Danijel Rohde
Mario Štorga
Dorian Marjanović

ISSN 1333-1124
eISSN 1849-1391

DESIGN RATIONALE CAPTURING MODEL FOR USE DURING THE EMBODIMENT PHASE OF THE PRODUCT DESIGN

Summary

An analysis of a design rationale focusing on the embodiment design stage is presented through the characterization of information describing design problems, alternative solutions, arguments, explanations and decisions of designers. The characterization of the design questions resulted from empirical research. The findings have been used to describe in detail elements of a design rationale capturing model. The main attributes describe this model during the embodiment phase of the product design, possible relations between the design rationale elements and the definition rules.

Key words: *knowledge management, issue, design problem, alternative solution, argument, decision, design rationale*

1. Introduction

Nowadays, it is easy to notice changes in the global market which dramatically influence industrial companies. The traditional understanding of how the companies view their product and the product's role in their business model is currently being reworked. Design cycles are becoming faster and pressure to save time and reduce costs is exerted from large corporations down to small enterprises confronted with tough rules in order to survive on the market.

It is known that the most important factors behind the company's success are loyal employees and the reuse of previously acquired knowledge. Unlike the times when people worked almost their entire lives in the same company at the same workplace, today's tendency is that people frequently change jobs, both inside and outside of the company. Such rapid changes result in a lack of time for employees to gain expertise in different, specialized areas of their activity. As these employees (mechanical engineers) begin to retire, they take with them their vast knowledge, which can be seen as another problem for the company. The design process has a significant impact on costs in product development [1]. For this reason, the pressure to save time and money leads to the acceleration of the product development cycles. In this situation, new employees do not have enough time to gain experience in designing new products in the domain of the company. Lack of knowledge related to the design rationale [2] can be a big step back and results in the prolongation of the design activities.

The primary aim of the research presented in this paper is the development of a design rationale capturing model that could be utilised during the embodiment phase of the product design. The research approach that is presented in this paper encompasses the results obtained from the developed classification of explanations and arguments related to the corresponding design problem solutions and decisions that could be used in the conceptual, embodiment and detail design stages as a part of complex and long-term design projects.

The motivation for the presented research originates from challenges that are common to engineering design departments in small and medium sized companies. Designing as a concept is used extensively in various disciplines, but understanding of the concept is not unique. According to Burge, the design process consists of a set of steps or actions that have to be taken to achieve the purpose of the product [2]. In this process, designers make decisions that are based on different reasons and arguments (such as economic, technological, time, or experience.). An improved process and a better final design can be achieved through the efficient and effective utilisation of information and knowledge resources for engineering design [3]. In the product development process, it is crucial to understand previously considered design solutions in order to redesign the existing product, adapt these solutions to design a new product and/or evaluate new design solutions. Knowledge management is one of the critical issues for the company, and there is a need for capturing, storing and reusing knowledge [4].

The structure of the paper is as follows. Section 2 gives a brief overview of the concept "design rationale" and its value for designers. Section 3 reports on the approach adopted for this research, and in Section 4, on the results of the empirical study. In Section 5, a design rationale capturing model is presented followed by an evaluation in Section 6. The final section discusses the implications and conclusions.

2. Background and related work

Literature review shows that during the development process, designers sometimes spend more than 50% of their time manipulating information. Also, over 50% of the information required for the realization of the product development is unavailable or unknown at the beginning of the process [3]. Designers typically spend 20-30% of their working time searching for and retrieving information [5]. For this reason, the efficiency of the design process and product quality is very dependent of the access to information and the ability to use large amounts of information [3]. For improving an existing product and developing a new product, the ability of designers to interpret past design solutions and decisions that such previous design solutions imply is crucial.

Technical description of a product, recorded in formal documents such as reports, technical drawings, and 3D models, describes results of the development process but not the development process and the decision-making process themselves. Especially, there are no records on decisions and arguments that are based on experience. This problem leads to a loss of knowledge gained while solving design problems. In literature, this kind of knowledge is called design rationale. Design rationale includes the reasons behind a design decision, the justification for a design decision, other considered alternatives, evaluated tradeoffs and argumentation that led to a decision [6].

In general, the systems proposed in the literature for capturing, storing and reusing information and knowledge (design rationale) can be divided into: (1) the systems which require involvement of designers during the capturing of information; and (2) the systems where this procedure is more or less automated by the use of artificial intelligence [7], [8]. Automated systems are applicable only when development tasks are previously known. For example, this occurs during the detailed design phase especially in the development of variant products. Systems that require an involvement of designers in the capturing of information and knowledge shall apply during all stages of the development of new products or new functionalities of existing products. In both cases, if the employee would like to capture "informal" information and knowledge it is necessary to include designers in the knowledge capturing process.

Hu et al. [9] present a review of design rationale systems and representation languages proposed by different authors up to 1999. They found that the most common argument structures for selecting and organizing information are IBIS, PHI, QOC and DRL:

1. IBIS (Issue Based Information System) [6]: issue, positions, arguments
2. PHI (Procedural Hierarchy of Issues) [10]: issue, answers, arguments
3. QOC (Question, Option and Criteria) [11]: question, options, criteria
4. DRL (Decision Rationale Language) [12]: issue, alternatives, goals

Despite over 30 years of research, there are still few rationale systems used in practice [13]. There is a strong consensus that rationale is valuable, but there is an equally strong concern that the costs of its capture may be too high. In order to justify the costs of its capture, it is essential to establish ways in which rationale can be useful that exceed a simple provision of additional design documentation [13]. Design rationale capturing tools are beginning to be accepted in the industry, e.g. the Design Rationale editor (DRed) developed by Engineering Design Centre (EDC) of Cambridge University [14]. Preliminary research on the DRed usage in practice shows that structured information is easier to interpret than traditional design definition reports [15]. DRed is a graphical software tool for acquiring knowledge in the early stages of the design process and solving the problems that occur during the product servicing.

As a starting point of our research the previously mentioned DRed and Compendium have been used [16]. The available literature shows that there is an unexplored area of the knowledge management, specifically relating to the classification of design problems, explanations regarding alternatives and arguments that lead to design decisions.

2.1 Design reasoning process

By describing relationships between data, information and knowledge the authors [17] summarized that in the field of engineering design there exist two aspects of knowledge production: (1) knowledge processes – generated by an individual through the understanding, assimilation and application of information and other knowledge elements; and (2) knowledge elements – produced by the learning processes. As well as other humans, designers infer new knowledge elements from information, other knowledge elements or a combination of all. Roozenburg and Eekels call this inferring process reasoning. Reasoning can be defined as a cognitive process of searching for reasons for own beliefs, conclusions, actions or feelings [18].

The result of the design process is a technical product. During that process, different kinds of information are appearing. By analyzing data collected during the development projects involved in this study it was observed that the collected information can be grouped into:

1. information about projects, people involved and product development stages
2. information that represent the content of the design reasoning
3. information about how to retrieve information captured during previous design reasoning processes (information that concern storing, accessing and searching)

In line with the research goal the focus in this chapter is on information that forms the design reasoning content. There was observed the existence of formal and informal type of information which can be grouped into:

1. information that affects decisions in the design process – related to the objectives and requirements defined in any of the product development stages
2. information that describes (explains) decisions in the design process – descriptions of problems, alternatives and arguments used for the evaluation of alternatives
3. information about the design artefact – product, components, assembly, assembly connections and constraints

3. Research methodology

Following *the Design Research Methodology* [19], stages and the main outcomes of the research are presented [Fig. 1].

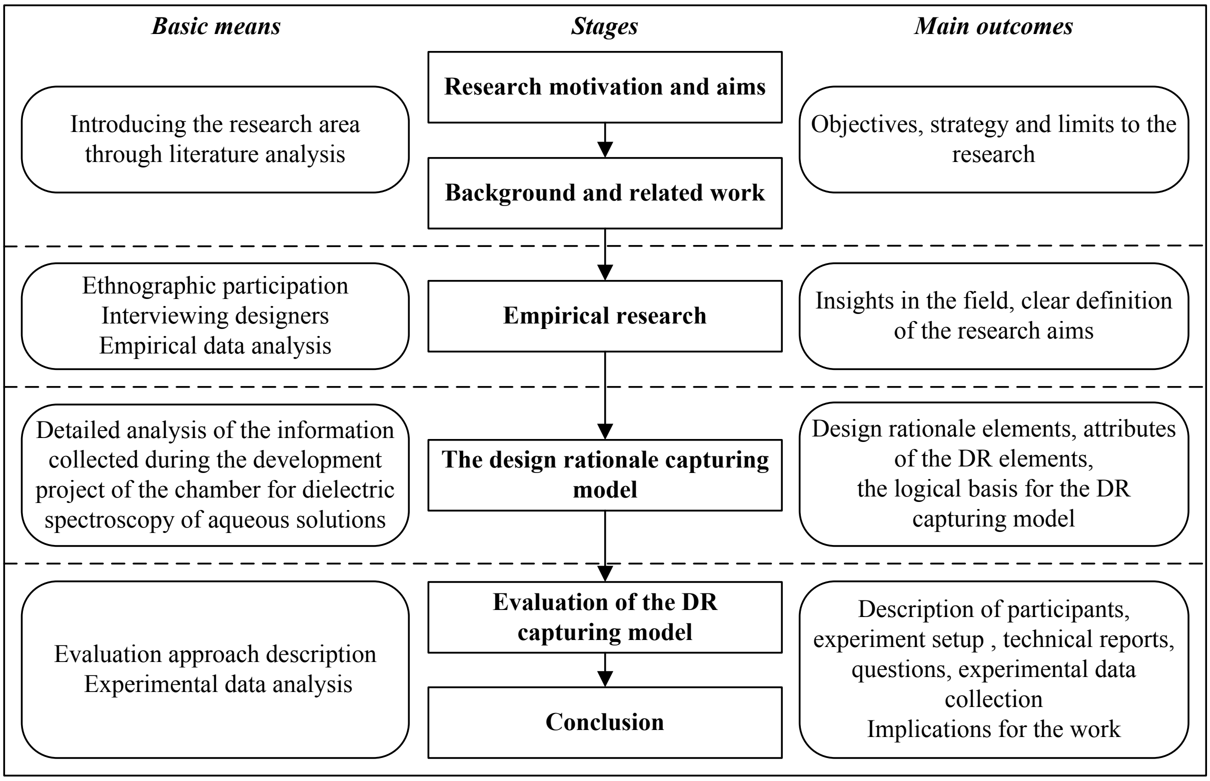

Fig. 1 The structure of the research methodology

During the literature review not enough evidence to clearly determine all aspects related to resolving design problems during the embodiment design stage was found, so authors decided to observe and interview designers at work to obtain a better understanding of the research issues. The design rationale analysis showed the complexity of questions involved in knowledge capturing during the embodiment design phase. The study was conducted with an emphasis on solving design problems. The results of the analysis are distinctive attributes of the design rationale (*DR*) elements: (1) problem, (2) alternative, (3) argument, (4) explanation and (5) decision. Description of the attributes, possible connection rules between the *DR* elements and the definition rules for the *DR* elements are the basis of the proposed design rationale capturing model for use during the embodiment phase of the product design. A detailed analysis of the proposed model in a real product development process is done only partially by conducting an experiment and analysing the data collected during the experiment.

4. Empirical study

The goal of the empirical study was to answer the following research question:

1. What kind of data, information and knowledge should be captured (considering the needs of companies in the observed domain)?

Since the answer to that research question required direct and indirect observation of designers during the product developing process, it was necessary to take the following into account:

- time that the respondents (designers) can be available for discussion with the researcher.
- confidentiality of information obtained from the observation and discussion with the designers.

This phase of research is divided into three subphases:

1. Ethnographic research
2. Interviewing designers
3. Data analysis

A benefit of the investigation conducted in industry is that the researcher is involved in the social and technical context and has an opportunity to get to know the rest of the team [20]. It has been observed that ideal case is when the researcher's level of knowledge is close to the level of the knowledge of professional designers. For these reasons, the researcher has spent some time (18 months) working on one of the current projects of the associate company.

The ethnographic research was conducted through the active participation of the researcher in solving design problems during the development process of a backpack sprayer and duster unit for agricultural usage. This kind of active participation enabled us to collect data in a neutral atmosphere. We had assumed that it would help us to reduce bias regarding the behaviour of participants because they knew they were participants in an experiment. During the project the researcher recorded his decisions, contacts, calculations, drawings, presentations, CAD models, sketches and time spent on each design task in a log-book. Interviews with other designers were conducted to obtain: (1) relevant information about the design process that was implemented in the company and (2) designers' opinions about the types of information and knowledge that should be collected for use in future projects.

The collected data were subsequently analyzed, and the results were used in the next phase of the study. Because designers define design problems in the form of questions, there was a possibility of applying research results from the scientific literature. So, in the experiment we decided to use Aurisicchio's taxonomy of knowledge requests (questions to which designers are looking for answers) [20]. Aurisicchio has proposed a coding scheme for his taxonomy in which each element of the taxonomy is indexed with two digit code. One particular request is characterised (indexed) with an appropriate combination of items from each group of categories e.g. D2-E1-F1-G5-H7-I5 (*Objective - Subject - Response process - Response type - Direction of reasoning - Behaviour type*). The applicability of Aurisicchio's coding scheme has already been investigated [21]. We started to apply such a coding scheme to a subset of knowledge requests that were generated in the development process of the backpack sprayer and duster unit for agricultural usage. From the log-book, written during the ethnographic research, we extracted 120 knowledge requests concerning the development of the dosage handle subassembly. Each knowledge request was further described as the following set of recorded attributes: *aim, information source, media, start/end date, outcome and comments*. Each knowledge request being treated as a class of a knowledge element (chunk) had in addition a set of hyperlinks to relevant documents and CAD files. The next phase of the experiment included the generation of codes for each knowledge request recorded in the database [21].

In the process of coding, for some requests it was not always easy to decide which subcategory the request should belong to. For this reason, we tested the repeatability of the generated codes. One week after the first coding, the coding process was repeated on the whole record set of knowledge requests. The code comparison showed that for only 40% of requests an entirely the same identical code was made in both attempts. Then, we compared only the first four categories (D-E-F-G), which showed that there were 68.33% of knowledge requests with the same code. In the end, we compared the combination of three categories (D-E-F; D-E-G; D-F-G and E-F-G), and the results showed that there were 100 (83.33%) of knowledge requests with the same code. As a result of this analysis, we noticed a need for further investigation of information requests to select those classes that would ensure unambiguous categorization. It was also necessary to explore other design rationale elements during the embodiment design phase.

5. The design rationale capturing model

The literature does not often show a clear difference between the problem-solving process and the task solving process. Problem-solving can be associated with the case when in the beginning of the process the methodology is not known. So, if the solving methodology is not known in advance, it could be said that the designer resolves the design problem, otherwise the designer resolves the design task. Also, it is important to emphasize the dominance of the intuitive and creative thinking (during the design problem-solving process) that is very difficult to formalize. We assume that only the designers who are responsible for the problem resolving can describe *DR* elements, and they should be included in the information capturing process. To help designers to define *DR* elements and connections/links between the *DR* elements it is necessary to propose attributes and their possible values.

The design rationale capturing model [22] is described by primary attributes and possible relations between the design rationale elements. These elements are briefly explained below and in the following chapters.

The basic types of *Design Rationale Scheme* (DRS) elements are: (1) problem (P); (2) alternative (S); (3) argument (A); (4) explanation (E); and (5) decision (D), [Fig. 2].

A *Design Rationale Scheme* (DRS_i) consists of a finite set of basic elements linked with directed arrows. It necessarily contains only one problem P_i for which it is necessary to define all attributes. Essential attributes which describe the problem are: (1) Type; (2) Question; (3) Question goal; (4) Problem description; (5) Cause of the problem; (6) Subject; (7) Direction of reasoning; and (8) Status. For a problem P_i there can be proposed a finite set of alternatives $S_j = \{S_1 \cup S_2 \cup \ldots \cup S_u\}$. The set S_j can be an empty set $S_j = \phi$, which means that for the defined P_i no alternative is proposed. For alternatives, S_j it is necessary to define all of their attributes. Essential attributes that describe the alternative are: (1) Type; (2) Source; (3) Representation; (4) Description; (5) Usefulness; and (6) Status. All defined alternatives S_j need to be linked to the problem P_i. The link between the alternative and the problem contains one of the attribute values of the alternative named *usefulness*: (1) unsatisfactory alternative; (2) just about acceptable alternative; (3) satisfactory alternative; (4) good alternative; and (5) very good alternative. Values, that an attribute named *usefulness* can take, depend on the evaluation range that is used in the company. Here, the assessment according to the guideline VDI 2225 [23] is used. Evaluation and assessment are not the focus of this paper, but the identification of the attributes that describe the alternatives is.

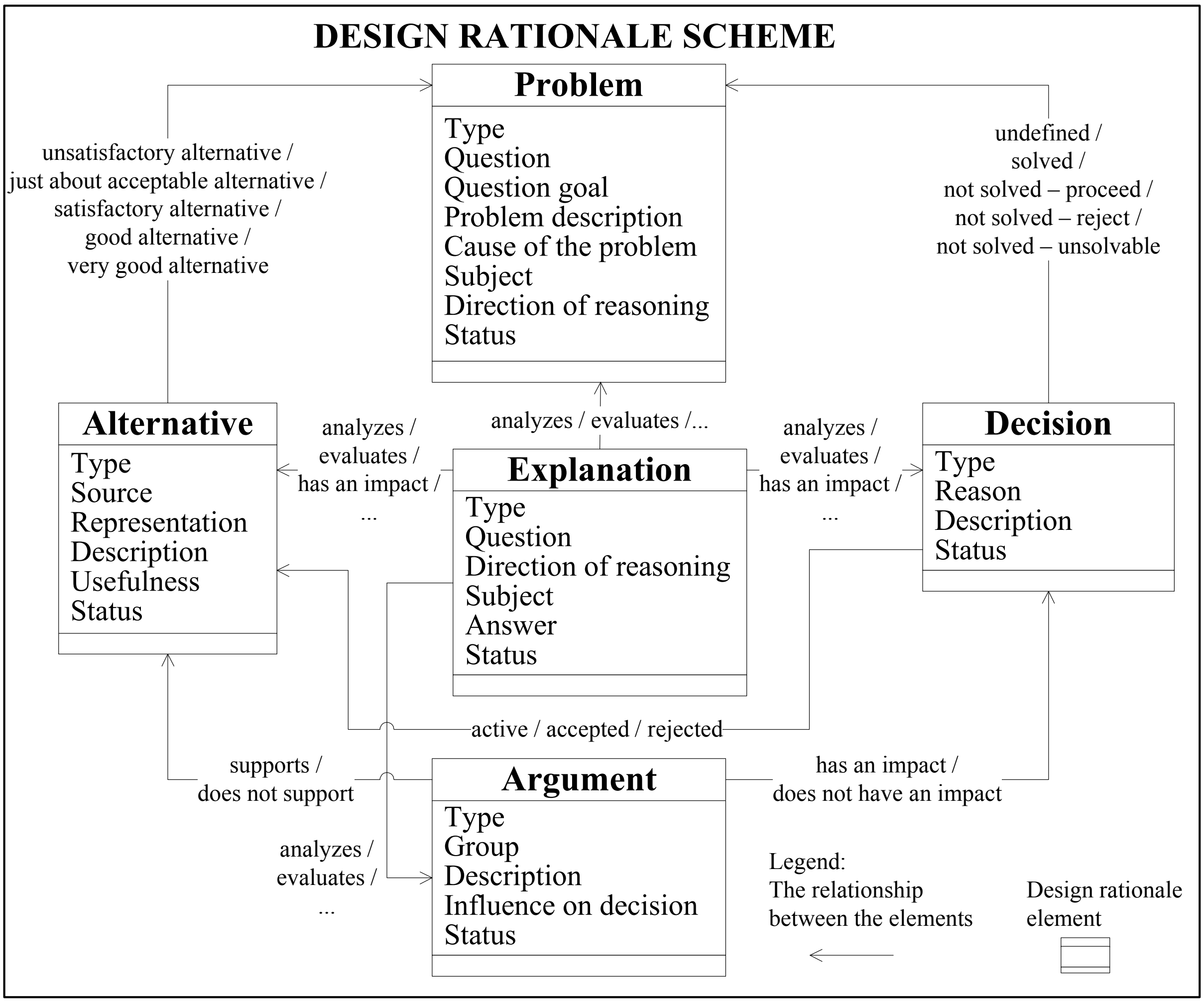

Fig. 2 Design rationale capturing model

With every defined alternative $S_u \in S_j$ a finite set of arguments can be proposed: $A_k \subseteq A_l$, where $A_l = \{A_1 \cup A_2 \cup \ldots \cup A_s\}$ is a set of all proposed arguments in one DRS_i. Set A_k can be an empty set $A_k = \phi$, which means that for a proposed alternative $S_u \in S_j$ no argument is proposed. Set A_l can also be an empty set $A_l = \phi$, which means that for the proposed alternatives S_j no argument is proposed. If $A_l = \phi$ is valid, then $A_k = \phi$ is valid too since $A_k \subseteq A_l$. For arguments, A_k it is necessary to define all their attributes. Basic attributes that describe an argument are: (1) Type; (2) Group; (3) Description; (4) Influence on decision; and (5) Status. There is a need to link all of the defined arguments A_k with the alternative $S_u \in S_j$. The link between the argument and the alternative contains one of the attribute values of the *Argument type*: (1) supports an alternative, or (2) does not support an alternative. An argument $A_s \in A_l$ can be linked with more alternatives from the set S_j. In this way, the attributes that describe the argument are not changed except the *Argument type* that is an integral part of the link.

A problem P_i is necessarily linked with only one decision D_i, for which it is necessary to define all of its attributes. Basic attributes that describe a decision are: (1) Type; (2) Reason; (3) Description; and (4) Status. The table shows changes in individual attributes values of the problem and the decision from the initial state to the final state [Table 1].

Table 1 Possible change in individual attributes values of the problem and the decision

Case		Decision type	Decision status	Decision identification ($i, j \in N$)	Problem status	Problem identification ($i, j \in N$)
A	initially	Undefined	Active	D_i^j	Active	P_i^j
	…	Proceed	Accepted	D_i^{j+1}	Active	P_i^{j+1}
	finally	Solved	Accepted	D_i^{j+2}	Solved	P_i^{j+2}
B	initially	Undefined	Active	D_i^j	Active	P_i^j
	finally	Solved	Accepted	D_i^{j+1}	Solved	P_i^{j+1}
C	initially	Undefined	Active	D_i^j	Active	P_i^j
	finally	Not solved – reject	Accepted	D_i^{j+1}	Rejected	P_i^{j+1}
D	initially	Undefined	Active	D_i^j	Active	P_i^j
	finally	Unsolvable	Accepted	D_i^{j+1}	Unsolvable	P_i^{j+1}
E	initially	Solved or Proceed or Not solved – reject	Accepted	D_i^{j+1}	Solved or Active or Rejected	P_i^{j+1}
	…	Undefined	Rejected	D_i^{j+2}	Active	P_i^{j+2}
	…	Undefined	Active	D_i^{j+3}	Active	P_i^{j+3}
	finally	Solved or Not solved – proceed or Not solved – reject	Accepted	D_i^{j+4}	Solved or Active or Rejected	P_i^{j+4}

The decision D_i is linked to the problem P_i. The link between the decision and the issue contains one of the attribute values of the *Decision type*: (1) undefined; (2) solved; (3) not solved – proceed; (4) not solved – reject; and (5) not solved – unsolvable (there exist strong arguments in the designer's point of view that the problem is unsolvable, at least at that point of time.

Arguments from the set A_l are linked to the decision D_i. The link between the argument and the decision contains one of the attribute values of the *Impact on decision*: (1) has an impact, or (2) does not have an impact. The decision D_i is linked to the alternatives S_j. The link between the decision and the alternative contains one of the attribute values of the *Alternative status*: (1) active; (2) accepted, or (3) rejected.

One DRS_i can include a finite set of explanations $E_m = \{E_1 \cup E_2 \cup \ldots \cup E_r\}$, for which it is necessary to define all of their attributes. Basic attributes which describe the explanation are: (1) Type; (2) Question; (3) Direction of reasoning (4) Subject; (5) Answer; and (6) Status. The set E_m can be an empty set $E_m = \phi$, which means that for the proposed DRS_i no explanation is proposed. All proposed explanations E_m are necessarily linked to at least one element as a part of DRS_i (problem, alternative, argument, explanation, decision). The link between the explanation and other *DR* elements can have the following values: (1) is a qualitative fact; (2) is a quantitative fact; (3) analyzes; (4) evaluates; and (5) has an impact on the decision.

All proposed *DR* elements as a part of DRS_i can be linked with the finite set of the *Information objects* $IO_n = \{IO_1 \cup IO_2 \cup \ldots \cup IO_p\}$ [24]. The link can have a value: is connected. One IO_p can be linked with one or more *DR* elements that are part of the finite set $DRS_o = \{DRS_1 \cup DRS_2 \cup \ldots \cup DRS_i\}$. The *DR* element (problem, alternative, argument, explanation, decision) can be deleted only when DRS_i and P_i are in the active status. The status and identification of DRS_i are the same as the status and identification of P_i.

There is a possibility to connect two or more Design Rationale Schemes which are part of the finite set $DRS_o = \{DRS_1 \cup DRS_2 \cup \ldots \cup DRS_i\}$ to each other. For example the proposed *DR* elements that are a part of DRS_i can be linked with *DR* elements as a part of DR_1.

The model should be flexible, which implies the possibility of adding new values for *DR* element's attributes.

6. Evaluation of the *DR* capturing model

6.1 Evaluation approach

Experimental evaluation was carried out by two groups of students (who were in the final year of the engineering design study programme, product development project related course). Each group consisted of five interviewees who read and interpreted technical reports with the purpose to find answers to predefined questions. The authors prepared questions and technical reports in order to determine how the defined DR elements and their attributes affect:

- the understanding of information of the product and product development;
- the information searching process.

The experiment was designed to simulate the real working situation in which designers retrieve information and knowledge captured during previous product development projects. It is important to emphasize that the participants (interviewees) had no previous contact with any information related to the project used in the preparation of technical reports for the experiment. In this way, authors wanted to eliminate the influence of past experience and knowledge of the experimental results. The experiment consisted of four major steps [Table 2]. At the beginning of each question, interviewees had a specified amount of time (30 minutes for each group of questions) for reading, interpreting and understanding (the test experiment with another group of students had been carried out in advance in order to determine the time required to answer the questions). After that, they read and interpreted the information from the predefined technical report. In the third step, they retrieved information from the report and wrote their answers in the pre-prepared templates. Before the experiment started, the interviewees had been randomly divided into two groups. Each group consisted of five interviewees. To answer the questions, two groups of participants (*A,* B) alternately used two types (*I,* II) of technical reports. The reports were related to the development of a chamber for dielectric spectroscopy. Report I had usual content (such as description of tasks, a list of requirements, description of technical processes) written in MS Office and Adobe PDF documents. Report II had the same information as report I but it was recorded as a graphical representation by using a DR capturing model. It is also necessary to say that the interviewees had access to the final version of the technical drawings with both reports.

Group A started the experiment by using the first type of report (Report I) while group B began the experiment by using the second type of report (Report II). After they finished with the first group of questions, Group *A* continued with Report II, and Group B with Report I. In the same way, the participants responded to all questions prepared in the experiment [Table 2]. Since the experiment required concentration of the participants, it was necessary to limit the time allowed for answering questions. In order to determine the required time authors made a test experiment, which involved three interviewees. All prerequisites for the main experiment were also related to the testing experiment (participants without experience regarding the observed project).

Table 2 Procedure of the experiment

Step	Group of interviewees	Group of questions	Report
1	A	1	I
	B	1	II
2	A	2	II
	B	2	I
3	A	3	I
	B	3	II
4	A	4	II
	B	4	I

In the test experiment, participants answered one question from each group. They needed in average 15min for answering one question. The information obtained and the experience gained from the test experiment was subsequently used to prepare the main experiment.

All answers to the given questions collected during the experiment were evaluated for completeness and correctness [Table 3].

Table 3 Evaluation metrics

Evaluation metrics	Definition
completeness	The completeness of the information returned by a participant. How much relevant information is offered in a given answer.
correctness	The correctness of the information returned by a participant

Table 4 An example of a question and associated answer

Question	Answer
Which external properties had an impact on the choice of one of the given alternatives for connecting the chamber to the "Agilent" measuring device?	1. Exploitation properties 2. Economical properties 3. Manufacturing properties 4. Functionality

All questions required answers composed of multiple entities [Table 4]. For this reason, the completeness and correctness were determined by counting correct and incorrect entities returned by the participants in the experiment:

- Completeness: the number of correct entities in a given answer (from one participant) divided by the total number of possible correct entities in the technical report.
- Correctness: the number of correct entities in a given answer divided by the total number of entities in the answer (from one participant).

In this way, answers could be compared because the limiting values (minimum and maximum) for completeness and correctness were the same (0 to 1).

6.2 Content of the reports used in the experiment

The documentation that was utilized in the preparation and implementation of the experiment was made during the development of a chamber for dielectric spectroscopy of aqueous solutions. The document with information structured in eight chapters was used in the experiment as Report I [Table 5].

Table 5 Content of the first type of report (Report I)

Chapter	Content
1	Introduction
2	Tasks clarification
3	Functional decomposition
4	Technical solution principles
5	Evaluation and selection of the best alternative
6	Engineering calculations
7	Conclusion
8	Appendix: Technical documentation

Information from Report I was also used for preparing Report II but written in a structured way using the design rationale elements proposed in the model in chapter 5 [Table 6].

Report II contained 22 DRSs related to 22 problems for which the development team proposed 52 alternatives in total. There were also 55 arguments identified related to these alternatives that were a base for 22 decisions. These problems were linked with 19 explanations with information about facts, analyses and evaluations. Figure 3 shows an example of one DRS with three alternatives for a given problem, four arguments, two explanations and one decision [Fig. 3].

Table 6 Design rationale elements used in Report II

	Active	Accepted/ Solved	Rejected
Problem	Problem Active	Problem Solved	Problem Rejected
Alternative	Alternative Active	Alternative Accepted	Alternative Rejected
Argument	Argument Active	Argument Accepted	Argument Rejected
Explanation – fact	Explanation Active Fact	Explanation Accepted Fact	Explanation Rejected Fact
Explanation – analysis	Explanation Active Analysis	Explanation Accepted Analysis	Explanation Rejected Analysis
Explanation – evaluation	Explanation Active Evaluation	Explanation Accepted Evaluation	Explanation Rejected Evaluation
Decision	Decision Active	Decision Accepted	Decision Rejected

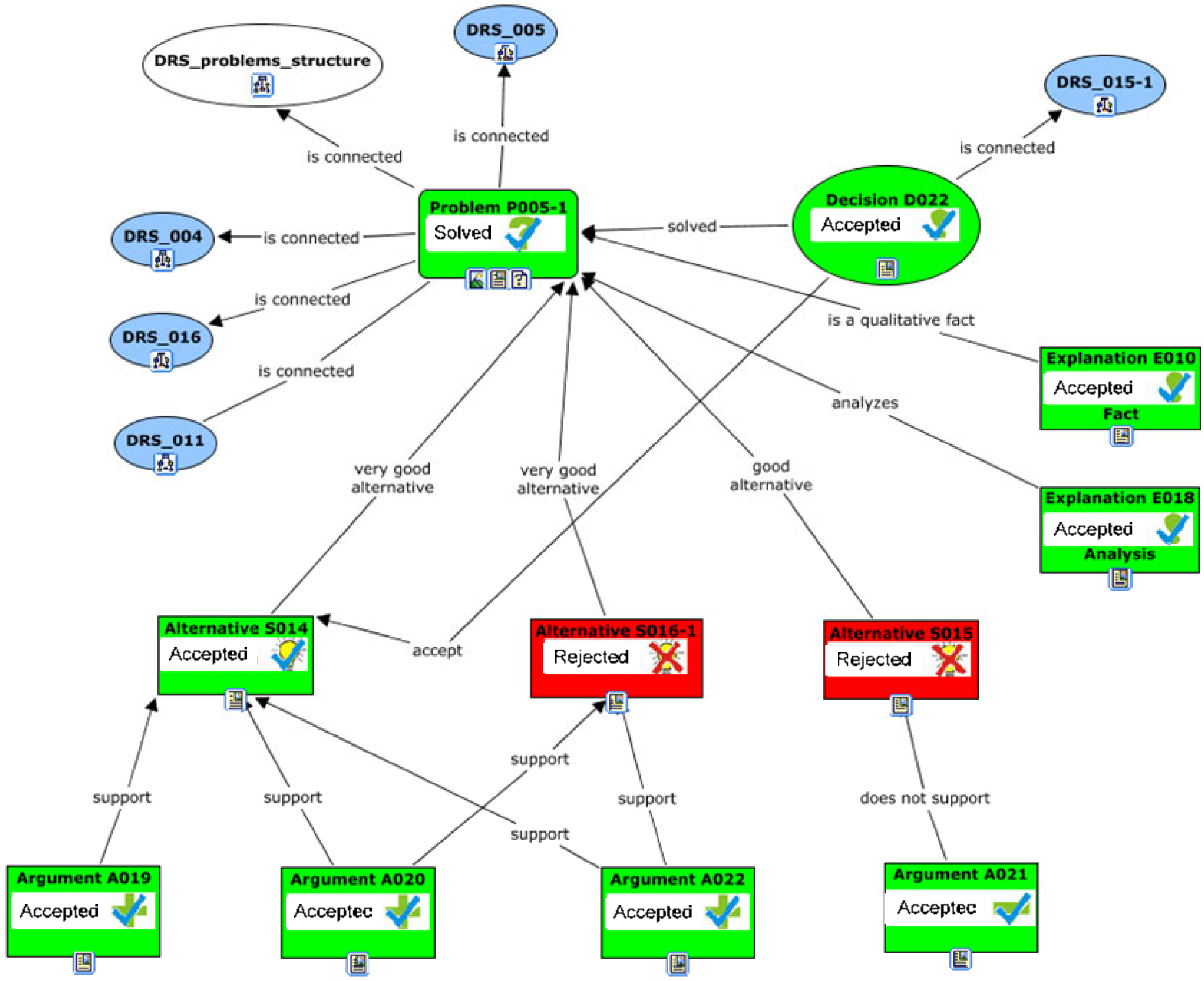

Fig. 3 Graphical representation of one *DRS*

Using the CmapTools V5.04 tools it was possible to link the DR elements with files which contain additional information, for example a graphical representation of the functional decomposition (Fig. 4, continuous line) or a table with attributes of the DR element (Fig. 4, dashed line). There are 69 information objects (IO) used in Report II.

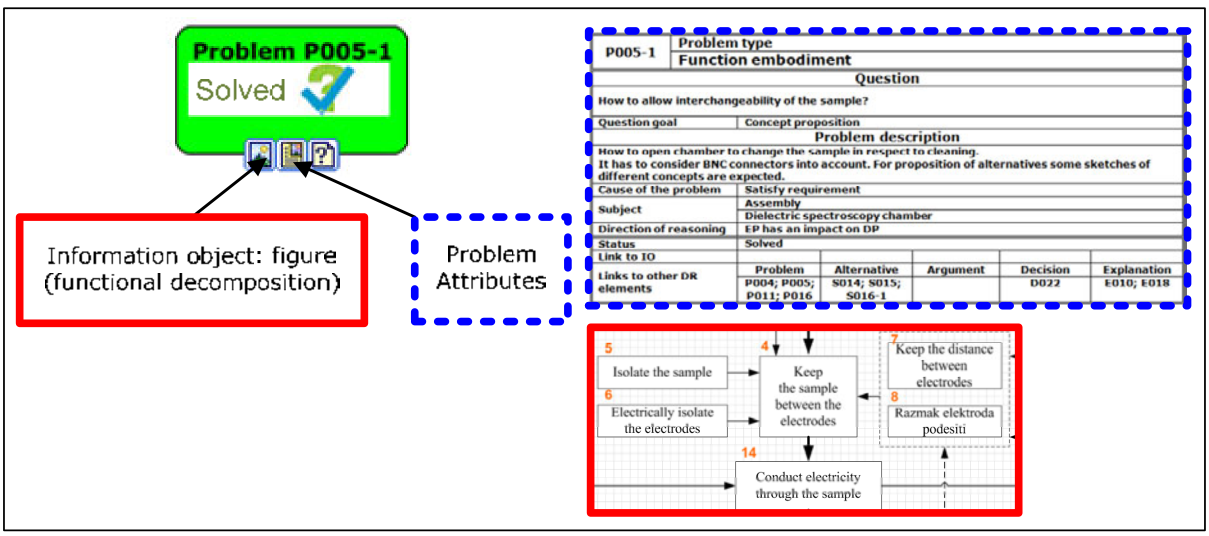

Fig. 4 *DR* element and its information objects

Besides the links between the DR elements of one DRS, these elements can be linked with the DR elements that are part of the other DRS (Fig. 5, dashed line). If the user, by using the cursor passes over the DR element in the DRS structure then the cloud with short information pops up (Fig. 5, continuous line). All elements in the DRS have a unique ID, e.g. P005, R049 (Fig. 5).

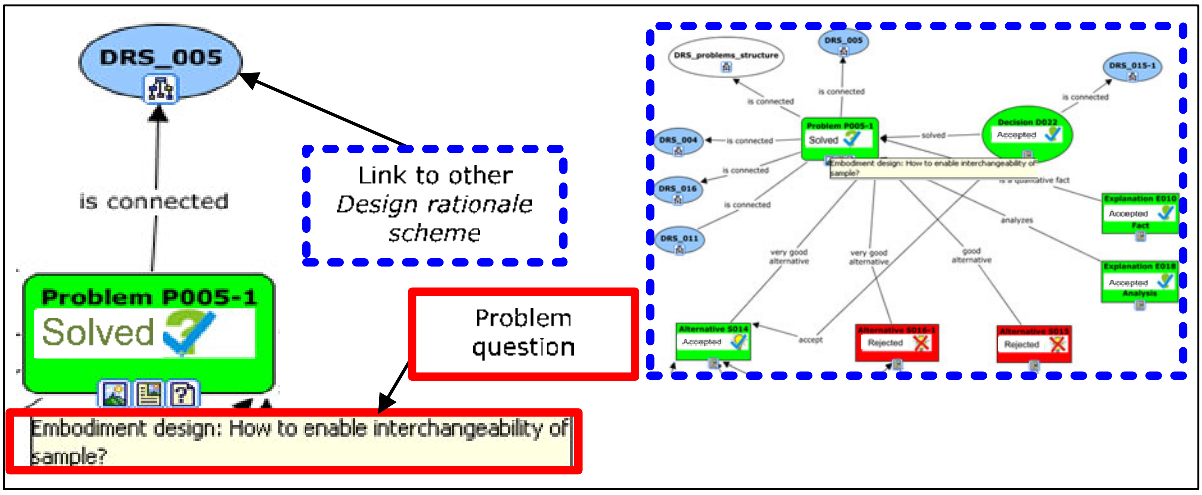

Fig. 5 The link between the problem and the other *DRS*

The comparison of the completeness and correctness of the answers offered by two groups (A and B) shows that Group A has given better results (Fig. 6). Although the mean value for completeness is 11% better for Group A, it can be seen that for questions 1, 2, 5 and 6, the results of Group B are better. This result is such because participants from Group B provided answers to the mentioned questions by using Report II. The mean value for correctness is 5% better for Group A, however, the results of Group B are not better for questions 5 and 6 although they used Report II. This can be explained by the fact that for questions 5 and 6 Group B suggested more entities (in the given answer), but some of them were wrong (Fig. 6).

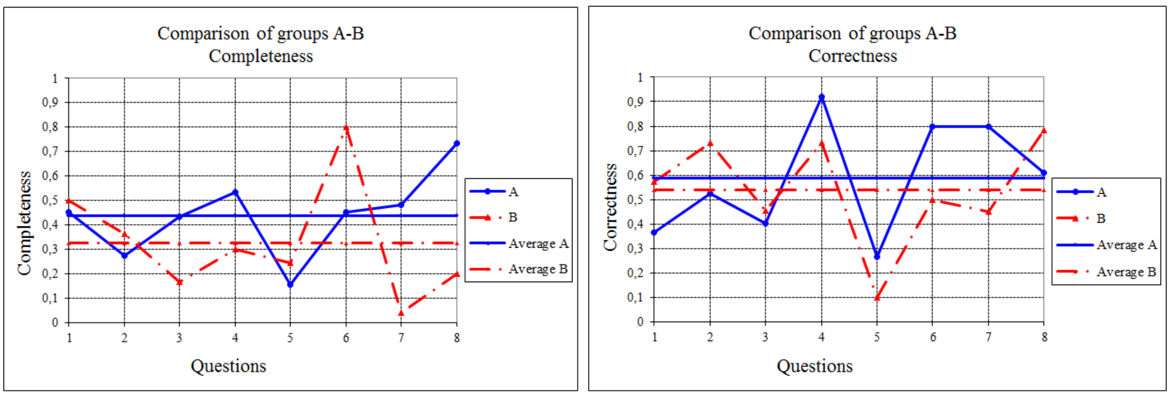

Fig. 6 The comparison of completeness and correctness of answers for two groups (A and B)

The analysis of the experimental results gives a comparison between completeness and correctness of answers to all questions (Fig. 7). It can be seen that the mean value for completeness is 26% and for correctness it is 16% better when participants used Report II.

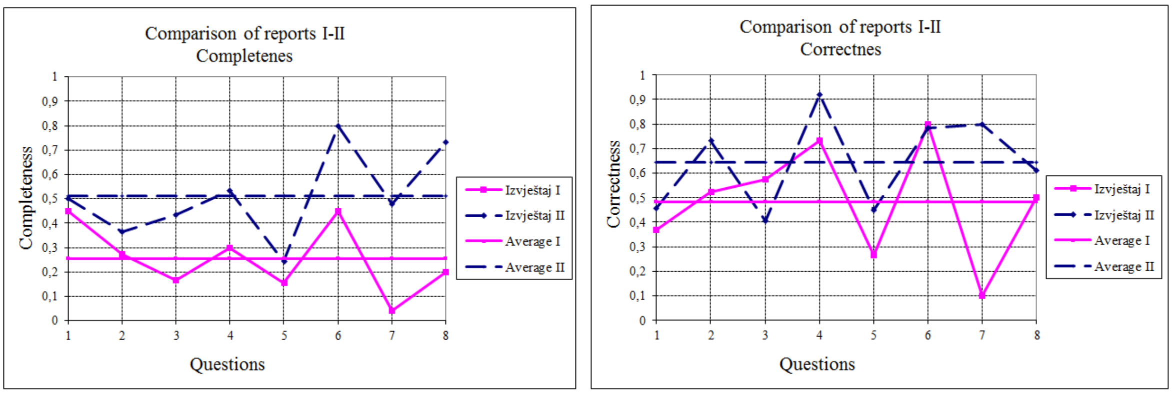

Fig. 7 The comparison between completeness and correctness of answers using two reports (I and II)

7. Conclusion

This paper reveals our motivation for conducting the research. In order to deal with the primary aim, the research area and methodology are outlined. A review of the literature identified key concepts and their relationships required for understanding the issues related to design rationale capturing. The proposed design rationale capturing model is presented as a result of the classification of explanations and arguments related to the corresponding design problem solutions and decisions that could be used in the conceptual, embodiment and detail design stages. Most of these items of information are not documented in practice although they are essential for the decision process in product development within a particular context. Some of the *DR* systems proposed in the literature predict automated collection, storage and retrieval of information and knowledge. However, due to the informal nature of information and knowledge, which are the subject of this study, there is a need to involve designers in the process of knowledge management within the organisation.

Focusing on the embodiment design stage, the fundamental logic of the IBIS system was expanded with some new elements and attributes. The design rationale scheme incorporates five elements: design problem, alternative solutions, arguments, explanations and decisions. The proposed model has been evaluated by an experiment. In the experiment, the items of information gathered during the process of developing a chamber for dielectric spectroscopy of aqueous solutions were used.

In the first part of this research, we investigated how the people involved in the product development (designers) were motivated to capture knowledge and how much time they were ready to devote to that activity. It is obvious that this extra work is not stimulating, but through conversations with them, we found that by applying certain measures desired cooperation could be achieved. Such findings primarily relate to the characteristics of the future knowledge capturing system that has to tackle the needs of users (designers), but employees need to be stimulated with some reward, too.

The comparison between completeness and correctness of the answers which were offered by two groups of interviewees showed that the defined elements and attributes of the design reasoning facilitate the understanding of the proposed design rationale capturing model and provide a basis for effective search of information. Regardless of the good experimental results it is necessary to implement the proposed model into the environment of modern tools that support the product development process.

Although capturing a significant amount of information into some kind of product data management (PDM) or product lifecycle management (PLM) system will take time, it should be of immediate benefit as it provides designers with information (reasons, arguments, alternatives, explanations) which influenced in previous decisions. We believe that such an approach could offer the following advantages:

- the use of a PLM system database and search mechanisms should provide efficient and simple procedures for the retrieval of captured knowledge
- consistent and relatively well-known methods and procedures for resolving team collaboration issues in simultaneous work and the team sharing of design rationale scheme diagrams
- the use of the "standard" PLM mechanism for creation and maintenance of external links to all kinds of product documentation
- possibilities to develop sets of "predefined" elements and/or queries that could make design rationale capturing and retrieval processes easier and faster

Of course, the mentioned *advantages* must be measured and validated by experimental usage of such a system in design practice. In parallel with practical validation, future work will be focused on some unresolved issues and dilemmas that could be sources of problems in practical usage, for example: how deep is it necessary to use "standard" PLM document management mechanisms – a compromise must be reached between potential benefits and the amount of extra designers' work.

We hope that the presented approach could contribute to research efforts in design rationale capturing tools as well as in the development of a new generation of PLM systems whose application should be extended to broader contexts and domains outside the standard design documentation which serves as a description of the final design only.

REFERENCES

[1] Hubka, V.: Theorie der Konstruktionsprozesse, Berlin, Heidelberg, New York: Springer - Verlag, 1976.

[2] Burge, J., Brown, D. C.: Integrating Design Rationale with a Process Model, Workshop on Design Process Modelling, Artificial Intelligence in Design '02, Cambridge, UK, 2002.

[3] Conway, A. P.: State of Art Review - Approaches to Design Information Capture, University of Strathclyde, 2006.

[4] Ahmed, S.: Encouraging reuse of design knowledge: a method to index knowledge, Elsevier Science, Design Studies 26, 565–592, 2005.

[5] McAlpine, H., Hicks, B. J., Huet, G., Culley, S. J.: An investigation into the use and content of the engineer's logbook, Elsevier Science, Design Studies 27, 481–504, 2006.

[6] Lee, J.: Design rationale systems: understanding the issues, IEEE Expert: Intelligent Systems and Their Applications 12(3), 78–85, 1997.

[7] Myers, K. L., Zumel, N.B., Garcia, P.: Acquiring Design Rationale Automatically, Artificial Intelligence for Engineering Design, Analysis and Manufacturing, Volume 14 Issue 2, Cambridge University Press, 2000.

[8] Ishino, Y., Jin, Y.: An information value based approach to design procedure capture, Advanced Engineering Informatics 20, 89–107, 2006.

[9] Hu, X., Pang, J., Pang, Y., Atwood, M., Sun, W., Regli, W. C., A survey on design rationale: representation, capture and retrieval, ASME Design engineering technical conferences, Baltimore, Maryland, 2000.

[10] McCall, R. J.: PHI: A Conceptual Foundation for Design Hypermedia, Design Studies 12 (1), 30–41, 1991.

[11] Maclean, A., Young, RM., Bellotti, V. M. E., Moran, T., Questions, Options, and Criteria: Elements of Design Space Analysis, in Moran, T., Carroll, J., Design Rationale Concepts, Techniques, and Use, Lawrence Erlbaum Associates, pp. 53-106, 1996.

[12] Lee, J., Lai, K.: What's in design rationale, Human-Comput. Interaction, 6, pp. 251–280, 1991.

[13] Burge J. E., Bracewell R.: Guest editorial of special issue: Design Rationale, Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 22, 309–310, 2008.

[14] Bracewell R., Gourtovaia M., Moss M., Knott D., Wallace K., Clarkson J. P.: Dred 2.0: A Method and Tool for Capture and Communication of Design Knowledge Deliberated in the Creation of Technical Products, Proceedings of the 17th Int. Conf. on Engineering Design – ICED 09, Stanford University, Stanford, USA, 2009.

[15] Aurisicchio M., Gourtovaia M., Bracewell R. H. and Wallace K. M.: Evaluation of how DRed design rationale is interpreted, Proceedings of the 16th Int. Conf. on Eng. Design – ICED 07, Paris, 2007.

[16] Buckingham Shum S. et al., From gIBIS to MEMETIC Evolving a Research Vision into a Practical Tool, Design Rationale Workshop: Design, Computing & Cognition Conference, Eindhoven, 2006.

[17] Hicks, B. J., Culley, S. J., Allen, R. D., Mullineux, G.: A framework for the requirements of capturing, storing and reusing information and knowledge in engineering design, International Journal of Information Management 22, 263–280, 2002.

[18] Roozenburg, N. F. M., Eekels, J.: Product Design: Fundamentals and Methods, ed. N. Cross and N. Roozenburg, John Wiley & Sons, Chichester, 1995.

[19] Blessing, L. T. M., Chakrabarti, A.: DRM, a Design Research Methodology, Springer-Verlag London Limited, 2009.

[20] Aurisicchio, M.: Characterising Information Acquisition in Engineering Design, Thesis, Cambridge University Engineering Department, 2005.

[21] Rohde, D., Pavković, N., Aurisicchio, M.: Design Knowledge Indexing in Relational Database, International Design Conference – DESIGN 2008, Dubrovnik – Cavtat, 2008.

[22] Rohde, D.: Embodiment design rationale, Thesis, University of Zagreb, Faculty of mechanical engineering and naval architecture, Zagreb, 2011.

[23] N. N.: VDI guidelines – VDI 2225, http://www.vdi.de/index.php?id=301, 05.06.2009.

[24] Štorga, M., Darlington, M., Culely, S., Marjanović, D.: Traceability of the development of 'information objects in the engineering design process, Research into Design - Supporting Multiple facets of Product Development / Chakarabarti, Amaresh (ur.), ICoRD'09 Bangalore, Research Publishing, India, 263-270, 2009.

Submitted: 26.6.2014
Accepted: 04.3.2015

Danijel Rohde
Danijel.Rohde@inetec.hr
INETEC – Institut for Nuclear Technology
Dolenica 28, 10250 Zagreb
Mario Štorga
Dorian Marjanović
University of Zagreb
Faculty of Mechanical Engineering and Naval Architecture
Ivana Lučića 5, 10002 Zagreb