SUPPORT MANAGEMENT OF PRODUCT FAMILIES AND THE CORRESPONDING AUTOMATION SYSTEMS – A METHOD TO CAPTURE AND SHARE DESIGN RATIONALE

Poorkiany, Morteza; Johansson, Joel; Elgh, Fredrik
Jönköping University, Sweden

Abstract

The ability to innovate and launch customized products that are well matched to customer demands is a competitive factor for many manufacturing companies. Development of highly customized products requires following an engineer-to-order business process to tailor the products according to customers' specifications, which brings more value to the customer and profit to the company. Using design automation systems to automate repetitive and time consuming design tasks enables the manufacturers to perform custom engineering in minimum time. To manage and maintain a product family and the corresponding automation systems, updating the design knowledge is required. Use of design rationale will normally become a necessity to allow a better understanding of the knowledge. Consequently, there is a need of principles and methods to enable capture and effectively share the design rationale. In this paper a method for capturing and sharing design rationale is presented. The results are evaluated in a case company which is a supplier of tooling for manufacturing industry.

Keywords: Design automation system, Computer Aided Design (CAD), Design engineering, Decision making, Design rationale

Contact:
Morteza Poorkiany
Jönköping University
Product Development
Sweden
morteza.poorkiany@ju.se

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 21st International Conference on Engineering Design (ICED17), Vol. 6: Design Information and Knowledge, Vancouver, Canada, 21.-25.08.2017.

1 INTRODUCTION

As the markets' needs change rapidly, developing a variety of products that meet customer's diverse needs is a challenge for many manufacturing companies. Design of a new product variant involves a large amount of repetitive and time consuming tasks but also information handling activities that are sometimes beyond human capabilities. Such work that does not rely so much on creativity can be carried out more efficiently by applying design automation systems. The benefits of design automation systems implemented in different areas vary concerning the objectives of the systems, but are mainly connected to shortening lead time, improving product performance, and ultimately decreasing cost (Johansson, 2008). Further, design automation systems often facilitate the documentation and maintenance of corporate knowledge, and enable the designers to focus their work on solving problems that need skill, experience, and creativity (Elgh and Cederfeldt, 2005).

Regardless of the scope and complexity of the design automation system, challenges often arise when a system is to be implemented as well as in its management. Implementation concerns the aspects relating to the organization, IT infrastructure, process, methods and tools to ensure proper and effective use. Management, on the other hand, concerns adaptation of rules and models to changes (for example, new product requirements, new standards or legislations, or changes in technology), updating frameworks and documentation, version control and traceability.

A vast amount of information and knowledge is used or produced throughout the design of a product. Successfully capturing the design knowledge and effectively representing it is essential to improve the design process (Poorkiany et al., 2016a). The generation of feasible design alternatives needs the effective utilization and application of this information and knowledge (Hicks et al., 2002). The need for knowledge support in order to effectively enable capture, representation, retrieval and reuse of product knowledge is emphasized in literature (Ouertani et al., 2011; Szykman et al., 2001; Stokes, 2001). In order to enable reuse, a major problem is to identify which knowledge and information to capture, and once identified, what extent of capture is required to make the information and knowledge truly useful (Hicks et al., 2002).

Based on the content, design knowledge can be categorized into two groups (Elgh, 2011). One is design definition describing the results of the design, without any explanation concerning the reasons and argumentations behind the design. Such information can more often answer the "what" question. A CAD model, a design table, or a test report are some examples of design definition which are mostly based upon insights, experience, trade-offs, calculation, simulations, etc.

The other one is design rationale explaining the purpose and reasons behind the design in more details. Design rationale provides a better understanding for design definition and often aims at explaining the artefact in the way it is designed answering the "why" question. For instance, *why a CAD model looks like it does?*

The potential value of design rationale in development and maintenance of a product family is significant, however, the systems developed to record, document, and manage design rationale are not widespread used in industry. Effective capturing of rationale is one central obstacle when developing a design rationale system (Dutoit et al., 2006). Yet, after many years' research on design rationale, many companies still struggle in capturing their rationales during decision making. Answer to questions such as *what information and knowledge should be captured? when should the design rationale be captured? who should capture the design rationale? and how to share the design rationale for other engineers?* is still unclear for many manufacturing companies. This has been the focus of the research presented in this paper. The findings of the research are examined in an industrial partner which is a global manufacturer for machinery tooling.

2 THEORETICAL FRAMEWORK

Design rationale is defined as reasons behind a design decision, the justification for it, the other alternatives considered, the argumentation that led to the decision and the trade-offs evaluated (Lee, 1997). The potential uses of design rationale are discussed in (Burge, 2005). Four of them which are the focus in this study are listed below:

- Design verification: to verify that design meets the requirements.

- Design evaluation: to evaluate design and design choices relative to one another to detect inconsistencies.
- Design reuse: to indicate which part of design can be reused, or in some cases suggest where should be modified to meet new requirements.
- Design maintenance: to determine where changes need to be performed to modify the design. Design maintenance includes adaptive maintenance which is the revising and improving the usability of the product, and enhancive maintenance which is extending the functionality of the product.

To rationalize the design, it is important to understand what type of information and knowledge is used. This has been studied in (Mayer et al., 1992) where the investigations into the nature of rationale shows that design rationale includes arguments based upon the following: the philosophy of design including process description of intended system operation, design limitations expressed as range restrictions or environmental factors, factors considered in trade-off decisions such as requirements matching, budget or timing constraints, technology available to implement and test the design, design goals, and even personal evaluation factors or constraints. Further it is suggested that the type of rationale that should be captured can be discovered by experiment; observing the engineers trying to understand the design. The observation can be completed by structure interviews asking about a number of interested topics. The knowledge required to answer these questions is requirements for design rationale capture method. The interested topics are, for instance, product structure/model, purpose relative to the intended behaviour of the artefacts, supportability of beliefs used as rationale for design decisions, the design process (how it was planned and carried out?), and design behaviour or failure (the what if ...? questions).
Advances in collaborative document repositories enable the organizations to provide tools to support the capture and retrieval of design decisions and their rationales. A knowledge-based tool is introduced in (Hooey et al., 2012) to capture design rationale during making critical decisions in T-NASA (Taxiway Navigation and Situation Awareness) system in order to support design verification and reuse. Three categories of design knowledge are included in the tool: design driver referring to high-level design goals and assumptions, design requirement referring to functional needs and performance requirements, and design elements referring to the analysed solution which meets the requirements. The proposed tool allows the designers to capture design drivers and requirements, and map them to design elements. Each element can be related to one or more requirements and each requirement can be satisfied by one or more elements. By adaptation of the QFD (Quality Function Deployment) matrix, rationale object explains the connections between the requirements and elements and why a requirement is satisfied by a particular element.
Enabling traceability of design knowledge was the aim of the study presented in (Ouertani et al., 2011) to support the decision making process during engineering change management and also to enhance the sharing and use of design knowledge during the development process. A product knowledge model was proposed to address: what product knowledge is created or represented? who are the actors playing roles in creating or using the knowledge? where is the product knowledge created? why was certain knowledge created or modified? when was the knowledge created or modified? and how is the knowledge being created or modified? Further, a design rationale model is explained that in addition to the constructs mentioned above, also represents the variability (likelihood of changing design requirements), sensitivity (the design risk when a change occurs) and completeness (requiring other knowledge to achieve to task).
Using argumentation-based models is one way to represent design rationale. The models usually use nodes representing a component, and link representing a relationship between components. IBIS (issue-based information system) (Kunz and Rittel, 1970) is an example of such models which represents a debate of controversial questions that arise in design (Dutoit et al., 2006). An issue is recorded and it is answered by options (alternatives). The options and their pros and cons are evaluated based on arguments. IBIS includes multilevel structure of arguments for and against alternatives which provides a comprehensive discussion about design issue and generation of design alternatives. During last years, the IBIS method has been reviewed and developed further by the research community due to its simplicity. More details about the IBIS method and the systems developed based on that can be found in (Bracewell et al., 2009).

3 CASE COMPANY, RESEARCH SCOPE AND METHODOLOGY

The industrial partner in this research is a global supplier of tooling for manufacturing industry and follows an engineer-to-order business process. The development process at the company starts with identifying customer needs and marketing research, and ends with developing application systems to automatically generate new design variants. The process is divided into three sub-processes: *product development*, *engineering design* and *design programming*.

During the product development process, each product family gets its own defined standard instances like any other product on the market but the process also includes the establishment of design rules to enable generation of new product variants within a so called “product space”. Parametric design modules are developed and the product space is governed by a set of design modules and the way these modules are combined. This enables the company to have standard products as well as generate individualized variants within the product space according to the customer demands. The customers sometimes demand highly specified products which are beyond the product space. Such special orders are to be processed individually in a separate design department called “Special Design” by a design team called “Special Designer”.

Automation of the design process was started in the early 1980’s and is achieved by developing and implementing advanced rule-based programs which at run time select, modify, and combine design modules based on customer specified input parameters. The output of the automation system is product related information such as 3D models, drawings, selected production unit, NC codes and measurement instruction.

In a research (Elgh and Poorkiany, 2012) which was performed previously in the company, the concept of Product Family Description (PFD) was introduced. PFD is a document with the aim of structuring the design information and knowledge (e.g. design rules and modules) that is to be used for communication purposes between the design and design programmer, and also to provide an overview and understanding of the product family to support reuse and maintenance. The PFD mainly contains the final results of the design without any explanation about the design history and the reasons behind design decisions which makes it almost unhelpful for engineers in product maintenance and special design. Since these engineers usually perform changes in design, expand the product space or design a new variant beyond the product space, they are very interested in to learn about design rationales (answers to the question why, for example, *why does the product looks like this?* or *what happens if I change the dimension?*).

To conduct scientific research, a research method is required to explain and guide the process of selection and application of suitable methods and approaches. While a research method assists in making plans to implement and proceed the research, it is necessary to consider the chances to achieve valid results. Further, it is necessary to practically deploy and evaluate the results. The study presented in this paper is a part of a three-year research project in a partner industry. The research follows Design Research Methodology (DRM) (Blessing and Chakrabarti, 2009). The outcomes of the first and second stages of the DRM which are mainly clarifying the research goal and focus, and understanding the current situations are published in (Poorkiany et al., 2016b). Three workshops and three project meetings were carried out in the case company. In addition, a set of qualitative interviews with engineers were performed in two rounds. The goal of the first interviews which was accomplished in the beginning of the research project was to understand the development process and to learn about the systems, tools and applications that are used by the practitioners. Based on the interviews, the project’s goal was formulated and the criteria needed to be supported for achieving the projects’ goals were identified. Increasing quality of documents is the identified success criteria. Further, to improve the identified criteria, three factors were determined such as managing design rationale including capture, structure and access, traceability, and possibility to define product space from rules and parametric models.

The second round of interviews was performed to study the development process in more details and determine factors to elaborate the initial objectives. The generation, dissemination, and usage of design knowledge specifically design rationale across the development project was explored. In addition, the limitations and difficulties that the practitioners face when recording or searching for information were discussed in open discussions.

Currently, the research is in line with the third stage of the DRM, i.e. Prescriptive Study, where the concepts of the design support are developed by using the identified factors. The support is explained in the next section.

4 SUPPORTING CAPTURING AND SHARING DESIGN RATIONALE

The results of the interviews carried out at the company show that the engineers see the importance of design rationale and they acknowledge that it could support them to perform their tasks. However, when they were asked to provide some examples about what type of information is useful for them, the answers were quite broad. For example, one pointed at the need for explaining an expression in a CAD model, what the expression does, how changes in that expression effect other expressions, etc. One would like to have general information about a module, the reasons behind its design and its relations with other modules. One would like to know about the product requirements and product limitations (e.g. constraints in production) and how they affect the product space. The examples show how broad the design rationale is which makes the design rationale capture even more challenging. This requires capturing a complete history of the design including all design activities and steps and even those decisions that were taken first but rejected later. As one of the engineers in the company mentioned, this needs big investment to allocate the resources to understand, capture, analyse and maintain the knowledge. In addition, by capturing so much information and knowledge, there would be a big risk for information overload which is a bottleneck for rationale retrieval. Thus, it is necessary to find a balance between what is needed and what should be captured.

To get a better understanding about what rationale should be captured and in what details, a milling product was selected in the company. To introduce the IBIS method to the company, the design rationale for the milling product was captured according to the IBIS method. IBIS-based tools such as DRed (Bracewell et al., 2009) and Compendium (Shum et al., 2006) have been developed for capturing design rationale. However, in this paper a software called MindManager (MindManager) was used as an application to record the design rationale and present the functionality of the IBIS method (see Figure 1). The software is a mind mapping tool that enables organization of ideas and information.

The example presented in Figure 1 shows how to capture the design rationale. Please consider that this is a simplified example and just some relevant arguments and comments are presented here. An issue has raised in design (*how to ensure strength in the product?*). Two options which can affect the strength are identified by the designers. One is the size of the screw in the insert, the other one is the distance between the inserts in the cutter body. The example focuses on the design of cutter body and the distance between the inserts which is effected by other parameters such as number of inserts, design of inserts, chip material and design of the Chip space. The options, issues and arguments about the Chip space are captured. Some changes in design of Chip space are performed and a prototype is made. The results of the prototype test are captured and the arguments are written. The prototype is modified and tested again. Finally, the results meet the requirements. "P" in Figure 1 refers to expressions in the Chip space module in CAD model.

Figure 2 shows the development process in the company. The company develops different types of products (e.g. milling, turning and drilling). The product structure is basically the same within each type of product and is constituted by several pre-developed modules. For example, every milling product has a body, insert, insert screw, coupling, etc. Since it is a parametric design, the modules are flexible in size and can be reused in every new development project. So, when there is a new development project, the designers based on experiences from previous projects, have good knowledge about the product structure, what modules are going to be used, and what issues and problems need to be solved. For instance, *how to ensure strength in the product?* is an issue which raises up in every milling development project. This can be standardized by providing IBIS templates. In the IBIS template, the potential issues and their corresponding options and arguments are listed and as the design proceeds, the designer fills in or updates the template. Figure 1 shows an example of the parts that could be used to form the IBIS template. Currently, the PFD includes only design definition (e.g. CAD models and rules). By using the IBIS template, the design rationale is also captured.

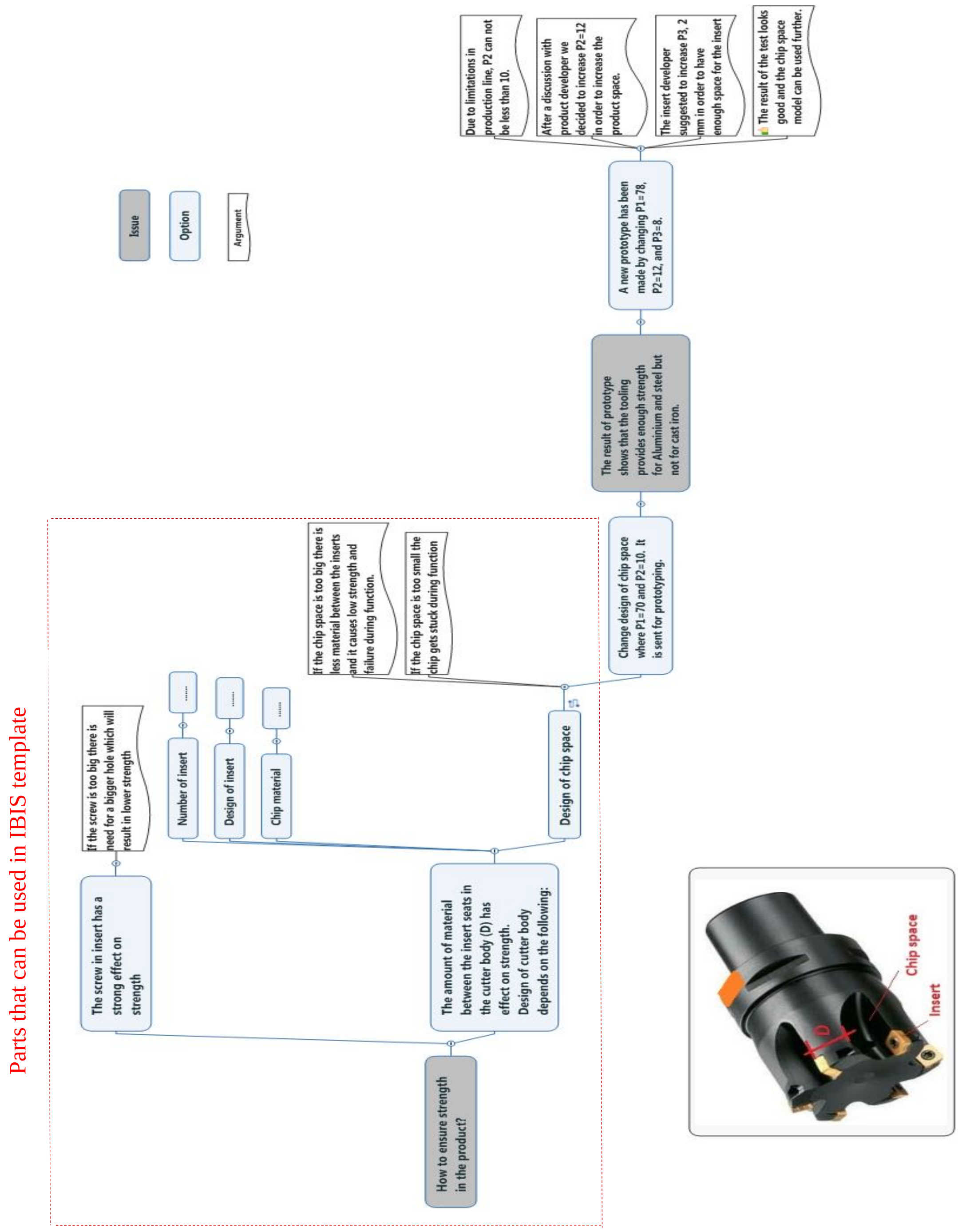

Figure 1. Capturing design rationale based on IBIS method

In the design programming step, an executable product family model is developed. The PFD is used as the basis for developing the product family model. Ideally, if the PFD is well structured, the design knowledge and design rules can be automatically extracted from the PFD to develop the product family model, however, currently it is done manually in the company. By executing the product family model based on customer specifications, unique product variants (within the product space), and corresponding information such as drawings, CMM and CAM instructions are generated. This process is completely automated.

Maintenance of the product family is carried out manually by the engineers and it is basically when a change is required based upon requests from marketing or production. To maintain and update the product family, the executable product family model and the corresponding design knowledge can be used. When changes are performed in the product family, the product family model can be executed to test the validity and effects of changes.

In addition, the product family model can provide different views of design rationale for different users. For example, the special designers can use the product family model to filter the information and access to what they need. Since both design definition and design rationale now exist in the product family model, any change or update in the product family is to be done in the product family model and not in the PFD or IBIS model.
The method and the example were presented for the engineers in the company through one meeting and two workshops following by a discussion about the applicability of the method. The feedback from the workshops was positive, however, there are some issues that should be addressed relating to the use of method across the development process and even during product maintenance. In addition, questions raised concerning *what rationale should be captured? who should capture the rationale, when should the rationale be captured?* and *how to share the rationale?* These are addressed in the following paragraphs.

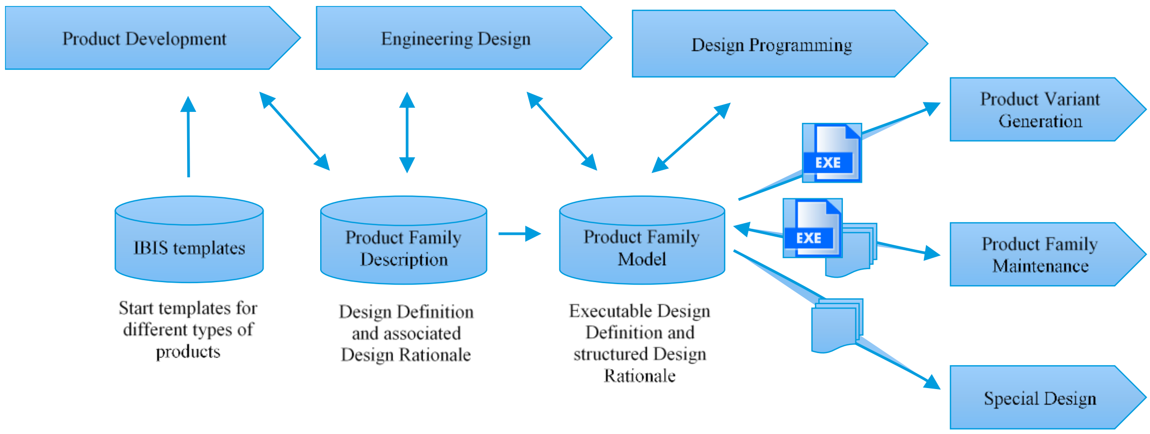

Figure 2. Capturing and sharing design rationale

4.1 What information and knowledge should be captured?

During the development process the designers make thousands of decisions. For example, a module in a CAD software could contain hundreds of features which each of them are designed in a way that to some extent correspond to product functionality and performance. In addition, there is a lot of dependencies between the features both internally inside the module and externally with other modules. So, capturing all reasons behind such decisions is not efficient and is time consuming.
There is a correlation between capturing the rationale and access and use of it. Capturing design rationale will be meaningful only when it is going to be used by people who are interested in it. Thus, it is important to investigate who are the users of design rationale and what information they need. Identifying the users is easy, however, identifying what information they need is a very difficult question since in principle every design information can be a part of design rationale.
Therefore, after a discussion with engineers it was suggested to start by capturing the major issues that have been solved during the development process. However, these issues should be the ones that will most probably raise again in a similar design project. For examples, issues such as *how to ensure strength in the product? how to prevent vibration in the product? what is the minimum amount of required material between the insert seats?* and *what is the relation between the chip space dimensions and type of material to be machined?* are to be solved in every development project. Although, the answer to these issues might differ a little bit from one product family to another family, but still there is a similar strategy to solve them. For example, as shown in Figure 1, in case of low strength in a milling product, the first options that a designer should consider are *size of the screw* and *the amount of material between the insert seats*.
Such type of rationale which is a bit general could be suitable for all design team members. However, users with different roles might require additional information that are captured specifically for their use. For example, the special designers asked for a general description about each module, as well as a description about the module's geometry including the user expressions, what each expression does and what is the acceptable range for the expression and why (the limitations or constraints for the expression). Table 1 shows an example of such information that is captured for the chip space. P1, P2

and P3 are the Chip space's expressions discussed in Figure 1. Please consider that this is a simplified example.

Table 1. Capturing and sharing design rationale for special designers

Expression	What does it do?	min/max	What are the limitations and constraints? Motivate (e.g. due to geometry, production, etc.)
P1	Wall angle, directional 2	Normal use is 20°-80°	Can be used with smaller and larger angle but it mainly depends on size of P2.
P2	Set the length, front to P3	Larger than 10 mm Larger than 12 if P4=90°	Due to limitations in manufacturing, it cannot be less than 10 mm.
P3	Wall radius 1	Larger than 0 and equal or larger than tool radius used in manufacturing.	Cannot be too small if P4 is smaller than 90° due to geometrical properties.

4.2 Who should capture the design rationale?

It is important to consider that a product development project might take a long time until the product family is ready to be released into the market. The output of the project is not only a product, but it is also the knowledge and experience that have been developed during this time. So, it is vital for the company to record their design knowledge for further reuse.

One suggestion is that the designers record design rationale as the design proceeds. However, recording the design rationale in a way that is easy for everyone to understand, requires training and competence. Another suggestion which was discussed during the workshop was to use an expert person for the sake of capturing and representing the rationale. The expert person should be skilled in modelling design information and knowledge. The expert person shall meet and interview the designers in different occasions and go through the rationale model and complete or modify the model as the design progresses. One advantage of having an expert person is that he/she might ask questions that the designers wouldn't ask each other or that they would consider it is obvious for everyone. The disadvantage, of course beside the cost for company, could be that the expert person might not be aware of the details of design intents.

Since it is the designer who is aware of the design intent, reasons, and downstream effects of his/her decisions, our suggestion is that the designer should be responsible for capturing the design rationale as the design proceeds. This happens currently in the case company into some extent where the designers record their decisions, intentions and reasons in form of test reports or technical memos. However, due to the extreme load of information which is recorded during a development process, after a while, the design rationale is invisible in the documentations. Since the design rationale capture might be a time-consuming task for the designers, use of advanced computer based tools, preferably with low intrusion to the design, can assist the designer in design rationale capture. In addition, providing IBIS templates containing those issues that are common in each development project and how they are solved can speed up the capturing process as the designer just needs to fill in or update the template.

An issue in capturing design rationale is that the designer might not be skilled in recording the information and knowledge in a way that is easy for others to understand, or that the designer is not sure of what information should be captured. Therefore, an expert person is required to guide and instruct the designers, provide a common language among the design team, and go through the rationale model and control it.

4.3 When should the design rationale be captured?

The author's experience shows that capturing design rationale in the end of a development project is not an efficient approach. As a part of collecting data for this research, the first author (could be considered as an expert person) tried to capture design rationale for the product case study. The product family had been released one year ago. During several interviews with the product designer, it was realized that

how difficult it was for the designer to remember what decisions he had made while ago and why. Therefore, it is suggested to capture, update and maintain the rationale as soon as a decision is made.
Having defined milestones where the designers and expert person meet and go through the rationale model can also emphasize and support the capture process. The designers in the case company evaluate their design by making prototypes of the CAD model (the number of prototypes made during development of the studied product was approximately 40). The designers usually write a technical report after each test describing the results, progress, issues and recommendations. This is a document that the designers rely on for making further decisions. So, our suggestion for the case company is to use the prototype tests as milestones where the expert person meets the designers and they extract the rationale from test reports.

4.4 How to share the design rationale for other engineers?

Currently, there is only one documentation (PFD) containing all design information and knowledge which is to be used by all engineers involved in the development project, maintenance and special design. This was an issue that raised up many times by the engineers during the interviews. What they want is to provide different views of design rationale for different users and instant access and filtering of information based upon what problem they are dealing with. Although this is not a simple task, but it can be done in to some extent by automatically generating a customized documentation of design rationale based upon the user's need. For example, based upon the special designer's request, a table like table 1 can be generated showing design rationale only about that module.
How to share the design rationale depends on how the user prefers. It can either be presented directly in the CAD software in form of comments or annotations, in a spreadsheet with predefined templates, or in form of an electronic book. These solutions have been tested in the company, however, more investigation is required.

5 DISCUSSION AND CONCLUSION

Nowadays, design processes have become knowledge-intensive and collaborative, therefore, it is vital to support share of design knowledge across the organization to improve the product development process. To develop or modify a product variant it is necessary to understand the design and reasons behind it. Such understanding can be achieved by access to design rationale explaining the justifications and reasons behind the decisions. Research into capturing and sharing design rationale especially for complex decisions has been the focus for years. Despite the efforts put into this topic, there is still a need for professional solutions that are easy to implement and use.
In the case company, a product space is created and the product family model is developed in a design automation system which enables generation of customized product variants. The information and knowledge that exist in the system needs to be maintained over time. Moreover, the system is to provide different views of the information and knowledge for different users.
To introduce the IBIS method to the company, a product family was selected and the design rationale for that was captured by the first author. The method was presented in the company and issues concerning applicability of the method were discussed during two workshops. A product designer, a special designer, a platform developer, and a design programmer participated in the workshops. The feedback from the workshops was positive, however, the engineers stressed the need to practice and evaluate the findings more in details. The use of IBIS templates seems to be a good strategy to shorten the time for capturing design rationale and structuring and standardizing the documentation. But more investigations are required to test the idea in a broader scale.
It is not efficient to capture reasons behind every single design activity. So, answering to the question *what design rationale should be captured?* totally depends on the user's need. Some suggestions can be to capture only the decisions that directly address the design goals and requirements, capture only the decisions that limit the product space, or capture the strategies for solving the major issues that raise during each similar design process. This will be studied more in the future research.
There is always a trade-off between allocating resources (e.g. expert person) for capturing design rationale and how much the company gains from using design rationale. It is not possible to analyse it in short term. More research is required on that.

REFERENCES

Blessing, L. T. and Chakrabarti, A. (2009), *DRM: A Design Reseach Methodology*, Springer.

Bracewell, R., Wallace, K., Moss, M. and Knott, D. (2009), "Capturing design rationale", *Computer-Aided Design,* 41**,** 173-186.

Burge, J. E. (2005), *Software engineering using design RATionale,* University of Illinois Urbana-Champaign.

Dutoit, A. H., Mccall, R., Mistrík, I. and Paech, B. (2006), "Rationale management in software engineering: Concepts and techniques", *Rationale management in software engineering.* Springer.

Elgh, F. (2011), "Modeling and management of product knowledge in an engineer-to-order business model", *International Conference on Engineering Design (ICED 11)*, Copenhagen, Denmark, 15.-19.08. 2011.

Elgh, F. & Cederfeldt, M. (2005), "A Design Automation System Supporting Design for Cost-Underlying method, system applicability and user experiences", *International Conference on Concurrent Engineering*, Texas, United States.

Elgh, F. & Poorkiany, M. (2012), "Supporting Traceability of Design Rationale in an Automated Engineer-To-Order Business Model", *Design 2012*, Dubrovnik, Croatia, 1425-1434.

Hicks, B., Culley, S., Allen, R. and Mullineux, G. (2002), "A framework for the requirements of capturing, storing and reusing information and knowledge in engineering design", *International journal of information management,* 22**,** 263-280.

Hooey, B. L., Da Silva, J. C. and Foyle, D. C. (2012), *A design rationale capture tool to support design verification and re-use.*

Johansson, J. (2008), *Design Automation Systems for Production Preparation.* Thesis for Licentiate of Engineering, Chalmers University of Technology.

Kunz, W. and Rittel, H. W. (1970), *Issues as elements of information systems*, Institute of Urban and Regional Development, University of California Berkeley, California.

Lee, J. (1997), "Design rationale systems: understanding the issues", *IEEE Intelligent Systems,* 12**,** 78-85.

Mayer, R. J., Griffith, P. A. and MenzeL, C. P. (1992), "IDEF6: A Design Rationale Capture Method", Concept Paper, DTIC Document.

Mindmanager, Available at: www.mindmanager.com (Accessed 2016-12-18).

Ouertani, M.-Z., Baïna, S., Gzara, L. and Morel, G. (2011), "Traceability and management of dispersed product knowledge during design and manufacturing", *Computer-Aided Design,* 43**,** 546-562.

Poorkiany, M., Johansson, J. and Elgh, F. (2016 a), "Capturing, structuring and accessing design rationale in integrated product design and manufacturing processes", *Advanced Engineering Informatics,* 30**,** 522-536.

Poorkiany, M., Johansson, J. and Elgh, F. (2016 b), "An explorative study on management and maintenance of systems for design and manufacture of customized products", *Industrial Engineering and Engineering Management (IEEM)*, IEEE, 1453-1457.

Shum, S. J. B., Selvin, A. M., Sierhuis, M., Conklin, J., Haley, C. B. and Nuseibeh, B. (2006), "Hypermedia support for argumentation-based rationale", *Rationale management in software engineering*, Springer.

Stokes, M. (2001), *Managing engineering knowledge: MOKA: methodology for knowledge based engineering applications*, Professional Engineering Publishing London.

Szykman, S., Sriram, R. D. and Regli, W. C. (2001), "The role of knowledge in next-generation product development systems", *Journal of computing and information Science in Engineering,* 1**,** 3-11.

ACKNOWLEDGEMENT

The authors would like to express their gratitude towards the participating company for providing the technical facilities for the research. The study is a part of the Impact project financially supported by the Knowledge Foundation in Sweden.