Consolidation of Methods for Visualization and Management of Engineering Design Data Sets

Neven PAVKOVIĆ, Tomislav MARTINEC, Dorian MARJANOVIĆ

Abstract: Existing design support tools and methods struggle to deal with numerous data management situations that arise in medium and large-scale design projects. Authors analyse and compare the suitability and efficiency of several methods for managing and visualization of relationships between engineering objects and propose guidelines for building a framework/interface for consolidated usage of those methods in industrial practice. The proposed matrix-based consolidation methodology builds on findings gathered from two monitored design projects as well as previous research in areas of traceability and visualization. For engineering objects (data sets) that belong to one or two domains which are not hierarchically structured it is argued that a simple matrix interface is sufficient to record the relations. Furthermore, if the engineering objects are hierarchically structured the browser interface is suggested. In the most complex situation, where objects belong to multiple different domains the use of diagrams and digital white boards is suggested.

Keywords: design rationale; engineering data management; engineering data visualization; matrix-style interface; multiple-domain matrix

1 INTRODUCTION

Complexity of product development processes is permanently increasing, especially from the viewpoint of quantity and interconnectivity of engineering data sets that have to be managed. In medium and large-scale design projects many data management situations arise that the existing design support tools and methods struggle to deal with. Design communication in large teams has to take place within the framework(s) of networks of complex dependencies (relationships) between engineering objects of various domains that are very difficult and expensive to be efficiently managed [1]. The aims of this paper are: (1) to analyse and compare the suitability and efficiency of several methods for managing and visualization of relationships between engineering objects and (2) to investigate and propose guidelines for building a framework/interface for consolidated usage of those methods in industrial practice.

Based on organized information structures, modern visualization means provide possibility to interactively navigate and uncover the information engineers are looking for [2]. While searching for information it is often presumed that users are being unaware of the precise location of information where the information can be obtained or possesses incomplete specification regarding the information necessary to perform search.

Both cases might arise in the product development of the complex technical systems involving large data and information sets and multitude of stakeholders generating and interpreting information. In this paper we treat engineering design information and data sets as engineering objects (EO). We consider EO as an instance of any kind of document, physical component or an "abstract" notion that arises or exists in product development process. For instance these may be: CAD part, calculation, detailed design parameter, requirement, function, partial solution, testing report, production specification, etc. Research presented in the paper has been focused on two issues concerning engineering data management in industrial practice: (1) improving the quality of design communication (building shared understanding) during the NPD process and (2) improving the understanding of past design projects when previous solutions and knowledge have to be reused.

A large number of engineering objects belonging to many different domains exist in any sociotechnical system on levels of granularity that satisfy practical needs [1, 3]. Therefore a compromise subset of relationship network must be extracted resulting with an acceptable cost/benefit ratio in terms of efforts spent on relation recording, management and visualization of relation network.

Such approach leads to the following research questions:
- Concerning the communication improvement, shared understanding and knowledge reuse, which relations between EOs should be of primary concern in the design projects?
- Which method(s) of management and visualization are most suitable for each particular detected subset of beneficial relations?
- How to efficiently consolidate and merge the usage of several most "popular" visualization methods in industrial practice through one common interface/framework?

Here we argue that a visualized network of semantic relationships that exist in the life cycle of engineering objects could help engineering designers to better understand the structure and the evolution of complex engineering data sets. In [4] we argue that diagrams are convenient for both fast recording and retrieving of particular tracing context on design episode level, and we consider diagram networks as the promising basis for EO relation network visualization on the project level. A similar approach verified in design practice is proposed by Aurisicchio and Bracewell [5]. Improvements of visualization and relationship management methods as well as their consolidation should enable designers to find the right information easier and quicker and to reuse the stored information in the right context.

The application of the described research approach is illustrated with a detailed presentation of two monitored projects (case studies) where two different new mechatronic devices have been designed. Both devices were completely new products of high complexity, and they were designed by different design teams. These projects have been monitored in a company specialized for the development of devices for nuclear power plant examination and inspection. The company is medium-

sized according to EU definition, with approximately twenty engineers working in R&D office. Members of both design teams have been asked or suggested to use and compare several different tools and methods for creation and visualization of relations between engineering objects. Presented research has been focused on conceptual design phase where designers begin to generate first instances of engineering objects. In case of complex products many of these EOs are part of solution alternatives that have very complex structure and usually are not part of the official manufacturing documentation. We argue that visualization and early management of these data sets is very important for communication between members of design teams and for building of shared understanding.

2 BACKGROUND AND RELATED WORK
2.1 Communication and Information Flow in Large-Scale Projects

Eckert et al. [6] report on observations of how failure to achieve appropriate information flow in large-scale engineering design processes contributes to a variety of problems for designers and decision-makers. The list of manifestations of inadequate information flow that have been observed in their studies was a significant contribution of cited paper. Based on their findings we have directed our approach towards solutions that might overcome some of the problems identified in their research.

Flanagan et al. [7] emphasize that individuals often have very little idea how their own tasks fit into the context of the wider product despite being experts in their own field and understanding the tasks of the people they work with frequently. They may have little idea where information is coming from or going to and who is using it in the wider product context. Failure to exchange knowledge efficiently is often a symptom of communications problems. Flanagan et al. [8] argue that the overview and experience of senior designers play an important part in supporting teamwork by coordinating activities and facilitating proactive communication across large project teams.

The paper of Eckert et al. [9] identifies three layers of structure in design communication, each of which can be more or less formal: the design process, the interaction between participants, and the representations of design information that are constructed and used. These layers can be formal across a spectrum from explicit rules to habitual conventions. Authors have analysed how formality affects design interaction in different situations and process contexts. They concluded that mismatches in the understanding of formality can lead to misunderstandings, in particular across expertise boundaries and between designers and their clients or customers.

2.2 Advances in Implementation of Matrix Based Representation of Relations

Several methodologies exist for dealing with data complexity in product design, including the application of graph theory and matrix-based approaches [10]. The large variety of matrix-based methods in engineering can be classified by the quantity of the types of elements involved. Whereas some approaches focus on the representation and analysis in between elements of the same type (e.g., dependencies within product components), others consider linkages between two types (e.g., dependencies between customer requirements and product functions) [11]. According to Lindemann et al. [10] three types of matrix systems may be identified: intra-domain, inter-domain and multi-domain. Intra-domain matrix describes relations between elements belonging to the same domain. A commonly applied approach of an intra-domain matrix is the Dependency Structure Matrix (DSM). Matrices containing elements belonging to different domains are referred to as inter-domain matrices. For example, components and functions of a product can be considered as elements belonging to two different domains [10]. Some applications make use of combinations of intra- and inter-domain matrices. Such an approach is called the Multiple-Domain Matrix (MDM) [10]. MDM is a square matrix comparable to a DSM containing system elements in identical order on both axes. In contrast to a DSM, different types of system elements are included and grouped in domains; the MDM can be subdivided into DSMs and DMMs (Domain Mapping Matrices) according to the inherent domains. The MDM possesses features of a common DSM; in fact, it represents a DSM on a higher level of abstraction: If the domains are considered as single elements, the areas of the DMM subsets represent the matrix cells that can store dependencies between these elements. Applying this logic, the areas of the DSM subsets are located on the matrix diagonal and can represent self-reflexive [10]. An approach of MDM usage, focused to creation of employee knowledge maps is proposed in [12].

As shown, matrix-based approaches are widely applied in complexity management. In this research such approach is used as the basic architecture of interface framework for consolidation of different relation visualization methods as described in chapter 5.3.

2.3 Information and Knowledge Reuse

Hicks et al. [13] emphasize that reuse of knowledge is frustrated by semantics. The knowledge may not be structured or specific and for the particular application may require an altered perspective. Here the individual must dynamically step from knowledge back to information and then generate another perspective which may provide knowledge for the new or unfamiliar situation. This new perspective is generated from semantic interpretation of the old perspective. McMahon et al. [14] introduce a user interface based approach to the browsing of hierarchically organized information entities that avoids problems of word or phrase search.

The knowledge reuse in design practice is often ad-hoc since designers consider the time and effort needed to locate the information and investigate information usefulness as too costly, often resulting in little or no attempt at reuse [15]. In the design environments where employees have little or no experience with knowledge management systems, designers often find those processes too obtrusive and time consuming, with small benefits for them.

2.4 Shared Understanding in Collaborative Design

Kleinsmann and Valkenburg [16] explored what the barriers and enablers are for the creation of shared understanding during a collaborative design process. Barriers and enablers are clustered according to their content – five clusters and a remainder category originated. All the clusters concern a different type of interface – an interface is an interaction pattern between actors. The authors found the following interfaces in the automotive case: between marketing and development, design team and suppliers, design team and company, market researcher and the market, and between software development and the design team. These findings emphasize the need for consolidation of methods for creation and support of shared understanding that might overcome the identified barriers. According to [17] knowledge integration is important in collaborative new product development. Mohan and Ramesh [18] define knowledge integration as the synthesis of specialized knowledge that is distributed across different artefacts and phases of product development into situation specific systemic knowledge.

2.5 Visualization of Engineering Data with Network of Diagrams

Efficient visualization (and manipulation) of large networks of relations is arguably the primary condition for successful management of large and complex engineering data sets in industrial practice. Diagrams augment cognition [19]. As such, a good diagram augments the capacity of the diagram's user to achieve goals. Visualization literally "makes visible" (or "evident") things that might not otherwise be so [19] - authors made a review of existing diagramming tools and they concluded that:

– Simplicity is important. The simpler the tool – even though its scope may be limited as a result – the easier it is to use, and the more likely users are to adopt it willingly and "naturally".
– Network hypergraphs are essential. The richly interrelated information elements typical in early designing are highly coupled, and representing those relationships is essential.
– Diagram layout is essential. A proper diagram layout can actually simplify it without loss of semantics.

Based on their findings the authors argue that there is no existent tool fully suitable to engineering design support purposes and that a new framework for diagramming tools must be developed.

Design rationale may be viewed as traceability of design thinking and decision process. Shipman and McCall [20] view design rationale as a topic that implies different things to different people, some describing it as the capture and potential reuse of normal communication about design. Aurisicchio and Bracewell [21] proposed a novel approach to designing and its documentation by integrated diagrams formalized into a templated structure and illustrated by means of a case study in the aerospace engineering industry. This line of research continued in [22] where authors emphasize a need for increased investment in flexible visual tools to aid human thinking.

3 METHODOLOGY

Methods and tools used in conducted case studies have been selected and developed based on findings from TRENIN project [23] and our several previously finished case studies [4, 24, 25]. This work is also grounded on approach of Aurisicchio and Bracewell [5] and on complexity management methods developed by Maurer [11] and Lindemann et al. [10].

3.1 Characterization of Relations and Engineering Objects

To answer the posted research questions we have started with distinguishing and characterizing relations and EOs from the following perspectives:
– Stability over time periods
– Level of granularity
– Quantity of EOs in a particular domain
– Differences between EO domains
– Cardinality of relations (binary or multiple)

Considering dynamic nature of changes during product development process, product structure and/or product architecture (or at least their elements) could be viewed as a relatively static data structure (on higher levels of granularity) for majority of engineering design environments (e.g. the reuse of mechatronic systems in automotive industry). Although product structures of complex products could contain large sets of EOs and relations, they do not change significantly over project life-cycle (on higher levels of granularity). Therefore, we assume that it could be cost-effective to build relation templates for such predominantly static structures. Such an approach could be considered as a semi-automated method, because engineers would reuse and update templates while generating sets of relations.

In a similar manner, we assume that the majority of relations between other EOs from different domains have a more dynamic character, however, probably smaller sets of EOs need to be related, thus requiring less effort and time consumption.

This reasoning is also supported with research of Spanoudakis and Zisman [26] where they have proposed a comprehensive relation classification.

3.2 Characterization of Recording and Visualization Methods

Different kinds of tools and methods for recording and visualization of engineering data relations that have been analysed during presented research could be systematized based on four interface concepts:
– **Browser** (explorer) of hierarchical structure(s),
– **Diagram** (graph) consisting of various classes of nodes and edges
– **Matrix** (one domain or multiple domains)
– **Digital whiteboard**

Enabling the same approach during creation and/or recording of a particular relation type between engineering objects for all four of above given concepts is among primary objectives of this research.

The domain to which engineering objects belong in large scale determines the selection of best interface concept for recording and visualization of relations.

Probably the simplest problem is the creation of relations in-between the content of the file system (when files are being treated as objects of one same domain). Using a browser (file explorer) interface, creation of relation network between files can result in a well-established traceability of project documentation. This type of interface requires a minimal change in users' computer usage habits, since such applications are used on a daily basis. The relations are being created by selecting explorer items and assigning a desirable relationship. Detailed explanation of the proposed browser interface concept can be found in [4].

Relations between file-system content are usually not sufficient to provide efficient and complete support to designers - management of complex engineering data requires creation of complex relationships between objects from multiple different domains. Therefore engineering objects should also represent abstract notions from various domains (requirements, functions, changes, design tasks), "physical" objects like elements of product structure (components), or persons – designers and other company employees. A step forward would be to use browser interface for relating (linking) files and other "non-file" objects represented as notions organized as taxonomy and/or ontology.

Generally, we want to explore a methodology where a combination of several recording and visualization methods will work in synergy to serve as a framework for generation of network of relations in-between all domains mentioned above. The structure of this network should not depend on the type of the interface used to store or retrieve the data from it.

One way to represent abstract notions appearing in various domains is to draw diagrams. Diagram interface enables the creation of nodes and edges that link the nodes. Every diagram node is an information container - this information could be either engineering objects or hyperlinks to other types of content.

Other way of intra- and inter-domain linking can be done via the matrix interface. Matrix-based approaches to complexity management are widely applied, so we decided to use them as the basic architecture of the relation creation and recording framework. If relations between objects belonging to the same type (domain) are examined, the corresponding matrices can be defined as intra-domain, e.g. commonly applied Dependency Structure Matrix (DSM), while matrices combining different objects belonging to different domains are referred to as inter-domain [10]. Some applications make use of combinations of intra- and inter-domain matrices. Such an approach is called the Multiple-Domain Matrix (MDM) [10].

The digital whiteboard (web and cloud-based collaborative tool) seems to be a very promising and useful tool for various kinds of supporting teamwork in large scale projects. It offers a number of features, such as drawing lines and shapes, adding sticky notes, marking up content, sharing images, documents and screenshots, discussions, pointer sharing, linking office documents, task assignments, and presentations of work.

3.3 Experiences and Findings from Previous Research Work

When conceiving presented research we were interested in how the proposed consolidation methodology of four recording/visualization methods and interfaces might overcome problems and issues identified in our previous research.

Research conducted in [24] has been focused to visualization of indexing structures used to relate engineering design taxonomy and knowledge captured during design projects. Based on findings from this previous case study we argue that the following strategy is essential: When a need for information and/or knowledge reuse occurs, the interface and procedures for searching and/or tracing should rely on the same taxonomy/ontology visualization and navigation methods as it was for initial indexing. Similar line of reasoning is presented in Ahmed and Wallace [27] and Ahmed and Storga [28] came to the same conclusions. Such an approach is important for better and easier understanding of search context and for providing interface efficient and user-friendly enough to be unobtrusive (as possible) to designer overloaded with tasks and complex software tools. However, taxonomy/ontology based indexing implies problems that may occur in process of initial definition of entities and hierarchical structure of elements (entities) for practical usage in certain environment. It is very difficult to propose one common taxonomy structure that will equally suit the needs of all participants in particular product development process. Which notions (entities) should be on the top level(s)? This positioning directly influences the amount of time user needs for indexing and searching processes. Various stakeholders in PD process have different focal interests, implying different views on taxonomy structure.

In another previously conducted case study [25] we have tracked a development process of complex mechatronic product - the vehicle control unit (VCU) for new generation of regional train. VCU is responsible for control, measuring, sequencing, protection, supervision and communication tasks in the whole vehicle. For this research we have selected an approach where identified product development process structure, documentation as well as product hardware and software components have all been mapped to subsets of notions from proposed ontology. Findings from this case study showed us that besides mapping EOs to ontology, it is even more important to establish relationships between engineering objects, as well as reference lists for object contents. In other words, focusing only on information (digital) objects traceability is not sufficient to provide a complete and appropriate support for design communication and/or information reuse.

4 PREPARATION AND ARRANGEMENT OF CASE STUDIES

In cooperation with medium-sized, development focused company we have prepared and conducted two case studies where we have been primarily focused to examine (explore) cost/benefit ratios for creation and

recording different kinds of relations between EOs from different domains. Based on conclusions from our previous research the case studies presented in this paper have been focused to development of common interface for consolidation of various relation recording and visualization methods. We have monitored and tracked two completely new design projects of mechatronic devices for underwater inspection of welds.

First monitored project lasted for three months, the team has been composed of one experienced designer, one novice designer and one experienced IT specialist. They were asked to use a variety of previously described methods and tools to record and visualize relations between EOs. The design team extracted EO domains and sets of relations they considered to be of primary interest for design communication, for building a shared understanding of project on company level as well as for reuse in the future similar projects. Five EO domains have been dominant (Fig. 1): digital documents (files), requirements, functions, components and people (from the viewpoint of responsibility). When there was a need identified to create relations between particular domains, all four kinds of interfaces were discussed and the most convenient one was selected and used. Here we must emphasize that the cooperating company does not use any kind of commercial PLM system, just internally developed web-based production data management system.

The second monitored project has been done with the same industrial partner company, but this time the design team consisted of 35 students that formed 5 teams – actually this project has been conducted as one semester project based learning course [29, 30]. This course, named European Global Product Realization (EGPR), and its teaching methodology have been continuously developed since 2000 [31]. Student teams were composed from 4 participating universities (from different European countries) including 5 teachers as team "coaches". Each team included students from every university – that way we got multidisciplinary, multinational and geographically distributed teams. All team members were master students having experience in several student design projects. The device being designed was very complex, therefore each team has been responsible for only one or two subsystems. During the first phase of this project all participants faced several frustrating situations that aroused due to project complexity. Scanned handmade sketches prepared by students often were of insufficient quality – they were not understandable, even with textual explanations. Those sketches were also dispersed through several files – it was difficult to merge them and get an overview of whole product (concept) functionality. This was especially difficult for teachers and company representatives because often (when giving feedback) they were not able to properly decide whether the solution proposals were good or not. These problems have been much more noticeable in comparison to all previous EGPR projects and began to ruin (impair) the student motivation. In previous EGPR projects students were developing less complex products, but in this project there were many situations where nobody except the author himself has been able to completely understand how a proposed conceptual solution is supposed to work.

To overcome these significant problems we have decided to use the digital whiteboard tool.

Introduction of digital whiteboard proved to be very successful. Everyday communication has been significantly improved - everyone could be updated about new developments on one common place. Instead of scanned handmade sketches students then began to intensively use screenshots of 3D models paired with comments, messages and short discussions (Fig. 9). It has been proven that students' skills in developing rough 3D models give much better results comparing to handmade sketches of relatively complex subassemblies. Graphical content displayed on the digital whiteboard has also been intensively used in videoconferencing meetings where it strongly contributed in generating and understanding new ideas as well as correcting mistakes and improving shown solutions.

Simultaneously we have researched how to integrate digital whiteboard with other three interface concepts initially developed in previous research projects and improved and tested during the first monitored project. Additionally, due to project complexity and number of participants this was also the opportunity to give focus to Issue Based Information System (IBIS) diagrams [32] for design rationale capturing and to investigate their integration and consolidation with other methods. Therefore one of the students has been assigned the role of "knowledge manager" with the task to capture and record all design rationale created during the whole project. For this purpose IBIS diagrams have been used and stored together with the rest of the project documentation on a common cloud service. Hyperlinks were used to connect IBIS diagrams with the digital whiteboard content and vice versa. In that way whiteboard could be expanded as a diagram canvas.

The first monitored design project has started from 24 requirements resulting with 68 functions. 276 designed components have been documented with about 1000 files including CAD files, text files, spreadsheets, catalogues, etc. Team members created 15 IBIS diagrams, 4 diagrams showing function structure and 1 function analysis diagram.

The second monitored project was more extensive in terms of outputs. Development teams brought together 47 requirements. Final design was embodied with 788 components and documented with over 3000 documents. Knowledge manager created a total of 58 IBIS diagrams, few matrices, function structure and FAD diagrams and recorded over hundred file-to-file relations. Students, teachers and designers from partner company filled two digital whiteboards each of 50 m^2 canvas size.

5 CONSOLIDATION OF EMBEDDED VISUALIZATION METHODS USING MATRIX-STYLE INTERFACE

During the first monitored project we have developed an initial concept of interface framework that should consolidate the usage of different kinds of embedded recording and visualization methods. The interface framework is conceived as matrix-style, where a major part (left of dotted line) is a concept similar to multiple domain matrix concept (Fig. 1). The domains identified by members of design team to be of primary interest for

recording and visualization are included in the proposal. Matrix cells on such interface concept represent a set of relations between at least one pair of domains. Diagonal cells represent relations that exist between objects of the same domain, while the other cells represent relations of objects belonging to different domains. We consider such concept as very flexible since any particular design environment could build and adapt domains and relations in the initial matrix according to its own needs.

The main idea of the initial matrix interface is to provide common simple and "naturally" visualized set of entrance points to particular tools and methods for relation recording and visualization. These entrance points are embedded inside the cells of the corresponding row and column. For each of the proposed cells (representing class of relationships) we have analysed which of the four recording/visualization methods is most convenient from the viewpoints discussed in sections 2.1 and 2.2. By selecting entrance point the user is further guided to recording and visualization interface(s) that has been evaluated as the most convenient. For example, if a field representing relations in-between the documents domain is selected (mark A in Fig. 1), a browser interface is opened, where selecting documents through file explorer creates relations. Furthermore, some of the fields in the same way lead to matrix (mark B), some to diagram interfaces (marks C, D, E), and the others to digital whiteboard interfaces (mark F). If multiple diagrams or matrix records have been made for a particular domain pair, their references are all stored inside the corresponding field, and the user selects which record to open when information is being retrieved (e.g. multiple design rationale diagrams embedded inside the field named "functions that imply design problems").
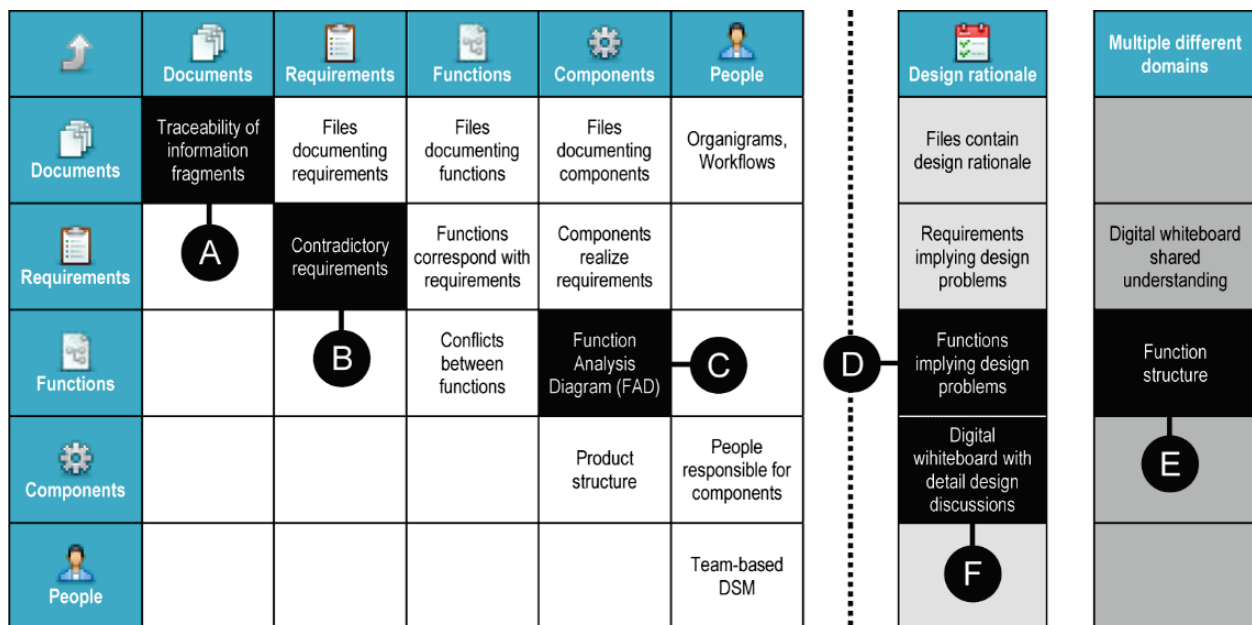

Figure 1 A concept of matrix-style consolidation interface

The part of the matrix-style interface that is on the left side of the dotted line in Fig. 1 could be considered and further developed similarly to multiple domain matrix concept where each cell represents the relationship between two domains. On the right side of the dotted line more complex relations are represented where objects (engineering data sets) from multiple different domains are related to each other in many different ways. Typical example is the function structure diagram where functions, energy, material and information (signals) are related in a very complex way. In presented research we have been focused to design rationale capturing, therefore a column with this domain is emphasised in Fig. 1.

Considering the digital whiteboard, a rich set of features such a tool offers gives plenty of opportunities to visualize relations between multiple different domains. On the "right side" of dotted line we see a huge open space for further research work on consolidation and proposal of new visualization methods. Development of tools able to integrate various visualizations in multiple domains might be a great challenge.

In the following four sections, each of the four proposed visualization methods and tools will be described in more detail together with examples of usage for particular situations.

5.1 Browser/Explorer Visualization Method

The browser interface serves as a file explorer (Fig. 2). Users can browse the computer/server file system and create relations between computer-stored files and folders. After it is assigned, the relationship is emphasized through the change of icons of the related objects. Additional information about the objects and their relations can be retrieved by selecting the corresponding explorer item. Besides that, browser objects can be enriched with information such as statuses, comments or design rationale. The development of the interface was started mainly to integrate diagrams into project documentation by allowing users to manually relate diagrams with computer-stored files and display these links in the browser interface. The interface was further upgraded and used as a file-to-file relating tool. Detailed explanation of the proposed browser interface concept can be found in [4].

During monitored projects browser/explorer method was used to relate objects from hierarchically structured domains such as computer file system and product structure (mark A in Fig. 1). These objects reflect project documentation involving all stored project outputs, including the records created using the proposed interface framework.
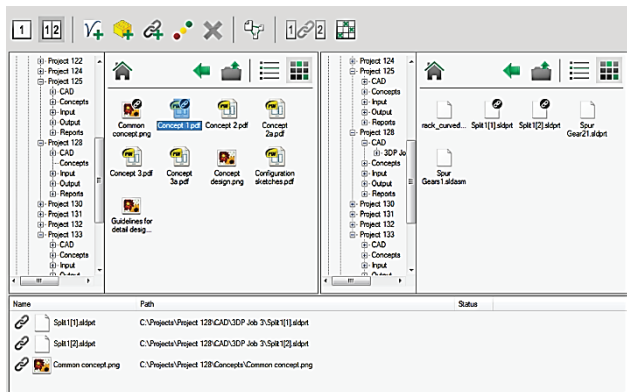

Figure 2 The browser interface developed for relating hierarchically structured objects

Browsing is the most efficient method to access objects that exist in the same domain and are already hierarchically structured. Such hierarchical structures are too complex to be presented in the form of a matrix or a diagram. Once the source and target objects are located browsing the structure, they can be selected and assigned with a relation (e.g. two project documents, or CAD file to IBIS diagram).

5.2 Diagrams as Visualization Methods

For engineering objects involved in multiple complex relations (both intra- and inter-domain) we argue that the most convenient technique is to create and visualize such a network of relations directly by drawing diagrams using the specific diagram methodology and interface tool. Information displayed in computer-drawn diagrams is stored within nodes and links, making them information containers. Nodes represent the objects and edges represent the relations. Every created diagram, no matter which diagramming tool was used, is a record of objects and their relationships, and contributes to overall relation network on a project level. In contrast to browser which does not require learning of new tools, diagrams excel other interfaces in terms of augmenting cognition, as a good diagram augments the capacity of its user to achieve goals.

A wide range of data can be stored, including digital entities storage information, such as hyperlinks to computer-stored files and other diagrams. In this way diagrams can relate engineering objects from both project documentation domain and abstract notion domains. We tried to integrate such diagrams (function structure, Function analysis diagram (FAD), IBIS, etc.) with the proposed matrix style interface concept in a way that would not change their original concepts and building methodology.

There are many existent diagramming methods and techniques supporting some aspects of design process

whose use has been proven in engineering practice. In most cases such diagrams include engineering objects from several different domains where each object could be related in various ways to several objects from other domains and vice-versa. One example of such diagram with relatively simple relations is Function Analysis Diagram (FAD), a method for function analysis as a form-dependent product representation (mark C in Fig. 1). An example of a FAD diagram is shown in Fig. 3. The FAD, unlike the Function Tree and the Function Structure, represents functions together with the physical elements of a product [33]. The diagram consists of blocks used to represent product components or functions, and relations with a label used to represent either useful or harmful actions.
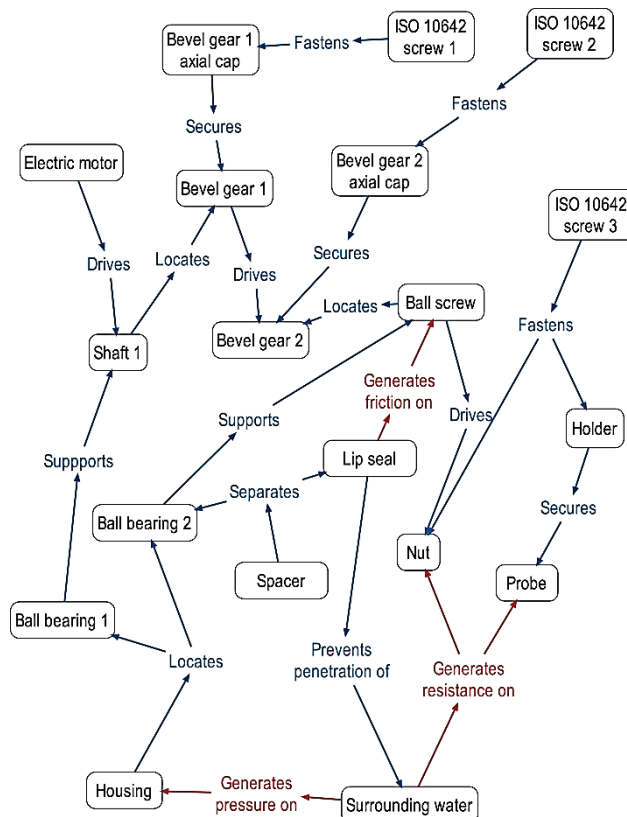
Figure 3 The Function Analysis Diagram (FAD) created using the general purpose diagram drawing tool

Another example is the Function Structure modelling (mark F in Fig. 1) where blocks are used to represent functions and energy/material/information flows. A fraction of created Function Structure diagram is shown in Fig. 4.

The research presented in [34] aims at contributing to a better understanding of the diverse functional modelling approaches proposed across disciplines. The gained insights lead to the identification of specific needs and opportunities, which could support the development of an integrated functional modelling approach. The findings suggest that there is no such shared sequence for functional modelling across disciplines. However, a shared functional modelling perspective has been identified across all reviewed disciplines, which could serve as a common basis for the development of an integrated functional modelling approach.
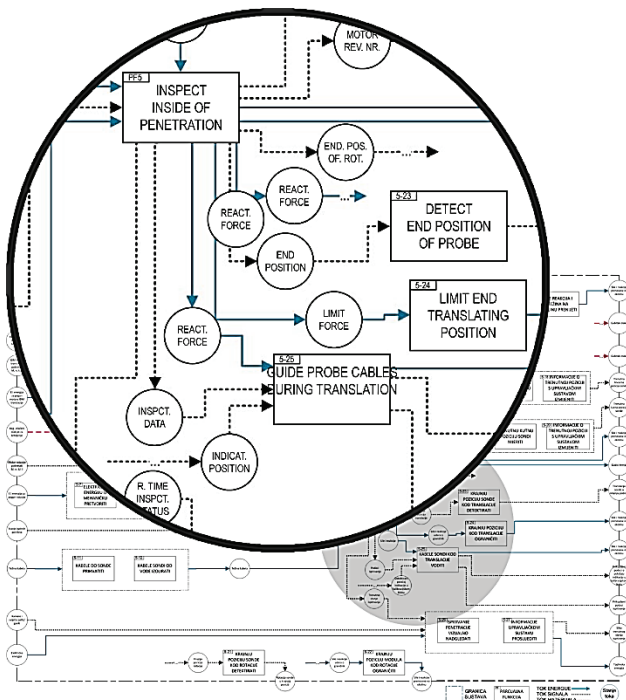
Figure 4 The Function Structure diagram created using the general purpose diagram drawing tool

The third (more complex example) is the IBIS diagram (mark D in Fig. 1). IBIS consists of a tree or directed graph, where nodes representing issues to be resolved, alternative solutions, and arguments in favour and against, are linked by arcs (Fig. 5) [32]. The method was progressively extended by Bracewell et al. [35] and Aurisicchio et al. [33] to support hyperlinks to other diagrams and computer-stored files assuring that engineering objects of any kind could be related in the IBIS diagram.

Figure 5 IBIS diagram created using the custom developed tool

A total of 15 IBIS diagrams were created during the first monitored project and the other 58 during the second monitored project. The network of interrelated IBIS diagrams created during the second monitored project is presented in Fig. 6. Each node represents one diagram. Three diagrams stand out as central points interconnected to a large number of other diagrams. These diagrams represent key issues that occurred during the project and raised a number of new, related issues.
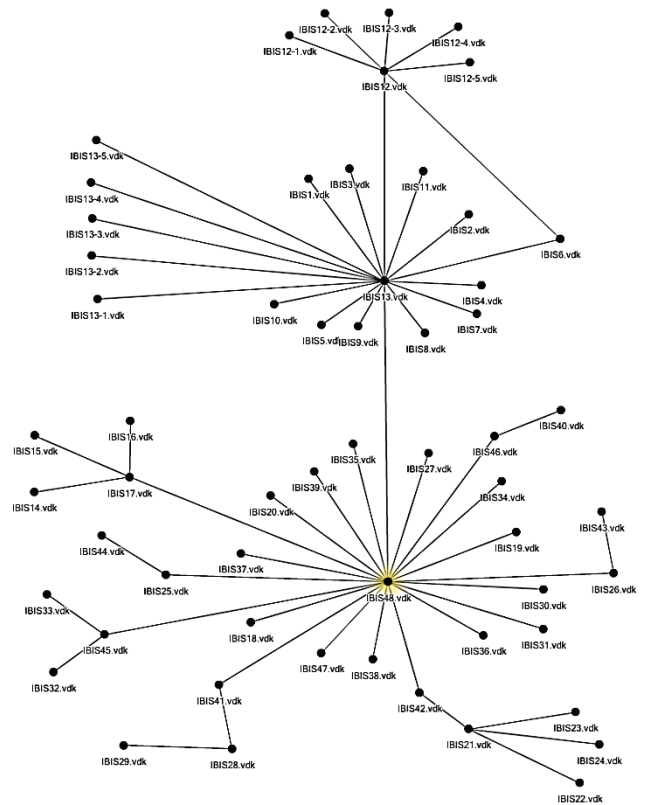
Figure 6 Network structure of IBIS diagram relations created in second monitored project

Relations recorded in IBIS diagrams can go beyond these networks. Besides diagram to diagram relations, it has been proven that it is very useful to include relations to other documents (EOs) in form of hyperlinks. This is shown in Fig. 7 where the issue node is related to the report document as an issue source. Also, some of the arguments are supported with relations to sketches and datasheets.
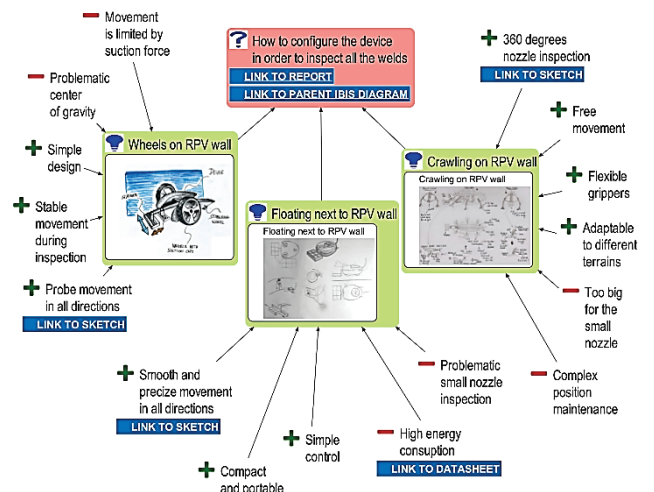
Figure 7 IBIS diagram related with other diagrams, reports, sketches and datasheets

Regarding this method of knowledge capturing and recording, in this research we have primarily been focused to explore the visualization and interlinkage issues and to comparison with "text based" reports as methods of knowledge recording. The student that has been given a role of knowledge manager has been deeply

involved in the project as simultaneously he has been given a role of a student advisor – teacher helper. He has been present at all project meetings and discussions and he has also been responsible for organization of communication between student teams. This way we tried to ensure that design rationale created by team members will be efficiently and correctly interpreted and recorded. Also teachers have helped him in structuring the diagrams and granularity of capturing.

5.3 Matrix as Relation Visualization Method

Matrix-based approaches to complexity management are widely applied. If the objects belonging to particular domain are not hierarchically structured, based on research presented in section 2.2 we argue that the simplest and the most convenient method to visualize their relationships is by using the matrix interface. In this research a matrix-based approach is used as the basic architecture of the relation creation (mapping) framework. Matrix-based methods used in engineering can be classified by the quantity of the types of objects involved [11]. Three types of matrices were used: intra-domain, inter-domain and multi-domain [10]. Following this it is obvious that the matrix interfaces used to relate objects of the same domain are in essence intra-domain matrices (or DSMs). Additionally, interfaces used to relate objects of two or more different domains are inter-domain (DMMs) and multi-domain matrices (MDMs). If the overall network of relations on project level is to be represented using the matrix interface it would be displayed as a large MDM.

Matrix interface also has potential of using various matrix analysis techniques (such as partitioning/ sequencing, clustering, banding and tearing), sensitivity analysis techniques and coupling techniques.

Presented intra-domain (DSM) example (Fig. 8) is the requirement conflict matrix (mark B in Fig. 1). Other examples are team-based and task-based DSMs, and e.g. the matrix which shows relations between functions (useful and harmful or conflicting). These matrices potentiate optimisation of people distribution, task disposition and product structure.

Figure 8 Matrix interface - an example of recording conflicts between requirements

An inter-domain (DMM) example is the set of relations between requirements and components, which shows the use of specific components requested through

requirement specification, or in other cases the components that satisfy requirements.

However, we found that there are several reasons to further develop and extend the current matrix models and tools, especially the process of manipulation with matrix – these issues are further discussed in section 6.

5.4 Digital Whiteboard Visualization Method

We have analysed a class of digital whiteboards that are web and cloud-based collaborative tools designed for large-scale distributed team projects. Such advanced digital whiteboards offer a number of features, such as sketching, drawing lines and shapes, adding sticky notes, marking up content, sharing images, documents and screenshots, discussions, pointer sharing, task assignment, and presentation of work.

Although at a glance very similar to diagrams, digital whiteboards put much more emphasis on discussion and explanations of solutions and building of shared understanding. Digital whiteboards are also a good representation of product evolution if implemented from the early stages - from the early concept sketches to the detail design.

The digital whiteboard tool proved to be very useful during the second monitored project, where a complex product was divided into several modules that were given to five teams to develop. Due to high interconnectivity between the modules, the teams needed a common platform to share and discuss the progress of each module. A digital whiteboard canvas was set up where they have put hand-drawn sketches and screenshots of CAD models. They would then discuss the design and define guidelines for the next steps of the overall design. This helped to build a shared understanding for all participants in project and in the same time record design rationale for decision makers and project evaluators.
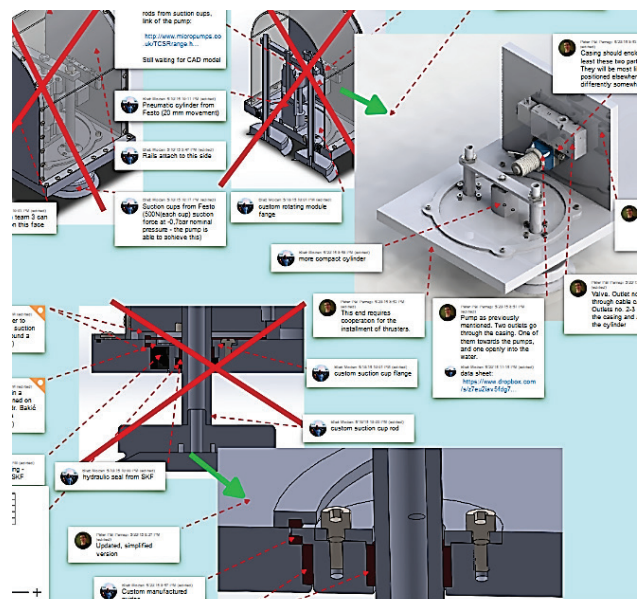

Figure 9 Digital whiteboard interface with example of segment of discussion on detail design

Apart from asynchronous work, whiteboard was intensively used on videoconferencing meetings where it strongly contributed in generating and understanding new

ideas as well as correcting mistakes and improving shown solutions. Quite wide available shared "area for displaying and discussing" also encouraged students to support the explanations of their proposals with a set of 3D CAD model screenshots rather than handmade sketches, especially in the late conceptual phase and during detailing. An interesting analysis and comparison between the role of computer-aided design (CAD) and sketching in engineering is presented in Veisz et al. [36].

The example in Fig. 9 represents a fraction of discussion on one of the subsystems. Here different participants in design process discuss and suggest improvements on the current state of design. This enabled to avoid some of the difficulties in the development of interfaces between different subsystems of designed product and to improve the combined performance of the device. During concept and detail design nineteen active users created two whiteboards and shared more than 300 screenshots and sketches, thus covering a total of 100 m^2 of whiteboard canvas.

6 FINDINGS AND DISCUSSION

This section summarizes and discusses the reasoning behind selection of most convenient recording and visualization interfaces for identified particular cases – represented by "consolidation" matrix cells (Fig.1). *The following answers to the second research question stated in section 1are mostly based on characterization of EOs, relations and domains' structure that were discussed in sections 3.1 and 3.2.*

- The simplest case for establishing relationships is when engineering objects (data sets) belong to one or two domains which are not hierarchically structured – it is argued that in such cases a simple matrix interface is sufficient to record the relations (Fig. 8). This proposal is actually very similar to commonly accepted DSM, DMM and MDM techniques. However, we found that further improvements are necessary to achieve better consolidation with other methods – e.g. developing approaches which would enrich the semantics of recorded relation. An example of such approach is presented in [37] - they have developed a method to improve systems engineering V (SE-V) process implementation by mapping multilevel data into design structure matrix (DSM) models. Authors have extended conventional DSM representation schema by incorporating multilevel test coverage data as vectors into the off-diagonal cells. These vectors provide a richer description of potential interactions between product architecture and SE-V integration test tasks than conventional domain mapping matrices.
- For engineering objects that are hierarchically structured (in their domains) we argue that the most convenient relation visualization method is the browser interface (Fig. 2). Examples are computer-stored files (project documentation) and product structure (components, modules, subassemblies, assemblies). Browsing is the most efficient method to access objects that exist in the same domain and are already in hierarchical relations. In both case studies this kind (class) of objects were the most numerous

one – consequently we faced the problems of manipulation and visualization of huge quantities of relations. This problem is further discussed in section 6.1.
- The most complex situation is dealing with relations that involve objects from multiple different domains. Instead of developing general method and tool for such problems we suggest to use already developed methodologies for particular cases – e.g. functional structure diagram, morphological matrix, FAD diagram, IBIS diagram etc. The main argument for such suggestion is the fact that industry is already familiar with these methods, therefore we assume that their usage could be easily accepted in practice if their original concepts and building methodology remain unchanged. It should not be expensive to develop simple tools for each kind of these diagrams - even some of general purpose diagram drawing tools could be used to create them. Still in most cases such diagrams include engineering objects from several different domains where each object could be related in various ways to several objects from other domains and vice-versa. That requires the development of links between objects from various kinds of diagrams and consequently a common naming and/or indexing of all objects whose relations are being visualized. This should be the further step of proposed consolidation.

6.1 Visualization Issues of Established Relation Networks

After a set of relations between objects and in-between different domains is established, users should be able to efficiently retrieve and expand the relationship data set. Since the relation establishment can be done through mentioned three types of interfaces (and additionally using digital whiteboard), the retrieval can be done the same way – by reading (visualizing) captured data in browser, matrices and diagrams. Although browsers and matrices are efficient and straightforward when it comes to capturing and recording relations between objects, when it comes to data retrieval and traces following, diagrammatic visualization showed up to be a better solution. Visual thinking, reasoning, and analytics emphasize the role of information visualization as the powerful medium for finding causality, forming hypotheses, and assessing available evidence [38].

Grounded on the previous work of information traceability visualization, large networks of interrelated records were created, where links between different files, diagrams, matrices and whiteboards allow users to cross boundaries of a single record and browse information spread through the network, by shifting from one interface to another. Relation network is automatically generated for the object selected in one of the interfaces, and represents nodes for each object that is in any way linked to the selected one, and edges for each of the relations [4].

The problem that needs to be tackled here is the filtering of the visualization results presented in form of a diagram. In general, displaying an entire large diagram may give an indication of the overall structure or a location within it, but makes it difficult to comprehend

[39]. While it is beneficial to give an insight to all the objects the selected object can be traced to, these kinds of representation can get highly complex and unclear (Fig. 10).

Figure 10 Visualization of the selected relevant part of complete relation network

Herman et al. [39] see the size of the graph (diagram) to view as a key issue in graph visualization. Besides complexity, large graphs pose several difficult problems such as performance compromise, viewing platform limits and discernment between nodes and edges, all resulting with poor usability issues [39]. The issues of usability and scalability are also mentioned as two of the ten unsolved information visualization problems by Chen [38]. Therefore, comprehension and detailed analysis of data in diagrams is easiest when the size of the displayed diagram is relatively small or in this case - when the displayed data is properly filtered. Filtering the represented objects and their relations (diagram nodes and edges) should reduce the overall diagram size and complexity, thus showing only the relevant data for a particular context. Both objects and relations have to be enriched with additional attributes that are processed by the filters. Regarding objects these attributes could be for example their domain, number of relations with other objects, or some other domain-specific properties. Relations can also largely affect the resulting diagram representation. First, one has to be able to assign different types of relations between objects. A very good general proposal based on overview of several approaches and types (dependency, evolution, satisfaction, overlap, generalisation/refinement, conflicting, rationalisation, contribution, etc.) could be found in [26]. Relation attributes such as direction, time/date or value could also be used to filter results.

If, on the other hand, the visualization of the overall relation network is done through the matrix interface, problems to be tackled stay practically the same - the matrix gets large, complex and unclear. The interface and the visualization capabilities of the tool that will manage such matrix have to provide the following mechanisms [40].

− Filtering on level of domains, and on level of rows and columns, enabling to hide/extract a set (combination) of rows and/or columns belonging to different domains or extracting several full domains.

Applying of filters should enable the user to extract and merge the areas of matrix that are of his current interest while working on matrix data. The extracted area should keep all the indicators of domains and particular rows and columns as they are visible on the whole matrix. Here by extracting we mean only visual extraction – the rest of the matrix is just being hidden.

− Extracting only the cells that have a symbol of relation from the set of selected (filtered) rows and columns (or domains).
− Efficient way of inserting/updating areas that are built and stored outside of the "main" matrix as predefined templates.
− Domain names and their instances (EOs) as well as the relationships should be based on specially developed ontology as proposed in [41] and [42].
− Layering / colouring schemes may also be beneficial in particular manipulation situations.

7 CONCLUSION AND FUTURE WORK

This paper proposes methods and interfaces for manual recording, management and visualization of relations between engineering objects from various domains during design process. The main contribution of the paper is the proposal of flexible interface framework for consolidated and integrated usage of different kinds of recording and visualization methods.

The consolidation proposal is based on findings gathered on two monitored complex design projects as well as long term experience and findings from several previous research projects in areas of traceability and visualization of engineering data. Based on thorough analyses and intensive discussions with designers in industrial practice the authors suggest the selection of most appropriate interface tools and methods for recording and visualization of relations between engineering objects depending on domains objects belong to. Methods and tools analysed in presented research can be divided into four basic interface concepts: browser, graph (diagram), matrix and digital whiteboard. An interface framework concept has been developed where these different methods work in synergy having an ultimate goal to create a network of relations in-between different domains. Interface framework is proposed as flexible multiple domain matrix-style area where each matrix cell leads to further procedure(s) involving browser, matrix or diagram methods/tools. *This interface framework proposal is an answer to the third research question stated in section 1.*

In order to achieve the synergy of proposed methods and tools, with ultimate goal to use the same notation for all engineering objects in single integrated network of visualization methods, the existing interfaces still have to be adapted and further developed. This implies reprogramming of used browsing, matrix and diagram tools in a way that they share the same data sets, so all engineering objects in common way would be available in each interface. Therefore it will be easier and faster to create relation records, and also a change made in one of the tools will affect the representation of objects and relations in other tools. The idea that has to be further

considered is to develop a visualization of common list of engineering objects that would be on disposal in the usage process of every particular relation recording tool. This direction of research should also be grounded on findings from Eckert et al. [9] where focus should be given on three layers of structure in design communication, each of which can be more or less formal: the design process, the interaction between participants, and the representations of design information that are constructed and used. *Therefore, as an answer to the first research question stated in section 1, we can conclude that relations between EOs within these three layers should be of primary concern in the visualization and management of engineering data sets.* The questions that have to be explored in this context are how to build acceptable explicit rules to use consolidated interface and how to include (or adapt to) habitual conventions of particular environment.

Other currently defined guidelines for further research issues are:

− A filtering approach is necessary to reduce the size of the overall relation network in order to overcome the issues of complexity, usability and scalability, thus providing user only the relevant data for a particular context.
− Methodology for matrix manipulation should be improved:
  − Huge number of EO relations in large-scale projects will generate large matrices, impossible to be manipulated and viewed as a whole – procedures and tools have to be developed that will enable hiding unnecessary areas and/or extracting and visualizing areas of current interest.
  − Semantics of relations should be added, additionally it would be beneficial if a cell would contain (or point to) more contents than just a mark of relation existence.
  − Mechanisms (procedures) for generating and inserting predefined templates of selected matrix areas should be developed and implemented.
− Both monitored design projects finished with detail design phase, in further research we should ask design teams to analyse the utility of recorded relations and their visualization in future similar design projects. Also it remains to capture and visualize objects and relations in process of prototype testing and requirement verification.
− Proposed framework has not been considered in the context of PLM environment - embedding and merging this proposal with PLM system has to be one of the major lines for further upgrades. In this phase of the research we have been focused to conceptual phase of product development. PLM systems lack the entities and modeling capabilities for the conceptual design, their use usually begins in the detail design phase when the product structure and components are determined. Merging of this proposal with PLM requires extending of PLM functionality with entities from conceptual domain as well as implementing the design methods, processes and workflows used in conceptual design – this seems to be a great

challenge. Similarly, in this phase of research we have neglected the issues of version management and access control. It has to be further investigated how important these issues are in conceptual design compared to detail design.

− Presented case studies have been conducted in a medium sized company. It would be necessary to repeat the study in a large company, e.g. in automotive sector, to explore and compare the differences in domains and collections of engineering objects as well as their characteristics.
− It should be verified if (and how) proposed concept of relations between domains (cells in matrix in Figure 1) contributes to building of shared understanding of design projects on the company level. These cells could be considered as predefined contexts for relation recording as well as guidance for information and knowledge retrieval.

Acknowledgements

8 REFERENCES

[1] Koehler, N., Naumann, T. & Vajna, S. (2014). Supporting the Modeling of Traceability Information. *Proceedings of the DESIGN 2014 13th International Design Conference* / Dubrovnik, 1811-1820.
[2] Visualizing Knowledge and Information: An Introduction. Knowledge and Information Visualization / Tergan, S.-O.; Keller, T. Springer Berlin Heidelberg, 2005, pp. 1-23. https://doi.org/10.1007/11510154_1
[3] Koenigs, S. F. Konzeption und Realisierung einer Methode zur templategestützten Systementwicklung. *Doctoral Dissertation,* Technical University of Berlin, Berlin, 2012.
[4] Martinec, T. & Pavkovic, N. (2014). Visualization of Information Traceability in Product Development. *Proceedings of the 13th International Design Conference (DESIGN 2014)* / Dubrovnik, 2014, 1831-1842.
[5] Aurisicchio, M. & Bracewell, R. (2013). Capturing an integrated design information space with a diagram based approach. *Journal of Engineering Design, 24*(6), 397-428. https://doi.org/10.1080/09544828.2012.757693
[6] Eckert, C., Clarkson, P. J. & Stacey, M. (2001). Information Flow in Engineering Companies: Problems and their Causes. *Design Management – Process and Information Issues, Proceedings of the 13th International Conference on Engineering Design (ICED'01)* / Glasgow, 2001, 43-50.
[7] Flanagan, T. L., Eckert, C. M. & Clarkson, P. J. (2003). Parameter trails. *Design Management, Proceedings of the*

14th International Conference on Engineering Design (ICED'03) / Stockholm, 2003.

[8] Flanagan, T., Eckert, C. & Clarkson, P. J. (2007). Externalizing tacit overview knowledge: A model-based approach to supporting design teams. *Artificial Intelligence for Engineering Design, Analysis, and Manufacturing, 21*(3), 227-242. https://doi.org/10.1017/S089006040700025X

[9] Eckert, C., Stacey, M. & Earl, C. (2013). Formality in design communication. *Artificial Intelligence for Engineering Design, Analysis, and Manufacturing, 27*(2), 91-103. https://doi.org/10.1017/S0890060413000073

[10] Lindemann, U., Maurer, M. & Braun, T. (2009). *Structural Complexity Management: An Approach for the Field of Product Design.* Springer-Verlag Berlin Heidelberg. https://doi.org/10.1007/978-3-540-87889-6

[11] Maurer, M. S. (2007). Structural Awareness in Complex Product Design. *Doctoral Dissertation*, Technical University of Munich, Munich, 2007.

[12] Wickel, M.; Schenkl, S.; Schmidt, D.; Hense, J.; Mandl, H.; Maurer, M. (2013). Knowledge structure maps based on Multiple Domain Matrices. InImpact: The Journal of Innovation Impact. 5, 1(2013), pp. 5-16.

[13] Hicks, B. J., Culley, S. J., Allen, R. D. & 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*(4), 263-280. https://doi.org/10.1016/S0268-4012(02)00012-9

[14] McMahon, C., Crossland, R., Lowe, A., Shah, T., Williams, J. S. & Culley, S. (2002). No zero match browsing of hierarchically categorized information entities. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 16*(3), 243-257. https://doi.org/10.1017/S0890060402163098

[15] Kim, S., Bracewell, R. H. & Wallace, K. M. (2007). Improving design reuse using context. *Design for X, Proceedings of the 16th International Conference on Engineering Design (ICED'07)* / Paris, 2007.

[16] Kleinsmann, M. & Valkenburg, R. (2008). Barriers and enablers for creating shared understanding in co-design projects. *Design Studies. 29*(4), 369-386. https://doi.org/10.1016/j.destud.2008.03.003

[17] Kleinsmann, M., Buijs, J. & Valkenburg, R. (2010). Understanding the complexity of knowledge integration in collaborative new product development teams: A case study. *Journal of Engineering and Technology Management, 27*(1-2), 20-32. https://doi.org/10.1016/j.jengtecman.2010.03.003

[18] Mohan, K. & Ramesh, B. (2007). Traceability-based knowledge integration in group decision and negotiation activities. *Decision Support Systems, 43*(3), 968-989. https://doi.org/10.1016/j.dss.2005.05.026

[19] Salustri, F. A., Eng, N. L. & Weerasinghe, J. S. (2008). Visualizing Information in the Early Stages of Engineering Design. *Computer-Aided Design and Applications, 5*(5), 697-714. https://doi.org/10.3722/cadaps.2008.697-714

[20] Shipman, F. M. & McCall, R. J. (1997). Integrating different perspectives on design rationale: Supporting the emergence of design rationale from design communication. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 11*(2), 141-154. https://doi.org/10.1017/S089006040000192X

[21] Aurisicchio, M. & Bracewell, R. (2009). Engineering Design by Integrated Diagrams. *Design Methods and Tools, Proceedings of the 17th International Conference on Engineering Design (ICED'09)* / Stanford, CA, 2009, 301-312.

[22] Eng, N., Aurisicchio, M., Bracewell, R. & Armstrong, G. (2012). Mapping for design decision support in industry. *24th International Conference on Design Theory and Methodology, ASME 2012 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* / Chicago, IL, 2012, 579-590. https://doi.org/10.1115/DETC2012-70959

[23] Marjanovic, D., Storga, M., Bojcetic, N., Pavkovic, N. & Stankovic, T. (2011). *EUREKA E4911 TRENIN Project Final Report.*

[24] Pavkovic, N., Bojcetic, N., Franic, L. & Marjanovic, D. (2011). Case studies to explore indexing issues in product design traceability framework. *Design Information and Knowledge, Proceedings of the 18th International Conference on Engineering Design (ICED'11)* / Copenhagen, 2011, 131-140.

[25] Pavkovic, N., Tecec Ribaric, Z. & Svilicic, T. (2012). Traceability Case Study on Rail Vehicle Control Unit Development Project. *Proceedings of the 12th International Design Conference (DESIGN 2012)* / Dubrovnik, 2012, 1567-1578.

[26] Spanoudakis, G. & Zisman, A. (2005). Software traceability: a roadmap. *Handbook of Software Engineering and Knowledge Engineering, 3*, 395-428. https://doi.org/10.1142/9789812775245_0014

[27] Ahmed, S. & Wallace, K. (2003). Indexing design knowledge based upon descriptions of design processes. *Design Process Research, Proceedings of the 14th International Conference on Engineering Design (ICED'03)* / Stockholm, 2003.

[28] Ahmed, S. & Storga, M. (2009). Merged ontology for engineering design: Contrasting empirical and theoretical approaches to develop engineering ontologies. *Artificial Intelligence for Engineering, Design, Analysis and Manufacturing, 23*(4), 391-407. https://doi.org/10.1017/S0890060409000146

[29] Zavbi, R. & Vukasinovic, N. (2014). A Concept of Academia-Industry Collaboration to Facilitate the Building of Technical and Professional Competencies in New Product Development. *International Journal of Engineering Education, 30*(6), 1562-1578.

[30] Zavbi R., Benedicic J. & Duhovnik J. (2010). Use of Mixed Academic-Industrial Teams for New Product Development: Delivering Educational and Industrial Value. *International Journal of Engineering Education, 26*(1), 178-194.

[31] Vidovics, B., Vukasinovic, N., Pavkovic, N. & Kovacevic, A. (2016). Introduction of Methodology for Distributed Collaborative Industry-Academia Project Based Learning. *Proceedings of the 14th International Design Conference (DESIGN 2016)* / Dubrovnik, 2016, 2125-2136.

[32] Kunz, W. & Rittel, H. W. J. (1970). Issues as elements of information systems. *Technical Report Working Paper No. 131,* Institute of Urban and Regional Development, University of California, Berkeley, 1970.

[33] Aurisicchio, M., Bracewell, R. & Armstrong, G. (2013). The function analysis diagram: Intended benefits and coexistence with other functional models. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 27*(3), 249-257. https://doi.org/10.1017/S0890060413000255

[34] Eisenbart, B., Gericke, K. & Blessing, L. (2013). An analysis of functional modeling approaches across disciplines. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 27*(3), 281-289. https://doi.org/10.1017/S0890060413000280

[35] Bracewell, R. H., Gourtovaia, M., Wallace, K. M. & Clarkson, P. J. (2007). Extending Design Rationale to Capture an Integrated Design Information Space. *Proceedings of the 16th International Conference on Engineering Design (ICED'07)* / Paris, 2007.

[36] Veisz, D., Namouz, E. Z., Joshi S. & Summers, J. D. (2012). Computer-aided design versus sketching: An exploratory case study. *Artificial Intelligence for*

Engineering Design, Analysis and Manufacturing, 26(3), 317-335. https://doi.org/10.1017/S0890060412000170

[37] Eppinger, S. D., Joglekar, N. R., Olechowski, A. & Teo, T. (2014). Improving the systems engineering process with multilevel analysis of interactions. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 28*(4), 323-337. https://doi.org/10.1017/S089006041400050X

[38] Chen, C. (2005). Top 10 unsolved information visualization problems. *IEEE Computer Graphics and Applications, 25*(4), 12-16. https://doi.org/10.1109/MCG.2005.91

[39] Herman, I., Melancon, G. & Marshall, M. S. (2000). Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on Visualization and Computer Graphics, 6*(1), 24-43. https://doi.org/10.1109/2945.841119

[40] Pavkovic, N., Martinec, T. & Storga M. (2014). Traceability - A Factor of Integration and a Method to Deal with Complexity. *Proceedings of the 10ᵗʰ International Workshop on Integrated Design Engineering* / Magdeburg, 2014, 93-105.

[41] Pavkovic, N., Storga, M., Bojcetic, N. & Marjanovic, D. (2013). Facilitating design communication through engineering information traceability. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 27*(2), 105-119. https://doi.org/10.1017/S0890060413000012

[42] Storga, M., Marjanovic, D. & Savsek, T. (2011). Reference model for traceability records implementation in engineering design environment. *Design Information and Knowledge Management, Proceedings of the 18ᵗʰ International Conference on Engineering Design (ICED'11)* / Copenhagen, 2011, 173-182.

Contact information:

Neven PAVKOVIĆ, Prof. Dr.
University of Zagreb,
Faculty of Mechanical Engineering and Naval Architecture
Ivana Lučića 5, 10000 Zagreb, Croatia
neven.pavkovic@fsb.hr

Tomislav MARTINEC, mag. ing. mech.
University of Zagreb,
Faculty of Mechanical Engineering and Naval Architecture
Ivana Lučića 5, 10000 Zagreb, Croatia
tomislav.martinec@fsb.hr

Dorian MARJANOVIĆ, Prof. Dr.
University of Zagreb,
Faculty of Mechanical Engineering and Naval Architecture
Ivana Lučića 5, 10000 Zagreb, Croatia
dorian.marjanovic@fsb.hr