SUPPORTING TOOLING DESIGN OF CUSTOMIZED PRODUCTS BY INSTANT ACCESS TO DESIGN RATIONALE

Morteza Poorkiany, Joel Johansson, Fredrik Elgh

Department of Mechanical Engineering, Jönköping University, Sweden

morteza.poorkiany@jth.hj.se

Abstract: In an integrated product and production development environment, where changes in tooling design affect the product design and vice versa, access to design rationale of tooling would support concurrent development of new product variants and required tooling. This paper presents an information model that enables easy capture and access to the design rationale of toolings, moreover, supports tracing relevant information within different design software applications. A solution based on integrating SolidWorks, Microsoft Excel, and Microsoft Word has been developed and an industrial case study, where the system is introduced and evaluated is presented.

Keywords: Computer supported engineering design systems, design rationale, design rationale accessibility, and design rationale representation.

1. INTRODUCTION

The development and manufacture of customized products requires a complete understanding of both the product design and the production system and how they affect each other. This can be achieved by integrated product and production platform development. The use of platforms entails large investments and they are expected to be used for several years. Although, to stay competitive over the years, the platforms have to evolve which requires updates and redesign of product, tooling and production resources. Product variants derived from a product platform allowing for a high level of customization requires special tooling and equipment for manufacture adapted to the product design which is a challenge for the tooling design engineers. On the other hand, it is required that the product design conforms with the constraints and properties of the intended manufacturing processes, therefore, a close collaboration and support for sharing information between product design, tooling design and production would be beneficial.

According to (Groover, 2007), a production system includes people, equipment, and procedures organized to perform the manufacturing operations in the enterprise. A challenge in organizing production systems is providing the required tooling and equipment. Applying computer supported engineering systems in order to manipulate and process the engineering information can aid the design of tooling and equipment and facilitate performing engineering tasks. These systems can retrieve, process, store, and distribute information (Hicks et al., 2002).

Up to 80% of design time is spent on the tasks such as modifying or redesigning already existing and proven solutions (Stokes, 2001). This shows the significant value of redesign in order to shorten the delivery time and cut the production costs. Redesign requires obtaining and retrieving the design information from previous design activities. The design process of tooling is complex and involves a number of tasks and activities, generating and using a vast amount of information from different resources, such as CAD (computer aided design) files, bill of material, figures and design tables, stored in different knowledge repositories. A part of the design information is accessible through the available documentations, catalogues, CAD files, features, and design rules. This type of information (design definition) usually describes the results of the design, more often answering the question

"what". But if a tooling is going to be redesigned, more detailed information is required regarding the purpose of the design, the reasons behind the design decisions, the evaluated trade-offs, and the design alternatives. This type of information (design rationale) often aims to explain the tooling in the way it is designed answering to a more detailed question such as "why". Thus, making the design rationale accessible is a valuable step towards redesign of the tooling.

There is a big interest in industrial organizations to provide an integrated design environment for manipulating the design information. Such integrated environment is constituted of common software applications that are already used by the designers and a computer-supported engineering system. The advantage of that integration is that there is no need to introduce a new software application to the available software at the organization. In modern organizations, modelling a product in CAD software is a common approach. CAD models provide specific information to understand the layout of assembly and detailed information for each part. However, according to (Wang et al., 2012), this type of information fail to provide high-level semantics. Obviously, there is need for other tools to explain the product and the process, represent product variants, and generally provide semantic explanation. Therefore, the software applications that are commonly used in organizations can be used for manipulating and representing data as well as explaining and discussing the issues. The objective of this paper is to develop and explore a system that enables easy capture and access to the design rationale of tooling and equipment, moreover, supports tracing relevant information within common design applications such as CAD software for designing, spreadsheets for creating design tables and manipulating data, and documents in textual format.

The presented work in this paper targets a production development process in a modern enterprise where developing a customized product as well as developing the required tooling and equipment are carried out together at the company. The study investigates the applicability of the system for production of roof racks in a company delivering accessories to cars, both as a sub-supplier and by retailing in the consumer market. This paper is organized as follows. The fundamental concepts of design rationale and basic methods and approaches for capturing and representing design rationale are addressed first. Next, related works performed in the research area are discussed. In the following sections, the proposed approach and system architecture are presented with an example of a case study. Further, the paper is summarized and conclusion is provided.

2. FUNDAMENTAL CONCEPTS

This section briefly describes the basic concepts of design rationale. Furthermore, fundamental methods and approaches for capturing and representing design rationale are discussed.

2.1. Design Rationale

The research community has defined design rationale as a way to know the reasons behind a decision (Bracewell et al., 2009; Wang et al., 2012), however, it could also be the justification for it, the design alternatives, and the evaluated trade-offs that led to the decision (Falessi et al., 2006). Tang, et al. (2006) performed a research that showed up to 85% of designers agree to justify the design by design rationale and up to 80% of the respondents state they fail to understand the reason behind a decision without design rationale support. Furthermore, about 74% of respondents forget their own decision's reasoning from previous projects.

Access to design rationale can support development of new tooling or modification of existing variants (design changes) and gives insight into the reasons of why a decision has been made. Usually, there is a lack of explicit explanation of the way of creating an artefact in knowledge documentation in companies (Poorkiany et al., 2013). This could be due to other prior tasks, lack of time, lack of skill or lack of required applications. Most organizations try to keep and manage the generated information (e.g. aims, procedures, and processes) and store them in different repositories (e.g. catalogue, computer's memory, and people's memory). Defining the type of the design rationale and how it should be retrieved from these repositories and in which level is a fundamental issue and depends upon the aim and scope of the system, additionally, overloading information should be avoided (Regli et al., 2000).

Regarding capturing design rationale, Regli, et al. (2000) discuss two major methods: 1) *user-intervention-based capture* which is the documentation method created by the designers and the intention is to record the history of design activities as the design process evolves. This type of documentation helps people outside the project to understand the process and activities. 2) *automatic rationale capture* which records the communication among the involved people to extract design rationale and decisions as they proceed in design project. A drawback for

the latter approach would be that some people might not feel convenient to record their communications including their e-mails and telephone calls.

2.2. Representing Design rationale

Tang, et al. (2007) discuss template-based and argumentation-based as two approaches for representing design rationale. Template-based is an approach where standard templates are used incorporating into the design process. On the other hand, argumentation-based approach uses nodes and links. A node represents a component, while a link represents a relationship between components. This approach is a way to deliberate reasoning and decisions made during design process. With argumentation-based method the designer can easily trace the decisions and their relationships to the components. Moreover, finding the relation of components to extra documents (Meta knowledge) or relevant information is possible.

In addition, Regli, et al. (2000) mention descriptive approach for representing design rationale that is usually used in dynamic design domains, in which the problem is unclear and is barely possible to understand the final solution. In a research (Elgh and Poorkiany, 2012), a method for representing design rationale is presented. In that work emphasize is on design space solutions rather than standard existing solutions. The presented system suggests that a description could be created in the early stage of development process and evolves progressively in three steps: 1) conceptualization design, 2) design process and product family, and 3) design programming and manufacturing preparation. The description records the workflow and all design activities as well as describing why and when decisions are made and by whom. In addition, the description includes information regarding components, parts, assemblies, features, CAD-files, design rules, and material.

3. RELATED WORK

Considerable effort has been put into developing design rationale systems. However, it seems that mostly the developed design rationale systems are not in widespread use in industry and challenges still exist regarding effectively deploying the systems in industry (Regli et al., 2000). A significant task is to capture the design rationale while making the design decisions. Usually, this parallel working is difficult to be performed. A main reason is that the systems often enable capturing design rationale after making the decisions or even when the product is already designed. While capturing design rationale is a significant task, simply accessing to the design rationale is at the same level of importance. As Baxter, et al. (2007) mention, around 20% of the designer's time is spent searching for information and only 40% of design information requirements are met by documentation sources. This implies that design information and knowledge is not often represented in a simply accessible knowledge base.

A fundamental problem in the accessing and retrieval of information is the variety of applications and software to represent the information. A research (Elgh and Cederfeldt, 2010) carried out with the focus on documentation and management of product related knowledge. The purpose of the project was to reveal problems related to the reuse of design rules in a case company. Investigations in the studied company show that it is difficult for individuals to share their good developed solutions. The reason given for this is that there is no present system for such documentation covering all organizational contexts (in that case Microsoft Word, CAD, and programming software). Moreover, they discuss the difficulty in communication among the design engineers and the design programmers and stress that people are reluctant to add a new application to the tools in hand in order to improve the situation. Such circumstances and also the need for information exchange leads the organizations to move towards integration of independent tools for the sake of bettering information representation and contextual communication (Lundin, 2012).

An integrated representation of knowledge provides awareness of knowledge sources, access to the knowledge and easily communication among system users. Sandberg, et al. (2013) propose an approach that provides knowledge retention and sharing across the product development process in CAD environment. The discussed method enables adding design rationale within 3D annotations carried out by designers allowing design and documentation in the same time. In this case, 3D annotations may also include general text notes or hyperlinks to other sources of information such as textual documents, figures, spreadsheets and URLs. Although, the system prevents adding a new application in the organization, it limits its applicability to just CAE (computer aided engineering) environment. A solution to improve the situation can be applying software development enabling implementation of more software.

According to (Kim et al., 2007), redesign can be improved by either making the designer aware of the available information or providing a reuse process (based on query) in an embedded system. In that work, a software

prototype is developed and integrated into Design Rationale editor (DRed) which enables searching the required information according to the current task of the designer. The prototype system helps designers exploit reuse opportunities by recommending useful information which is not known to them and yet might be relevant to their current tasks. However, the relativity of the recommended information should be examined by the designers.

4. PROPOSED SYSTEM

The proposed system in this paper is supported by an information model that outlines the backbone in a computer-based solution (see figure 1). SolidWorks for CAD modelling, Microsoft Excel for rules definition and Microsoft Word for specifications are the three selected application programs for the proposed system.

The proposed information model consists of seven object classes, of which *Design Rationale, Description, Connection Group,* and *Design Rationale Connection* are general while *Word Rationale, Excel Rationale*, and *SolidWorks Rationale* target specific software applications. Basically, the information model tells that a *Design Rationale* consists of a set of *Connection Groups* and a set of *Descriptions*. A *Connection Group* on the other hand consists of a set of *Design Rationale Connections* that is either a *Word Rationale,* an *Excel Rationale*, or a *SolidWorks Rationale* (the *Design Rational Connection* is an abstract class).

When using the information model, the *Design Rationale Connections* carries information about where the actual design content is stored. These connections can be viewed as html hyperlinks, which can point to a specific file on the hard drive or a web-page at a certain URL, but the connections can be more specific than that, pointing to a specific range of cells in Excel, a specific feature or dimension in SolidWorks, or a certain bookmark in Word. The information model can be extended to target other software applications by implementing new types of Design Rationale Connection classes, indicated with dashed lines in Figure 1. The *Connection Groups* are used to cluster the *Design Rational Connections* in joints that makes sense, so that it is possible to group bookmarks in different Word documents, some ranges of cells in Excel spread sheets, and some dimensions or features in some geometrical models in SolidWorks that has something to do with each other in a natural way. Such a group can be viewed and utilized as a bilateral hyperlink that makes it possible to go back and forth from one connection point to another. So, for instance, when selecting a feature in SolidWorks that is pointed to by a Design Rationale Connection, all other connections in its group could be monitored to the user making it possible to jump to the connected Excel or Word entities, or other SolidWorks entities. To make *Connection Groups* meaningful they are stored as a *Design Rationale* object that can contain a number of such groups that are somehow naturally connected to each other. The *Design Rationale* object can also have a *Description* adding general information about the connected pieces of design knowledge. In the prototype system web-pages and pdf-documents are pointed to as *Descriptions*, but other types of documents could be added as description if beneficial.
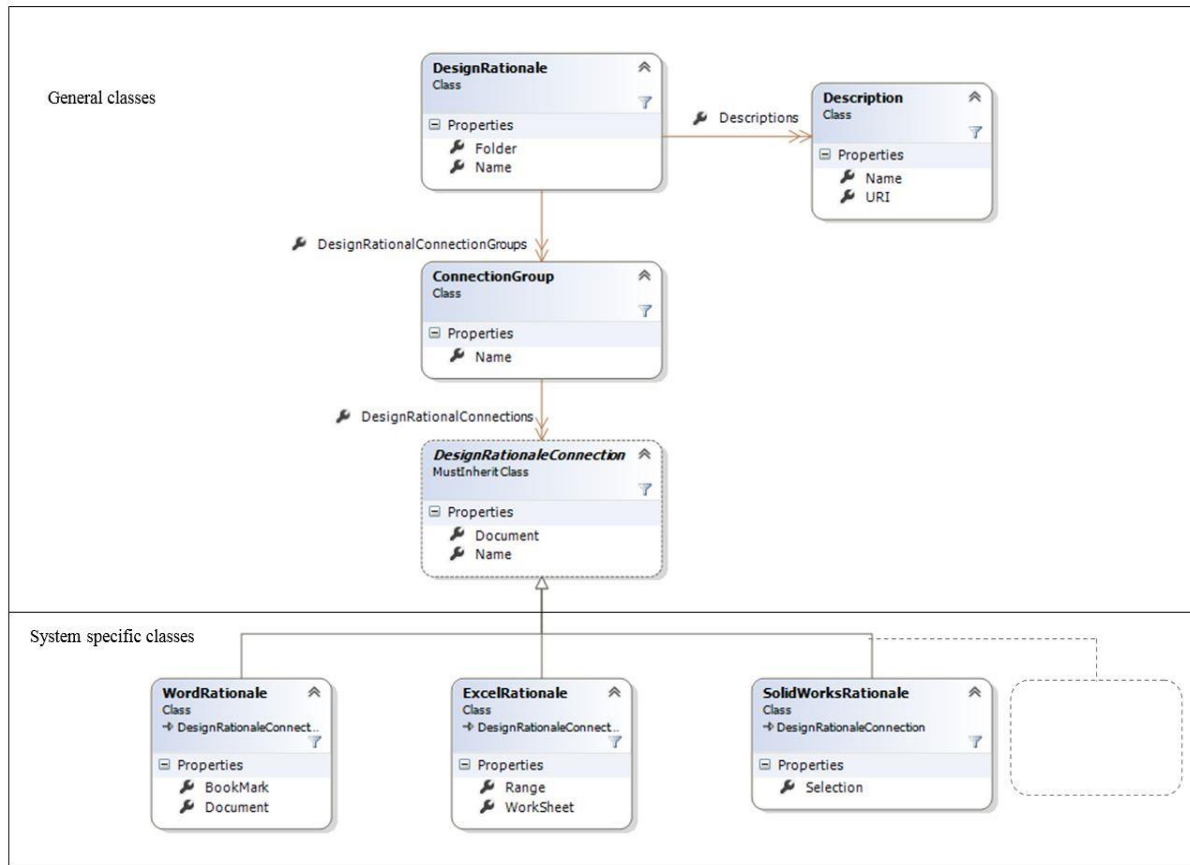
Fig. 1. The proposed information model.

4.1. System Architecture

A database that keeps track on all the design rationales with descriptions and connections can be used as the heart of the design rationale system. But in order to monitor available design rationales to the user when selecting different entities in the targeted software applications, a more accessible user interface has to be developed. Hence, it is suggested to as far as possible develop add-ins to the targeted software applications in a standardized way so that the users feel comfortable and recognize the system and the functions it stands for. In the standardized add-in user interface there should not only be functionality for monitoring available design rationales but also to make new connections and in such an accessible way capture the design rationale. To make the system complete it is also beneficial to develop an editor in which it is possible to overview and manage all the design rationales. The editor can include a vast number of functions of which the most fundamental includes, inserting, editing, deleting, and searching design rationales.

An overview of the proposed system architecture is presented in figure 2. The design rationale system can be implemented either as local installations on engineers desktop and laptop computers or as a centralized system on a server. In the former case, the design rationales are used for engineers like local scrapbooks to keep their own thoughts and connections representing the engineers' individual knowledge about the product. In the latter case, the design rationales stored on the server represent the corporate knowledge about its products.

4.2. Case Study

As a research case, an ongoing engineering design automation project was selected within automotive industry. The selected company for conducting the case study is Thule Group which is supplier of roof racks for about 95% of car models. Beside the roof racks, the company also provides bike and water sport carriers, roof boxes, and towing solutions. The company acts on the open market competing with car manufacturers and therefore, gets no nominal data of car's roof. Instead, the engineers have to collect geometrical information about car's roof by measuring. Then, based on collected geometrical data a roof rack is developed. Figure 3 presents a virtual model of roof rack. As it is shown in the picture, a roof rack contains two major parts labelled footpad and bracket in order to adjust the roof rack on the car's roof without need for rail. A roof rack will be adjusted to a

new car by slightly redesigning footpad and bracket. As the number of developed brackets (more than 1200) and footpads (more than 400) are increasing in the company (due to need of redesign for adaption to new cars), reuse of the previously developed solutions will be a big step towards delivering the product cheaper and faster. As an example reusing an existing bracket cuts the overall lead-time up to 40%.
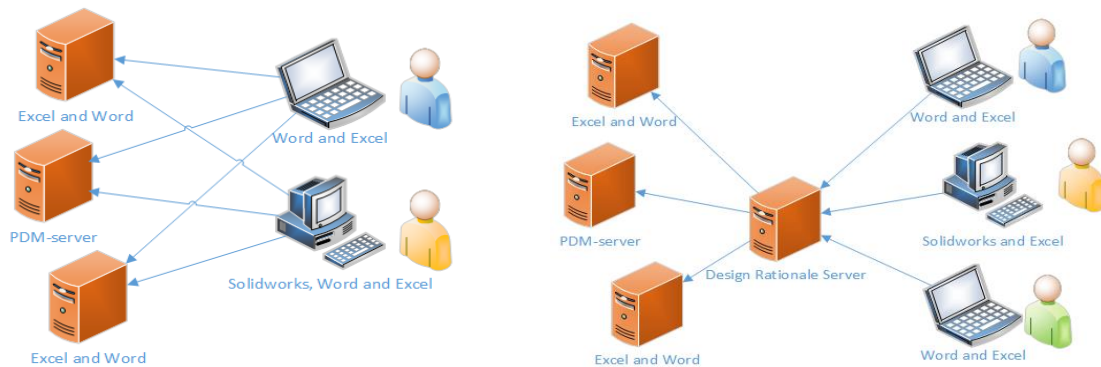

Fig. 2. The overview of the proposed system architecture. It can be either implemented to locally support engineers (at the left), or as a central server system supporting the whole corporation (at the right).
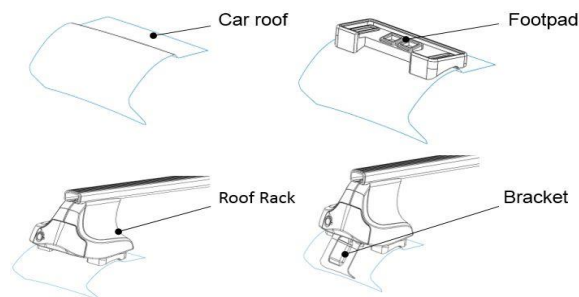

Fig. 3. A virtual model of roof rack.

In addition of developing the roof racks, the company develops tooling and equipment to be used for manufacturing of the roof racks. When a new car enters to the market, new bracket and footpad according to the car's roof specifications should be developed. Thus, developing new equipment and tooling are required to be used for production of new bracket and footpad. Manufacturing of the equipment is carried out traditionally at the company, using operations such as milling, pressing, and lathe machines. This requires skilled labour to perform the operations. To manufacture a new equipment the machines should be re-tooled and re-set according to the requirements. This is time consuming and causes in reducing efficiency and increasing manufacturing costs. Reuse of tooling will be a fundamental step towards decreasing their increment at the company and consequently reducing manufacturing costs. In order to reuse a tooling, access to its design rationale is required.

4.3. Prototype System

This project is initiated based on a concept and the concept is evaluated and evolved during the research. The mechanism of integrating the selected software applications in order to simply manipulate and communicate among them can be formulated in programming environment. Microsoft Visual Studio as programming software is selected. The program enables providing Add-Ins for each software.

The graphical user interface of the system consists of two task panes added to each targeted software application which one task pane (called *Design Rationale*) enables adding, removing, and browsing the design rationales instances, while the other one (called *Connections*) allows grouping the design rationales as well as linking the related design rationales together within different software applications. In addition, a window called D*esign rationale viewer* is added at the bottom of Excel and Word in which enables adding comments about an issue or a rationale instance or presents an attached web-page (see figures 4 and 5).

In order to test the functionality of the system, a tooling which is used for manufacturing of roof racks is selected. A design rationale instance can be created either in SolidWorks (by selecting a feature or component), Excel (by selecting a range of cells) or Word (by selecting a number of lines). Then, this rationale instance is presented in Design Rationale task panes in each software (no matter created in which one) and is possible to add another rationale instance to it as a connection in another software. Thus, the system enables linking, for instance, a dimension or feature in SolidWorks to a range of cells in Excel or a number of lines in Word and vice versa. Figures 4 and 5 present the selected tooling in SolidWorks and Excel. The added task panes to each software and the viewer window (just in Excel) are depicted by red colour. As an example, a design rationale instance is created in SolidWorks for the highlighted shaft. This is done by selecting the shaft, right clicking in the *Design Rationale* task pane and choosing Add in the opened window. Opening up Excel, the same created rationale will be visible in the *Design Rationale* task pane of Excel.

As figure 5 shows, the presented Excel worksheet contains a table, determining the valid tolerance for the diameter of the shaft. By selecting the desired cell(s) in the table, right clicking on the previously created rationale instance in *Connections* task pane, it is possible to create a connection between the shaft in SolidWorks and the design table in Excel. Now, if one clicks on the shaft in SolidWorks, all the connected rationales as well as the cell in Excel will be highlighted. In the same way, it is possible to link another Excel worksheet or workbook to the recently created rationale instances or link them to another SolidWorks file. If there is any attached file or web-page to the rationale instances, the file will be presented in the *Design Rational Viewer*.

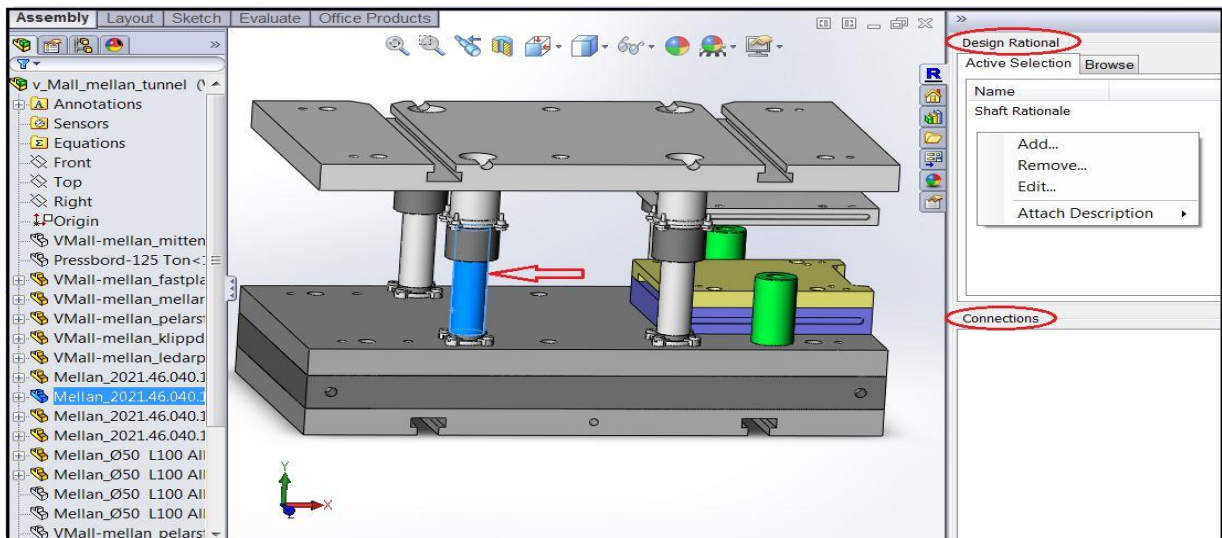Manipulating design rationales in Word is similar to Excel, therefore, it is not presented here.
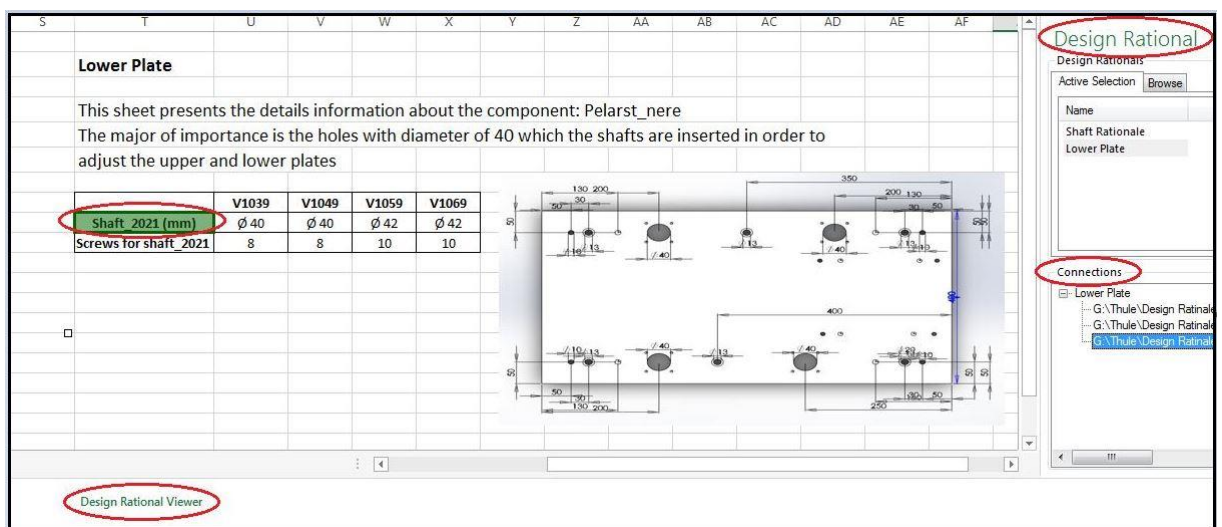

Fig. 4. A snapshot of SolidWorks.

Fig. 5. A snapshot of Excel.

5. DISCUSSION AND CONCLUSION

This research is performed to investigate capturing and accessing to design rationale within different contexts in the same time of designing a tooling. A strong emphasize is on implementing the method using the design software that are commonly used by practitioners instead of introducing new applications. A solution based on integrating SolidWorks, MS-Excel, and MS-Word has been developed. The system enables creating design rationale instances in each software and linking the related rationale instances together or to a knowledge source. The information model allows an interaction between the software applications, so, on selection of one feature in an application, connected features in other applications are highlighted. Thus, the user can provide more detailed information about the reasons of the design, for instance, the tolerance for a dimension in a CAD file, by attaching a description (e.g. pdf file or web-page) to it, or map it to a design table in spreadsheet or to a text file. If the CAD file is needed to be modified in the future, all the previously created rationale instances and the relevant information will be presented in the user interface.

So far, the prototype system is implemented as local installation on users' computers but in order to evaluate the functionality of the system, it is needed to be centralized on a server to be used by a group of users. In such scenario, a database is required to facilitate communication among the knowledge sources and also an editor to efficiently manage and edit the created design rationales. Although up to this point the created design rationales can be monitored according to their relations, but providing a search engine probably based on keywords would simply enable searching for required information. The practitioners at the case company found the proposed solution helpful and believe that it could considerably save time and cost in the redesign of the tooling. Although, the applicability of the system for a group of users, the communication among the users and the maintenance of the system are issues left for future research.

The proposed solution provides instant access to the design rationale of the tooling explaining the reasons behind its design. In an integrated product and production development where changes in product design affect the tooling design and vice versa, such detailed information about the tooling design would support redesign of the tooling as well as design of new product variants. It seems that, a more collaboration between product design and tooling design is achievable, however, this needs more investigations in future works. In order to evaluate information exchange between product design and tooling design, the authors aim to implement the proposed system in product design environment in future research.

6. ACKNOWLEDGMENT

REFERENCES

Baxter, D., Gao, J., Case, K., Harding, J., Young, B., Cochrane, S., and Dani, S. (2007). An engineering design knowledge reuse methodology using process modelling. *Research in engineering design* 18, 37-48.

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

Elgh, F., and Cederfeldt, M. (2010). Documentation and Management of Product Knowledge in a System for Automated Variant Design: A Case Study. *In* "New World Situation: New Directions in Concurrent Engineering", pp. 237-245. Springer.

Elgh, F., and Poorkiany, M. (2012). Supporting Traceability of Design Rationale in an Automated Engineer-To-Order Business Model. *In* "Proceedings of Design 2012, May 21-24, 2012, Dubrovnik, Croatia", pp. 1425-1434.

Falessi, D., Cantone, G., and Becker, M. (2006). Documenting design decision rationale to improve individual and team design decision making: an experimental evaluation. *In* "Proceedings of the 2006 ACM/IEEE international symposium on Empirical software engineering", pp. 134-143. ACM.

Groover, M. P. (2007). "Automation, production systems, and computer-integrated manufacturing," Prentice Hall Press.

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.

Kim, S., Bracewell, R., and Wallace, K. (2007). Improving design reuse using context. *In* "16th International Conference on Engineering Design (ICED'07)". Design Society.

Lundin, M. (2012). Knowledge Retention and Reuse.

Poorkiany, M., Johansson, J., and Elgh, F. (2013). A Case Study on Implementing Design Automation: Identified Issues and Solution for Documentation. *In* "20th ISPE International Conference on Concurrent Engineering: Proceedings", pp. 324. IOS Press.

Regli, W. C., Hu, X., Atwood, M., and Sun, W. (2000). A survey of design rationale systems: approaches, representation, capture and retrieval. *Engineering with computers* 16, 209-235.

Sandberg, S., Lundin, M., Näsström, M., Lindgren, L.-E., and Berglund, D. (2013). Supporting engineering decisions through contextual, model-oriented communication and knowledge-based engineering in simulation-driven product development: an automotive case study. *Journal of Engineering Design* 24, 45-63.

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

Tang, A., Babar, M. A., Gorton, I., and Han, J. (2006). A survey of architecture design rationale. *Journal of systems and software* 79, 1792-1804.

Tang, A., Jin, Y., and Han, J. (2007). A rationale-based architecture model for design traceability and reasoning. *Journal of Systems and Software* 80, 918-934.

Wang, H., Johnson, A. L., and Bracewell, R. H. (2012). The retrieval of structured design rationale for the re-use of design knowledge with an integrated representation. *Advanced Engineering Informatics* 26, 251-266.