaluation has two perspectives which have to be considered: IT functional requirements (Kruchten, 1998) and usability of the system (Nielsen, 1992). The collected data was based on observations and interviews with different stakeholders at the company from the domain experts to steering committee. We continued with data gathering during six months using systematic interviews with different stakeholders that have different expectations for the system. The people who participated in the data collection were domain experts of the tested projects, the configuration team and the steering committee of the projects.

Table 3 illustrates the interview questions and the results during the past six months at the company.

Table 3 – interviews results

Questions	**Configuration team**	**Domain experts**	**Steering committee and top managers**
How much time is needed for learning the system?	Answers in the range of 0.5 to 1 hour	Between 2 to 3 hours	Approximately 1 to 3 hours

How much time is saved using the system compared with the old routine methods?	Between 50 to 60% of the total time compared with the excel sheets	Between 70 to 80 % compared to previous process	30 to 40 % due to the understanding
How much are errors reduced due to the use of the system?	Approximately 30 to 40 %	Around 20 %	Around 20 %
What is the level of satisfaction and acceptance between the users?	Very high	Very high	high

Conclusion and further work

Documenting a product model is a very time consuming process as the product models grow and get more complex (Comptont & Jansen, 1990). As mentioned in the literature in most cases the documentation generated has not survived after the implementation. The reason was that nobody wants to update the configurator and a documentation system at the same time because it doubles the workload in the project and therefore the suggested framework eliminates redundancy and additional activities and therefore saves time and resources significantly. One of the outstanding points of the PCSs is that we are building a comprehensive documentation tool which could be used as the products' knowledge database. This framework might not be applicable for all PCSs as it needs to have the initial requirements in the PCS structure. There are some suggestions for the future research:

- Considering different modelling techniques for the phenomenon models
- Further test with different PCS (projects, companies, softwares).
- Having dialogues with the venders of configurators for prioritizing and assessing the applicability of the suggested documentation generation facilities in the PCSs.

References

Aldanondo, M., Rouge, S. & Veron, M., (2000), " Expert Configurator for Concurrent Engineering: Cameleon Software and Model", *Journal of Intelligent Manufacturing,* 11(2), pp. 127-134.

Ambler, S., (2002), *Agile Modeling :Effective Practices for eXtreme Programming and the Unified Process.* New York: Wiley Computer Publishing (John Wiley & Sons, Inc.).

Beck, K. & Cunningham, W., (1989), "A Laboratory for Teaching Object-Oriented Thinking, Object-Oriented Programming Systems.", *Languages and Applications,* pp. 1-6.

Briand, L. C., (2003)," Software Documentation: How Much Is Enough?", 7th European Conference on Software Maintenance and Reengineering, CSMR.

Chao, P. & Chen, T., (2001), "Analysis if Assembly Through Product Configuration", *Computers in Industry,* 44(2), pp. 189-203.

Comptont, P. & Jansen, R., (1990) , "A philosophical basis for knowledge acquisition", *Knowledge Acquisition,* Volume 2, pp. 241-257.

Duffy, A. H. B. & Andreasen, M. M., (1995), " Enhancing the evolution of design science", Proceedings of ICED.

Elgh, F., (2011), "Modelling and Management of Product Knowledge in an Engineer to Order Business Model", Lyngby/Copenhagen, Denmark, Proceedings of the 18th International Conference on Engineering Design (ICED 11).

Felfernig, A., Friedrich, G. E. & Jannach, D., (2000), " UML as domain specific language for theconstruction of knowledge-based configuration systems", *International Journal of Software Engineering and Knowledge Engineering,* 10(4), p. 449–469.
Franke, D. W., (1998), "Configuration research and commercial solutions", *AI EDAM,* 12(04), pp. 295-300.
Friedrich, G., Jannach, D., Stumptner, M. & Zanker, M., (2014), " Knowledge-based Configuration: From Research to Business Cases", in *Knowledge Engineering for Configuration Systems,* Morgan Kaufman, pp. 139-155.
Hansen, T., (2010), "Supporting the CPM Procedure", *Master Thesis*, Lyngby, Technical University of Denmark.
Harlou, D., (2006), "Developing Product Families Based on Architectures Contribution to a Theory of Product Families*", PhD thesis*, Lyngby, Technical University of Denmark.
Haug, A., (2007), "Representation of Industrial Knowledge as a basis for Developing and Maintaining Product Configuratoes*", PhD Thesis*, Lyngby, Technical University of Denmark.
Haug, A., (2010), "A software system to support the development and maintenance of complex product configurators", *The International Journal of Advanced Manufacturing Technology,* 49(1-4), pp. 393-406.
Haug, A., Hvam, L. & Mortensen, N. H., (2009), "Implementation of conceptual product models into configurators: From months to minutes", in 5th World Conference on Mass Customization and Personalization..
Hvam, L., (2006), "Mass Customization and Configuration of Process Plants", *International Journal of Mass Customization,* 1(4), pp. 445-462.
Hvam, L., Christensen, S. P., Jensen, K. L. & Riis, J., (2005), "Development and maintenance of product configuration systems: requirements for a documentation tool". *International Journal of Industrial Engineering-Theory Applications and Practice,* 12(1), pp. 79-88.
Hvam, L., Mortensen, N. H. & Riis, J., (2008), *Product Customization,* Springer.
Jinsong, Z., Qifu, W., Li, W. & Yifang, Z., (2005), "Configuration Oriented Product Modelling and Knowledge Management for Made-to-order Manufacturing Enterprises", *International Journal of Advanced Manufacturing Technoligy,* 25(1-2), pp. 41-52.
Kruchten, P., (1998), *The Rational Unified Process: An Introduction,* New York: Addison-Wesley.
Margo, D. & Torasso, P., (2003), "Decomposition Strategies for Configuration Problems", *Artificial Intelligence for Engineering Design Analysis and Manufacturing,* 17(1), pp. 51-73.
Mekhilef, M., Bourey, J. & Bigand, M., (2003), "An UML Modelling of an Architecture for Knowledge Documentation", Stockholm, the 14th International Conference on Engineering Design.
Mittal, S. & Frayman, F., (1989), "Towards a Generic Model of Configuraton Tasks". *IJCAI,* Volume 89, pp. 1395-1401.
Nielsen, J., (1992), "The usability engineering life cycle", *Computer,* 25(3), pp. 12-22.
Powel, W. & Gill, C., (2003), "WEb Content Management Systems in Higher Education", *Educause Quarterly,* Issue 2, pp. 43-50.
Rask, I., (1998), "Report 1. Rule-Based Product Development*", Industrial Research and Development Corporation,* Molndal.
Rockley, A., Kostur, P. & Manning, S., (2003), *Managing Enterprise Content: A Unified Content Strategy,* New Riders.
Selic, B., (2009), "Agile Documentation, Anyone?", *IEEE SOFTWARE,* 26(6), pp. 11-12.
Tiihonen, J., Heiskala, M., Anderson, A. & Soininen, T., (2013), "WeCoTin–A practical logic-based sales configurator", *AI Communications,* 26(1), pp. 99-131.
Tseng, H., Chang, C. & Chang, S., (2005), "Applying Case-Based Reasoning for Product Configuration in Mass Customization Environment", *Expert Systems with Applications,* 29(4), pp. 913-925.
Yang, D., Miao, R., Wu, H. & Zhou, Z., (2009), "Product Configuration Knowledge Modelling Using Ontology Web Language" *Expert System with Applications,* Volume 36, pp. 4399-4411.
Zhang, L. L., Lee, C. K. & Xu, Q., (2010), "Towards product customization: An integrated order fulfillment system", *Computers in Industry,* 61(3), pp. 213-222.