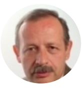
Case studies to explore indexing issues in product design traceability framework

Article · January 2011

4 authors, including:

Neven Pavković
University of Zagreb

Nenad Bojcetic
University of Zagreb



Dorian Marjanović
University of Zagreb

CASE STUDIES TO EXPLORE INDEXING ISSUES IN PRODUCT DESIGN TRACEABILITY FRAMEWORK

Neven Pavković[1], Nenad Bojčetić[1], Leonard Franić[2] and Dorian Marjanović[1]
(1) University of Zagreb, Croatia, (2) Končar MES, Zagreb, Croatia

ABSTRACT

Little is currently understood about the requirements for engineering information traceability in product development environment, and there are few methods by which effective traceability can be ensured. First part of paper presents two case studies: an analysis of current traceability practice in automotive industry supplier, and an experiment in implementation of taxonomy based software tool for knowledge indexing in medium sized company. Based on findings from case studies, the further research seeks the answers how the ontology based approach to defining the context and associated set of indices could lead towards generation of navigable semantic network that will be able to fulfil complex traceability requirements in customizable environment. Proposed approach suggests the definition of the context for tracing by "extracting" the subsets of ontology. Elements of ontology subset are associated with information objects (design documentation) belonging to design episode which is to be traced. Tracing procedure is focused (but not exclusively) on events that are the part of the process of information object management in PLM system.

Keywords: traceability, engineering information indexing, taxonomy-based indexing, ontology-based indexing

1 INTRODUCTION

Traceability of information provides the basis for assessing the credibility of engineering information, its better understanding and making judgments about the appropriateness of its use for a particular task [1]. Traceability has been considered as a quality attribute and many standards governing systems development require the creation of traceability procedures. In order to fully understand an item of information it is necessary to know the context in which it has been developed and recorded. Traceability is the ability to verify the past, location, or application of an item by means of documented recorded identification. The existing practice of recording the outcome of the engineering design process is almost exclusively based upon highly formalised model of the product, in the form of computer-aided engineering models, bills of materials, engineering change orders, etc. However, the detailed process, activities and rationale by which the design has been created and the engineering design information (EDI) developed (to the extent that they are recorded at all) are recorded largely in an informal manner [2]. A consequence is that is difficult to retrace or audit the engineering reasoning that has taken place during the process of EDI development without extensive work to assimilate and digest design documentation, and that identification of relevant parts of the information records within the documentation requires significant skill and often an intimate knowledge.

There are many similarities and overlapping issues between traceability issues and design rationale capturing. Design rationale capturing tools are beginning to be accepted in industry, e.g. the Design Rationale editor (DRed) developed by Engineering Design Centre (EDC) of Cambridge University [3]. The knowledge reuse is often ad-hoc and the designers often 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 [4]. The consequence is that designers are often sceptical about the purpose and usefulness of knowledge capture and indexing too. In the design environments that have little or no experience with knowledge management systems, designers often find those processes too obtrusive and time consuming, with small benefits for them. All the problems of knowledge capturing are equally present in engineering information tracing.

Little is currently understood about the requirements for engineering information traceability in product design and development environment, and there are few methods by which effective traceability can be ensured [5].

To explore aforementioned issues, the following research questions have been formulated:

1. When and how the requirements for tracing occur?
2. What are the situations that trigger the requirements for tracing?
3. What are the questions and requirements in most common "tracing" situations?
4. Which are most common "starting points" for tracing in current engineering practice – how they could be structured?
5. What is most often being looked for, and what is expected to be found?
6. In which form the answers are needed?
7. How to structure information fragments in information objects regarding to various contexts and phases of product development process?
8. How to represent and record informal information in traceability process?

The work reported here builds on the **TR**aceability of **EN**gineering **IN**formation - TRENIN (www.trenin.org) project by discussing the strategy for indexing engineering information and knowledge to provide efficient mechanisms for tracing and retrieval.

The objectives of the research presented here are:

- to develop simple as possible, but still efficient enough indexing system that will encourage designers to index and trace design information and knowledge,
- to identify and propose filter, query and browse mechanisms for information search and tracing,
- to offer the users (designers) an environment which is easily customizable for their particular contexts and domains.

The remainder of this paper is structured in the next three chapters. The results of two case studies have been presented in second chapter. The first case study has been done to analyze traceability practice and requirements in an automotive industry supplier for major European car manufacturers. The second case study is an experiment in implementation of taxonomy based software tool for information and knowledge indexing in medium sized company which produce electric motors, fans and electrical drives. The third chapter describes the research on how the ontology based approach for defining the context and associated set of indices could lead towards navigable semantic network that will be able to fulfil complex traceability requirements in customizable environment. Presented research findings together with related research efforts and future issues are discussed in the fourth chapter.

The common goal of presented case studies and research was to explore various methods and strategies of indexing and tracing engineering information in order to find the balance between complexity and suitability for everyday engineering practice.

2. CASE STUDEIS

2.1 Analysis of traceability in TRENIN industrial partner

The focus of this case study was to analyze current traceability practice in industrial partner company which is an automotive industry supplier. The main company product is car seat. Within automotive industry, traceability in production process has been brought to the very high level. Current efforts in company are directed to further develop and implement traceability in sales process (negotiation with customers) and design process. The analysis was based on interviews conducted with EO of main company departments, members of management board and with employees with significant responsibilities for the traceability process. In these interviews an emphasis was given on identifying current problems and especially on identifying good current traceability practice. Key documents, sets of rules and prescribed procedures regarding traceability have been identified and analysed. Company internal standards very precisely prescribe the initialization, flow, activities and responsibilities for all documents and some of their fragments involved in the particular business process. Those scenarios could be used as starting points for integration and further improvement of current traceability practice.

Traceability should help in maintaining a semantic network in which nodes represent information objects among which traceability is established through links (traces) of different types and strengths. The simplest traceability tools that have been found in engineering practice during the interviews with industrial partners involved in TRENIN project are suited only to support simple traceability procedures for personal use and provide limited support for full information objects (IOs) dependency analysis.

Based on analysis of information gathered through interviews, we have identified major traceability issues and requirements relevant for project management:

- what are (were) the implications on project management process for each transaction on one or more documents
- which documents are associated to one particular context or viewpoint, and where are they
- completeness and accuracy of document content in particular project milestone
- are all documents and information correctly and completely transferred from one main business process to another
- what were the major business changes in project portfolio, when and why they happen, how did they influenced currently active projects

These extracted requirements gave us some partial answers to research questions mentioned in introduction.

The current company efforts to achieve the first step of traceability on the document and information flow level could be summarized as following attributes that must be known for each created document:

- source (origin) of the document,
- all the destinations (users) of the document (all the persons who should receive the document),
- what the "receivers" should do with the document,
- where the document is being archived.

As common practice, business processes in most companies have a detailed workflow or prescribed scenario where each step includes detailed description of inputs – what, who, activities, and outputs – what, to whom.

In this way, a subset of "traces" of activities and information objects already exists in prescribed form enabling all process participants (who should perform tracing activities) to be aware of the traceability purpose, particularly to know in advance what, when, how and why should they record to achieve traceability. Business processes are represented with activity diagrams that comprise inputs and outputs, and responsible person. Main traceability issues that are prescribed (in internal process organization standards) and recorded in particular process are: who provided particular inputs, what these inputs were, who did the activity, what were the outputs of the activity and to whom the outputs were send. In such approach, all the destinations and activities being performed on documents and document fragments involved in the particular business process are precisely prescribed. This process works in practice, but currently it is a mix of "paper work" and computer supported document management whose weaknesses are discussed in the rest of this chapter.

Traceability points and procedures that exist in current company practice are mostly isolated islands managed by one particular person in each department. In most cases these islands are named "monitoring sheets" – created as spreadsheets or tables in text processor. Chunks of data from these isolated islands are being transferred to a structure of big and complex spreadsheets again managed by one responsible person – the EO of the "document management department". This department is the central place for documentation management and approval – and a place where actually the majority of traceability operations are performed. The most complex traceability document structure being created is "Project portfolio monitoring sheet" which traces major business changes and their implications to currently active design projects. Lacking a common database, many redundancies and unnecessary data transfers are being generated. Such a huge structure of spreadsheets provide only very limited indexing and search mechanisms. The consequences are: only one person could work with and access the file at the same time, it is difficult to manage huge amounts of data, there are limited possibilities for avoiding and controlling errors, only very simple structuring and capturing of newly created knowledge is possible. The first step to improve this situation is to build common integrated "monitoring" database. As already indicated, existing prescribed scenarios and procedures could be used as starting points for integration of "monitoring sheets" and further improvement of current traceability practice.

2.2 Taxonomy based knowledge and information objects indexing

This case study explores the proposition that a simple and intuitive interface combined with "easy to use" methods of indexing and searching might lead to a system that will be "efficient enough" and especially "comfortable enough" for practical usage. To demonstrate this idea a prototype software tool for knowledge indexing has been developed [6], together with the proposal of relational database structure for indexed knowledge records. "Knowledge record" is here considered as data structure that include a set of links to information objects and/or their fragments, coupled with textual explanation(s) of design rationale, being connected in particular context. Knowledge record is recorded in database together with associated set of tags (indices) (figure 1). Set of tags should provide precise knowledge retrieval of relevant knowledge records in the process of design knowledge reuse. In the proposed approach each tag corresponds to one element (tree node) in one particular taxonomy. It is assumed that a set of several taxonomies for a set of different contexts (or points of views) will be used.

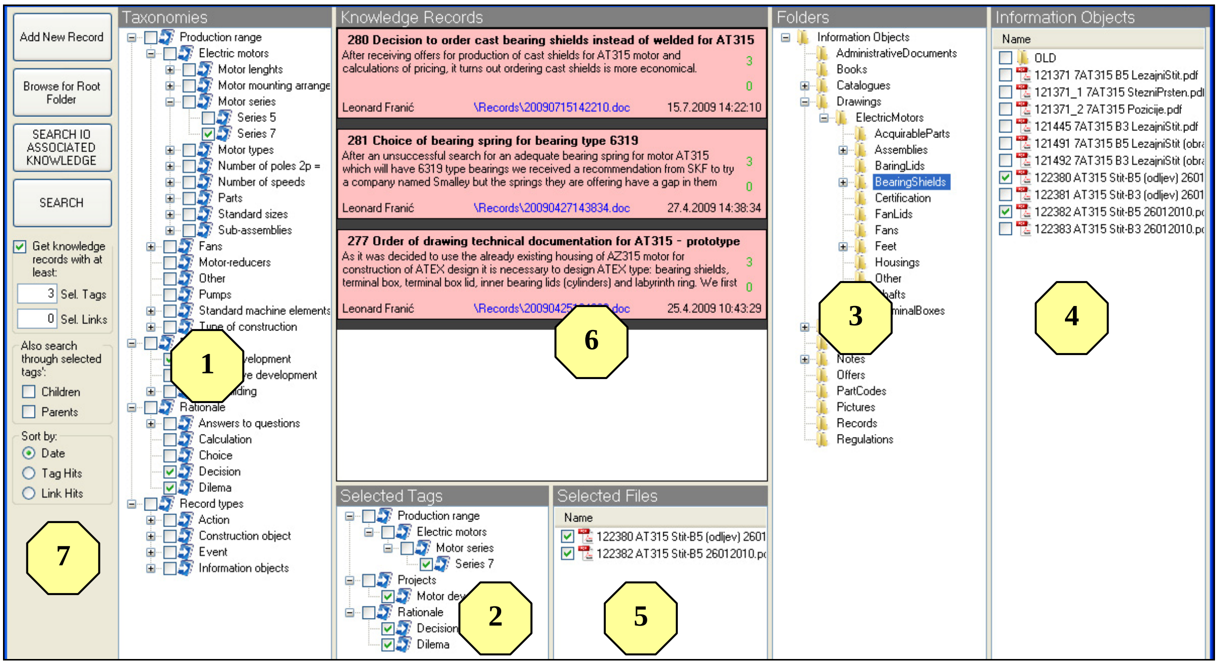


Figure 1. Main interface of proposed tool [6]

Figure 1 represents main form (window) where numbers indicate elements of interface:

1. "Tree View" control for manipulation with taxonomies – each element has a check box;
2. "Buffer" of currently selected (checked) taxonomy nodes that will be a set of tags for particular knowledge record indexing or will serve as a search criteria in knowledge reuse process;
3. Folder structure with design documentation (information objects) – a view on actual file structure on selected root folder;
4. List of files (information objects) in currently selected folder – with check boxes for selecting;
5. "Buffer" of currently selected files (inf. objects) – associated with particular knowledge record;
6. List of "knowledge records" that matched search criteria;
7. Main toolbar with buttons and options.

2.2.1 Overall usage scenario

Proposed usage scenario includes following steps: (1) *initial creation of taxonomies* that will be used for indexing, (2) *knowledge capture* process which comprise: writing of knowledge chunks in knowledge records, selecting sets of taxonomy nodes to index the records and selecting sets of relevant information objects associated with knowledge record, (3) *knowledge retrieval process* – search for previously indexed records.

When a need for knowledge reuse occurs, the search for relevant knowledge records begins with exploring (browsing) the same taxonomy structure that was used for knowledge records indexing (figures 1and 2). The designer has to select (mark) the check boxes near the taxonomy nodes that are relevant for his/her knowledge search. As a consequence the database has to be searched for all the knowledge records that contain one or more tags (indices) associated with selected taxonomy nodes.

Described tool has been implemented in design office of medium sized company which employs 5-6 designers. For such relatively small environment this level of knowledge and information objects indexing proved to be sufficient. An example of two knowledge records with associated information objects and selected tags (indices) is shown on figure 2.

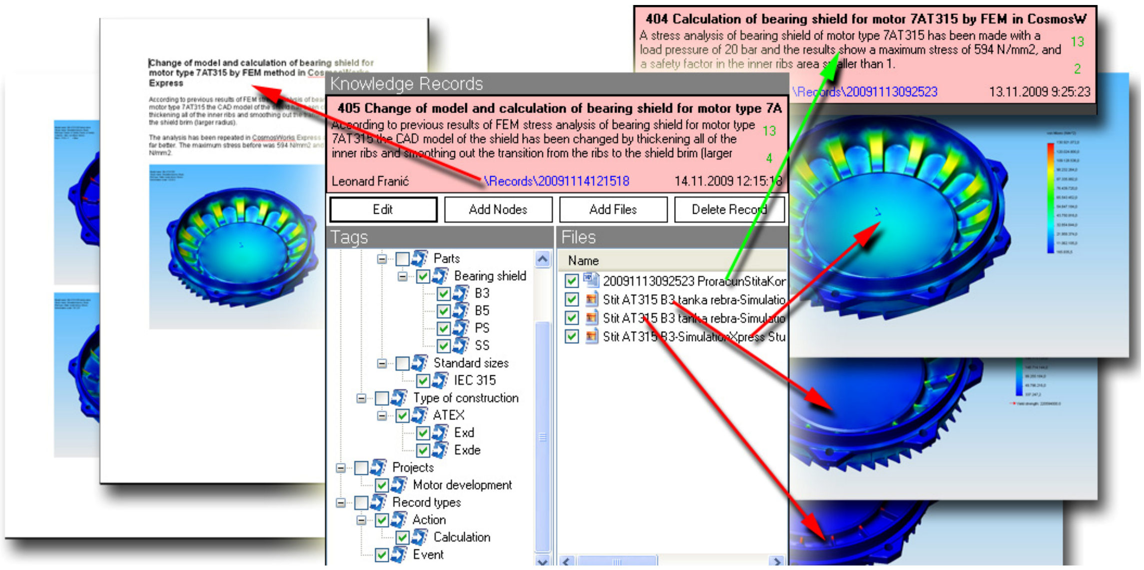


Figure 2. Example of knowledge records with associated documents [6]

3. ONTOLOGY BASED INDEXING AND DEFINING CONTEXT FOR TRACING

For more complex indexing and tracing issues, taxonomies are not sufficient. Therefore, an approach that relies on "contexts for tracing" viewed as subsets of ontology has been researched. Similarly as taxonomy serve for selecting a subset of hierarchically structured indices (tags) we are developing and testing a concept where a subset of ontology will define a broader and richer context for indexing and tracing the complex set of information objects, designers, their actions and situations in design process history. In this approach, ontology denotes a conceptual data schema that represents the relevant domain entities and their relationships by means of classes and relations [7]. There are several tools available for ontology management and a standard machine-readable language for representation – together offering an appropriate framework for exploring indexing and tracing strategies. These tools also have adequate ontology visualization capabilities, logic-based reasoning and ontology integrity checking. For presented research, the general product development ontology is defined in **TR**aceability of **EN**gineering **IN**formation - TRENIN project (www.trenin.org). This general ontology could be edited, adapted and extended according to needs of particular company.

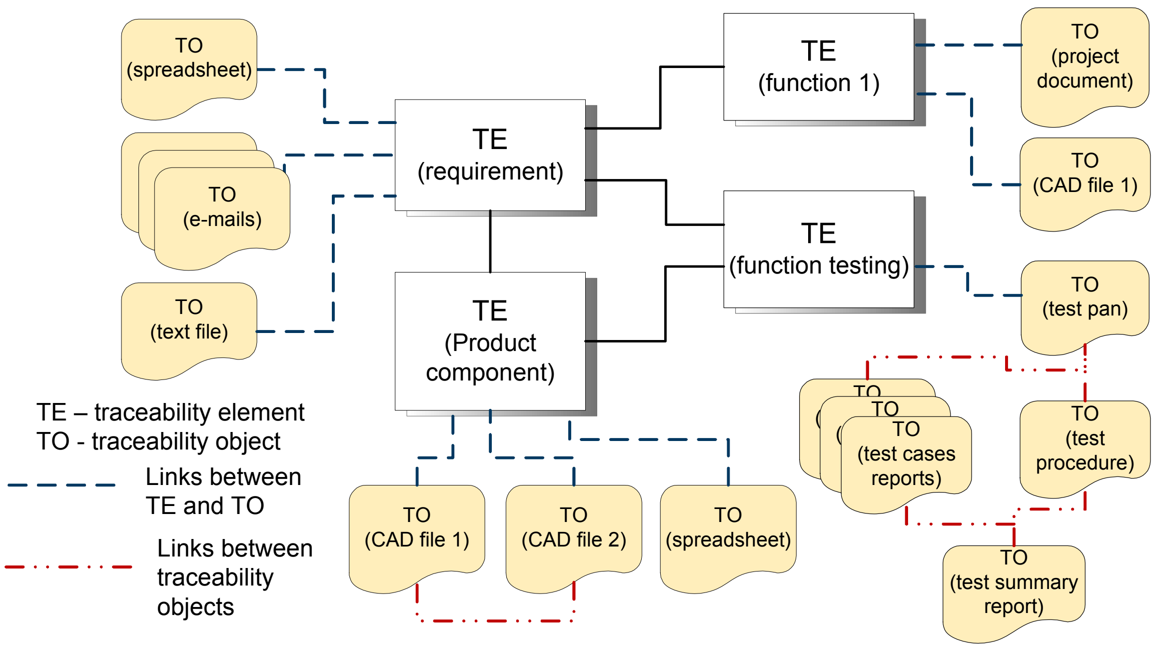


Figure 3. An example of traceability record scheme

In proposed ontology two main areas have been distinguished: *traceability elements* (TE) and *traceability objects* (TO) (figures 3 and 4). Traceability elements are entities from design process domain while traceability objects are entities from design organisation and management domain, with special focus on design documentation. Relations in proposed ontology are named *traceability links*. The main concept proposed for defining context in this case study is named *traceability record* (TR). This is a container for a subset of ontology which purpose is to trace the development process of selected traceability objects linked in specified context with selected traceability elements. Figure 3 is an example of traceability record scheme – each of four interrelated traceability elements has several associated traceability objects – all together these elements compose the particular context for tracing. Relations between traceability objects indicate either information flow or structural and/or chronological dependencies. While relations between traceability elements explain context, relations between information objects (TO) should be primarily viewed as traces of information development.

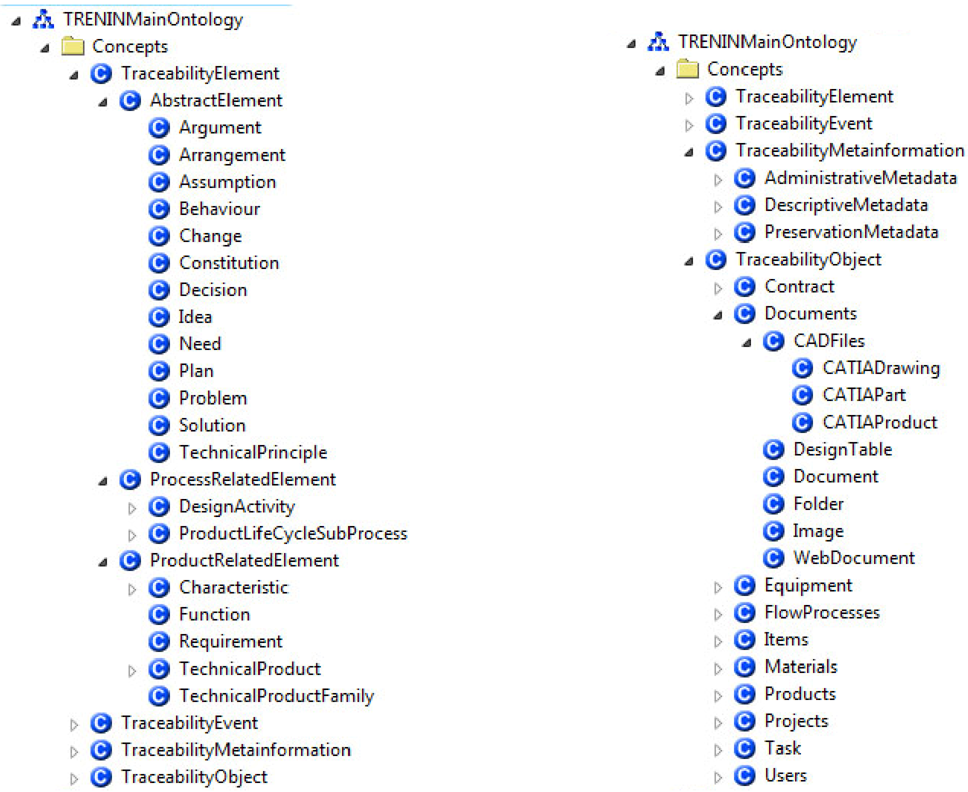


Figure 4. A proposal of traceability elements and traceability objects taxonomies

Software tool which will provide interface for definition and initialization of traceability records currently is in last phase of prototype development. Figure 5 shows first concept of interface for creating traceability record. This tool uses the standard ontology record format OWL. Before creating a particular TR, necessary instances of entities that will be "placed" on TR has to be created in ontology browser. For each ontology entity, lists of these instances and their relationships are offered for selection and "placement" on TR being created.

Three phases could be recognised in the process of creating a TR:

1. selecting of traceability elements and their relations relevant for particular context: designer firstly selects the desired concept in ontology browser, then a list of available concept instances is provided, selected instance becomes the "part" of TR,
2. pairs of TE should be selected to establish instances of relations between them,
3. selecting of traceability objects and associating them to traceability elements

Instances of traceability objects are mainly design documents in form of files (e.g. CAD parts, assemblies and drawings, textual documents, spreadsheets etc.).

On figure 5, a situation in third phase of TR creating process is captured: four concept instances (TE) have been selected from ontology structure visualization window and "placed" on TR, their relations (links) have also been established. The process of associating information objects whose development will be traced is undergoing by filtering and selecting using several interface elements.

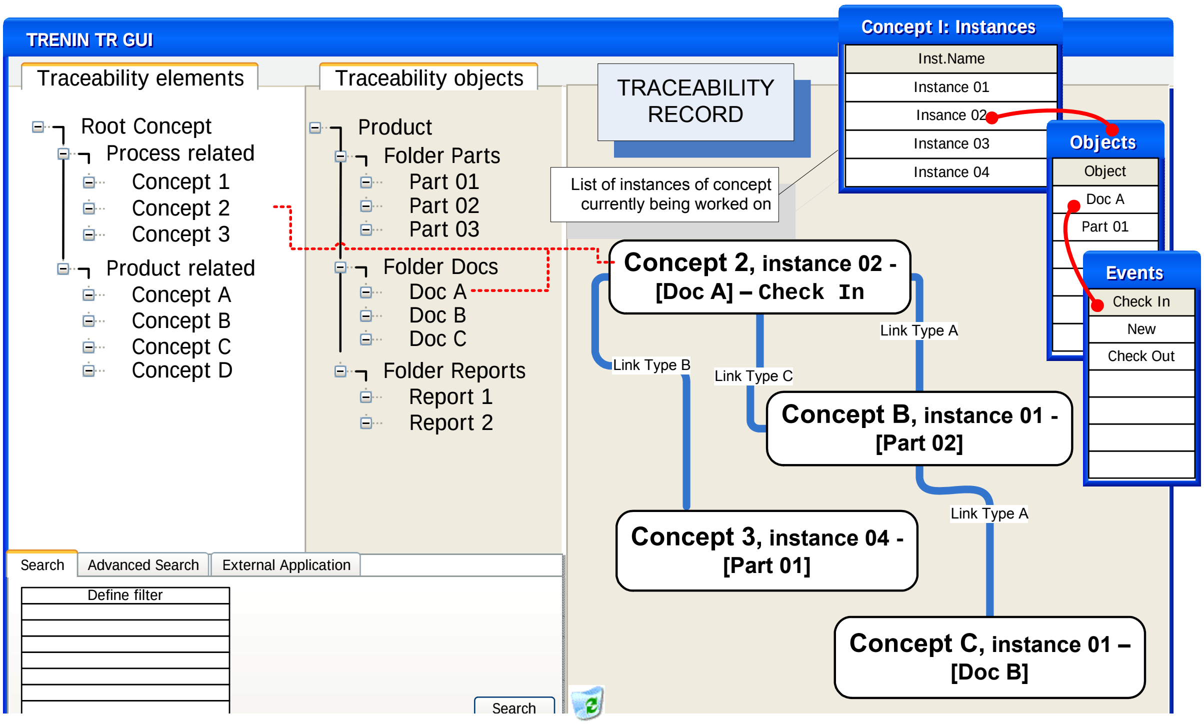


Figure 5. Interface for creating traceability record

At this phase, the following question has to be answered: how the traces of the design documentation development will be captured and recorded through the timeline of particular product development project? Proposed approach is to extend an existing PLM environment with intelligent agent technology in order to enable autonomous traceability actions necessary for traceability execution. The core of the idea is the *traceability engine* (TE) that, based on the specific events related to the PLM environment and PLM information objects, executes "intelligent" agents responsible for traceability tasks related to specific event (e.g. check-in, check out, etc.). The final result is the ability to record each change on set of design documents being managed in PLM environment. Desired (or relevant) changes are finally recorded as "event history" associated to each traceability object of particular traceability record. Database tables will contain records of the every change of the PLM information object (including content, attributes, links, etc) and what is especially important, context of the change provided automatically or by help of human user involved in traceability process. The main idea behind this approach is that the current state of the PLM information object is a superposition of initial state and changes over the time. Traceability object also includes the list of all versions throughout the information object life cycle.

The focus in presented research is to develop the traceability framework tightly connected to PLM environment, but traceability engine will also be able to communicate with other tools and applications. That means that instances of traceability objects are not necessarily only information objects from PLM system.

To further elaborate proposed approach it is necessary to distinguish three phases in traceability record (TR) life cycle:

1. initial definition – selection of the subset of ontology and association of information objects
2. the period of "active tracing" – while the TR is "active" - changes and states of information objects are being recorded
3. after the period of "active tracing" is finished, TR is archived and ready for searching and traceability processes

Instead being the pure static list of the information objects and hyperlinks between them, archived (finished) traceability record should be more "intelligent" and dynamic container of the traceability elements, information and links semantically enriched in order to provide the context of the informational content development. The research and development question that should be answered in further research and development is how to connect traceability records to form a semantic network. The final goal of such an approach should be the ability to navigate through such a semantic network and to perform a semantic search to achieve full traceability of product development process.

The initial implementation of proposed model based on researched ontology based indexing and context defining for tracing is planned in real environment of company which develops and produces electrical converters for railway vehicles.

4. DISCUSSION AND RELATED WORKS

The aim of this chapter is to discuss the proposed approach with related research efforts.

Gotel and Finkelstein [8] investigated and discussed the underlying nature of the requirements traceability problem. They found that providers of traceability and end-users have conflicting problems and needs. From the viewpoint of providers [8] traceability depends on:

- working practice – sufficient resources time and support, ongoing cooperation and coordination,
- awareness of information required to be traceable,
- ability to obtain and document required information,
- ability to organise and maintain required information for flexible traceability requirements of end-users (supporting change, restructuring, etc.).

From the viewpoint of end-users [8] traceability of information and access to and presentation of information depends on who wants it (user), why/when they want it and on project characteristics.

Our first case study resulted with very similar findings to those of Gotel and Finkelstein: main efforts in analysed company are directed towards improving abilities to identify, organise and maintain information required to be traceable (by prescribing necessary procedures) and to disperse awareness that this information should be traced.

4.1 Indexing strategies and issues

Ahmed et al. tried to identify concepts of the taxonomies required for indexing design knowledge [9,10]. This research identifies two main advantages in having a visible indexing structure: (1) to assist designers in focusing their query through browsing or navigating and indexing structure; and (2) to overcome difficulties in search engines not understanding the context of a query. Ahmed proposed the method consisting four taxonomies named as Engineering Design Integrated Taxonomy (EDIT) which have been integrated and implemented in software named CITED (Cambridge Integrated Taxonomy for Engineering Design).

Additionally to mentioned advantages of having a visible indexing structure [9,10], on the basis of presented case studies we argue that following strategy is essential:

When a need for tracing (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. Such 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. This requirement is completely fulfilled in second case study, but for the ontology-based concept it is still to be cleared how to solve this issue due to system complexity.

Second case study gave us an insight in problems that may occur in process of initial definition of entities and hierarchical structure of taxonomy and/or ontology for practical usage in particular 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. Due to ambiguity of abstract categories developed through examination of engineering knowledge using top-down approach, we argue for "bottom-up" approach where hierarchy is being built from leaf and/or lower level taxonomy nodes which in fact represent the end solutions - actual or concrete terms that could be easily defined and understood by designers. Further research question that arises here: is it possible to manipulate simultaneously several indexing taxonomies with different contexts on the "root" level(s), but having common entities on lower levels? Interesting discussions on similar issues could be found in [11] where two different approaches to design ontology development are compared.

Brandt et al. [7] argue that ontologies have two major advantages over conventional data schemas: firstly, they are highly flexible, enabling modifications and extensions of the data structures even during project execution. Secondly, they are represented in a machinereadable, logic-based language, which allows to formally define the semantics of the ontological concepts. This enables the computer to interpret and reason with the information stored in the ontology. In consequence, advanced support for information management and retrieval can be provided: for example, it is possible to perform a semantic search on the ontology.

Quertani et al. [12] propose an approach for traceability of product knowledge during product design and manufacturing. Their strategy is based on the Zachman framework [13] and they clearly distinguish design and manufacturing perspectives for traceability. Findings from our first case study confirms that these two perspectives are really very different with very little common implementation issues and problems. Therefore we must emphasize that implementation of traceability in design process is very labour intensive requiring organizational processes and software support of much higher complexity level than in manufacturing process.

4.2 Utilization

Mohan and Ramesh [14] present an approach to knowledge integration in group decision and negotiation activities using traceability. They define knowledge integration as the synthesis of specialised knowledge that is distributed across different artefacts and phases of product development into situation specific systemic knowledge. Therefore the traceability tool acts as a platform that links various knowledge chunks that are distributed across different stakeholders and their systems.

Based on proposed approach to defining context for tracing and indexing with traceability records that are subsets of ontology, the mentioned integration could be accomplished in two ways:

- using the existing relations in ontology to navigate (perform semantic search) between related elements of several traceability records (which define different contexts),
- establishing new (either hierarchical or binary) relationships between traceability records that will be recorded separately and used for solving complex traceability requests.

Both ways should lead towards building a complex navigable semantic network structure.

Brandt et al. [7] have developed the module that comprises concepts for structuring traces recorded during design process execution. In description of utilization of their ontology based model [7], they say that semantic relations cannot only be used to connect similar contents, but also to interrelate "content descriptions" with complementary contents, thus pointing to additional product knowledge that would otherwise remain undetected.

Findings from the first case study draw the conclusion that it is very difficult to predict the context(s) or to classify the complex traceability requests like e.g.:

- exploring the source (origin) of troubles occurred in manufacturing or malfunctions in exploitation,
- finding relevant information and documentation for simulating already executed design episode in another similar situation,
- using traces of previous design episodes for educational process for inexperienced designers, etc.

Therefore, in this phase of research the detailed overview of proposed model utilization scenario(s) is still not complete and clear. Further real implementations of finished model in several companies would give us new insights and offer new ideas for necessary improvements towards abilities to answer more complex traceability requests. In proposed model, traceability record structural elements contain numerous links (references) to information objects and the history of their states. The first (simplest) step in utilization is browsing the network of linked documentation. Described model and procedures are primarily focused to structure newly generated documents, information and knowledge, but it could also be used for existing resources in that case without the "event history" and versioning history.

5. CONCLUDING REMARKS

The contribution of this research is the development of procedures and indexing methods in ontology based framework for traceability, offering users semantically and technologically rich mechanisms to structure and compose paths of traces that will lead them towards desired answers to complex traceability questions. The novelty in proposed model is the process of "extracting" the subset of ontology resulting in a set of related concept instances. These concept instances are associated with information objects belonging to design episode which is to be traced. In such approach the selected ontology subset defines the semantically rich context for indexing and tracing particular design episode. The tracing process is based on events that occur in the information object management process in PLM system. The system trace the development history of set of information objects associated to contexts defined with subsets of ontology. More detailed explanation of PLM events tracing process is out of scope of this paper and will be given in future publications.

REFERENCES

[1] Štorga M., Darlington M., Culley S.J. and Marjanović D. Traceability of the development of 'information objects' in the engineering design process. *Proceedings of the 2nd International Conference on Research into Design ICoRD'09*, Bangalore, India, 2009.

[2] Giess M. D., Goh, Y. M., Ding L. and McMahon C. A. Improved product, process and rationale representation and information organisation to support design learning, *International Conference on engineering design ICED '07*, Paris, August 2007

[3] Bracewell R., Gourtovaia M., Wallace K. and Clarkson J. Extending design rationale to capture an integrated design information space. *Int. Conf. on Eng. Design ICED '07*, Paris, August 2007

[4] Kim S., Bracewell R., and Wallace K. Improving design reuse using context. *International Conference on Engineering Design ICED '07, Paris,* August 2007

[5] Štorga M., Darlington M., Culley S.J. and Marjanović D. Toward a process and method for tracing the development of information objects used in engineering design. *Proceedings of the ICED 09 International Conference on Engineering Design,* Stanford, USA, 2009.

[6] Franić L. A system for recording and indexing design knowledge. *Master thesis, Faculty of mechanical engineering & naval architecture, University of Zagreb*, 2010 (in Croatian)

[7] Brandt S.C., Morbach J., Miatidis M., Theiÿen M., Jarke M., Marquardt W. An ontology-based approach to knowledge management in design processes. *Computers & Chemical Engineering* 2008, 32 (1-2), pp 320-342.

[8] Gotel O.C.Z. and Finkelstein C.W. An Analysis of the Requirements Traceability Problem. *Proceedings of the First International Conference on Requirements Engineering*, Colorado Springs, CO, USA, April 1993, pp 94 – 101

[9] Ahmed S. and Wallace K. Indexing design knowledge based upon descriptions of design processes, *International conference on engineering design ICED '03*, Stockholm, August 2003

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

[11] Ahmed S. and Štorga M. Merged Ontology for Engineering Design: Contrasting an empirical and a theoretical approach to develop engineering ontology, Cambridge Univ. Press, *AIEDAM - Artificial Intelligence for Eng. Design, Analysis and Manuf.*, Vol. 23 No.4, 2009, pp. 391-407

[12] Ouertani M.Z., Baina S., Gzara L., Morel G. Traceability and management of dispersed product knowledge during design and manufacturing. *Computer-Aided Design, Elsevier, 2010.* In press

[13] John Zachman's Concise definition of the the Zachman FrameworkTM. Zachman International. 2008. http://www.zachmaninternational.com/concise20definition.pdf.

[14] Mohan K. and Ramesh B. Traceability-based knowledge integration in group decision and negotiation activities. *Decision Support System*, Elsevier, Volume 43 (2007), pp 968-989

Contact: Neven Pavković
University of Zagreb
Faculty of Mechanical Engineering & Naval Architecture
Ivana Lucica 5, 10000 Zagreb, CROATIA
Tel: +385 1 6168 545, Fax: +385 1 6168 284
Email: neven.pavkovic@fsb.hr, URL: http://www.cadlab.fsb.hr

Neven is Associate Professor of Engineering Design in the Department of Design and Product Development at the Faculty of Mechanical Engineering. His primary research areas are design organisation and management and knowledge management. Currently he is focused to development of efficient and less obtrusive methods of knowledge capturing and indexing to achieve traceability.