INTERNATIONAL DESIGN CONFERENCE - DESIGN 2014
Dubrovnik - Croatia, May 19 - 22, 2014.

TOWARDS A DECISION SUPPORT SYSTEM FOR ENGINEERING CHANGE MANAGEMENT

M. Niknam, E. Huang and J. Ovtcharova

Keywords: engineering change management (ECM), decision support system (DSS), product lifecycle management (PLM)

1. Introduction

With the growing complexity in products and product development processes, the rapid market alterations and the raising importance of time to market, the importance of Engineering Change Management (ECM) is becoming prominent to all organizations [Pikosz and Malmqvist 1998], [Jarrat et al. 2010]. The faulty product releases or late changes have caused vast product recalls or redesigns and consequently exerted huge financial and non-financial expenses on well-known brands such as Toyota, Apple, Volkswagen, etc. Therefore, the importance of having robust Configuration Management and Engineering Change Management processes in place are well understood by most entreprises. The reason behind the recent research trend on providing guidelines and solutions on managing the engineering changes effectively and efficiently is its high potential in affecting the product cost, leadtime and quality as well as the high potential for its maturity enhancement [Niknam and Ovtcharova 2013]. The more complex the product, production facility, organization and value-chain, the harder the in-time recognition of changes, understanding of their complete effect and distribution of their information among stakeholders [Niknam et al. 2013].

One of the primary challenges in this area is the management of all information associated with the change and taking a wise decision respectively. As the human experience and knowledge, could only consider limited amount of variables at a time, Artificial Intelligence (AI) in the frame of computer-based information processing systems could be of assistance. More specifically, Decision Support Systems (DSS) are known as useful tools to support the decision making process in highly information-dependent situations. Therefore in this paper the feasibility of developing a DSS for ECM is analysed and discussed.

The rest of this paper is organized in the following order. First the generic ECM process and the important factors involved in it are discussed. Then, an introduction of DSS is given and main components are introduced. After, the authors propose their proposed structure of the DSS for supporting ECM and explain its information flow. At the end, a conclusion and self-reflection is made and future research roadmap is identified.

2. Engineering Change Management

Terwiesch and Loch [1999] defined Engineering Changes (ECs) as "changes to parts, drawings or software artifacts that have already been released". Based on their definition, Jarratt et al. [2004a] added that "The changes can be of any size or type; the change can involve any number of people and take any length of time."

Eckert et al. [2004] categorize ECs according to their Reason; "Emergent Changes" which arise from the properties of the product itself and most often-seen ones include Error correction, Safety issues, Change of functions, Quality problems etc. as well as "Initiated Changes" which are improvements, enhancements or adaptions of a product and arise from outside the product. The triggers for such changes are variant among different industries, however in general they could be devided in the following critera.

Table 1. Triggers for Initiated Changes

Cause	Example
Market Environment	Customer requirements (e.g. cost, performance), Economic Situation (e.g. financial crisis)
Technology	State-of-the-art Equipment or Technologies available for performance enhancement
Legal background	Emission standard, Safety requirements
Company's strategy	In-house manufacturing or outsourcing
Political background	Political situation of the country e.g. Sanctions
Supply Chains	Supplier relationship, Contractual matters

Several other authors [DiPrima 1982], [Jarratt et al. 2010] classified ECs by their Priority; "Immediate Changes" are ECs that have to be applied immediately e.g. safety issues or critical customer requests. "Mandatory Changes" have to be applied as soon as possible, but with some degree of flexibility. "Convenience Changes" are not as vital as the other two groups and their application could be postponed to whenever it is practicable, e.g. small improvement.

For long lead-time products, Reidelbach [1991] suggests to classify ECs based on timing within the development process: "Early ECs", "Mid-production ECs" and "Late ECs". This categorization is basically due to the proven different impact level of each type with respect to their emergence time in product lifecycle.

2.1 Generic Process

A few authors have proposed generic process overview for ECM. Although the terminology used and the exact number of steps introduced might be different, more or less all processes follow the same procedure and phases. Jarratt et al. [2004a] suggested a comprehensive six-step process which is shown in figure 1.

The process starts with the realization of the need for change. The engineer initiates the change and includes solution(s) in the change request and proposes it to the decision making authorities. After an in-depth impact evaluation, the committee will decide if the EC is accepted. If yes, the EC will be implemented and if not the engineer gets a feedback on the reason. After the implementation, there is a review phase, during which the company can see if the result is desirable and document the lessons learned. Furthermore, there are some possible iteration loops when something need to be redone and 4 break points to have a short evaluation of the current progress.

2.2 Impact of ECs

Although some engineers and employees might consider ECs as problems due to their impacts and resource needs [Ring and Fricke 1998], [Acar et al. 1998], they are inevitable parts of all product development environments and important driving forces for product and process maturity enhancements.

Riviere, et al. [2002] list some possible aspects that an EC could exert influence on: *costs*, *lead-time*, *product performance*, *other physical components* and *supply chain*. Jarratt et al. [2004b] have a more general view on the subject and state that there are two categories where the impact acts upon; the product and the process (budgetary, organization and schedule considerations).

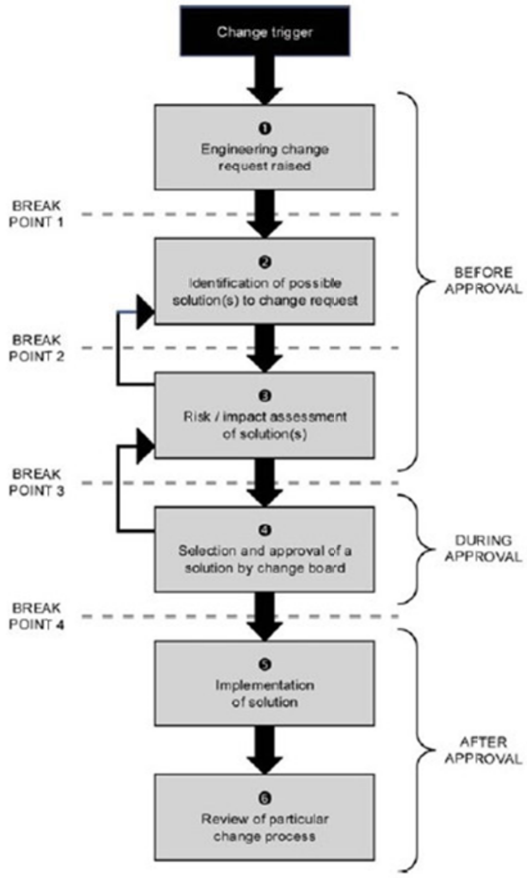

Figure 1. ECM Generic Process [Jarratt et al. 2004a]

2.2.1 Product Factors

The impact of a change on a product is governed by 3 factors: *the complexity*, *the architecture* and *the degree of innovation* [Jarratt et al. 2010].

The key type of *complexity* from an engineering change perspective is connectivity [Jarratt at el. 2010], which is the relationship or linkage between elements and their interaction. With a complex product, it is much harder to control all the relevant parameters and their impacts on each other [Fricke et al. 2000]. *Product architecture* influences the propagation of an EC. Product architecture is defined as "(1) the arrangement of functional elements; (2) the mapping from functional elements to physical components; (3) the specification of interfaces among the interacting physical components" [Ulrich 1995]. *Modular* and *integrated* are the main two types of product architecture. In practice, most products are allocated somewhere in between the two architecture types [Jarratt et al. 2010]. Obviously, ECs in a modular architecture should cause much less trouble than in an integrated architecture. But this is only when the interface to other modules are not affected.

Due to lack of information and knowledge, in products with a high *degree of innovation* e.g. high-tech engineering-to-order products, they are more prone to change occurance and propagation.

2.2.2 Process Factors

The extent of EC impact can also depend on when and where it happens during the product lifecycle.

The "when" stands for the timing which is one of the main factors in identifying how big the impact is to the processes. Even if the EC happens to the same element of the production, the later the EC occurs, the bigger its impact, since in the later stages of the lifecycle, the change propagates into other associated processes as well. Many authors [Clark and Fujimoto 1991], [Riviere et al. 2002] refer to a "Rule of 10" that the cost of implementing a change in a later phase increases on average by a factor of 10 compared to the previous phase. On the other hand the "where" stands for the function that the change is dependent on eg. supply chain, procurement, etc. Jarratt et al. [2010] noted that problems can be amplified by difficulties in the supply chain, as it compreses the schedule or compromises the product quality.

2.3 Current ECM support solutions

Some solutions have been introduced by a few researchers, which mainly focus on supporting the impact analysis and decision making phase [Jarrat et al. 2004], [Jarrat et al. 2010]. Several of these systems and their focus areas are presented below.

Table 2. Technologies for ECM Decision Support

Technology	Focus
PLM systems	Supporting the workflow and automation of the ECM processes
Computer Aided Design (CAD)	Analysis of the impact and mismatch of design changes based on geometrical values
Redesign IT [Ollinger and Stahovich 2001]	Can generate and evaluate proposals for redesign plans by using a product model with relevant physical quantities and causal relationships
Change FAvourable Representation (C-FAR)	It exams the attributes and interactions between the core elements of various entities in product models to predict the propagation effect
Change Prediction Method (CPM) [Clarkson et al. 2001]	Helping the understanding of how changes propagate through a product based on Design Structure Matrix (DSM)
ADVICE [Kocar and Akgunduz 2010]	Virtual environment for ECM with the prioritization agents and propagation prediction techniques
Hamraz et al. [2012]	Multidomain Engineering Change Propagation model utilizing DSM and Multidomain Matrices (MDM) for predicting the change propagation

2.4 Motivation

Most of the mentioned tools are either too general to bring added value to the decision making process or highly focused on specific topics such as propagation analysis which make their application highly dependent on specific data and still their support level is not comprehensive enough to cover all aspects of decision making. There is a trade-off between accuracy and resource usage e.g. time and computer memory. Most of the mentioned tools concentrate deeply on the accuracy which makes the calculations computationally heavy and therefore their applicability/effectivity reduces in highly complex environments. The authors noticed that, utilization of previous historical data is not considered in most tools which could be an interesting complementary option for both accuracy and speed. With the historical data, since there are existing data as references, the resource usage can be reduced. More than that, non-functional attributes such as cost, safety, reliability, etc, can also be recorded for more information. Additionally in this way the tacit knowledge can be documented in common language and be reused. In order to exploit such functionalities, Knowledge Management

(KM) tools are required and in this paper the Decision Support System as a potential tool is presented as an interesting candidate.

3. Decision Support System

A Decision Support System (DSS) is a computer-based information system that aids the decision-making process which is a cognitive process of selecting one course of action from alternatives [Gebus 2006]. This is done by analyzing the data about different aspects of the domain based on the underlying algorithms and proposing a suitable decision respectively [Sage 1991].

There are 5 types of DSS: Communication-Driven DSS, Data-Driven DSS, Model-Driven DSS, Document-Driven DSS and Knowledge-Driven DSS. With the capability of self-learning, identifying data associations and heuristic operations, Knowledge-Driven DSS (KDDSS) is realized by the authors as a suitable match to the characteristics of ECM problem.

A knowledge-driven DSS has the following components:

1. Knowledge-Base (KB): it is the core component of a KDDSS. Expertises and information are collected and stored in knowledge-base in the form of facts, rules, etc.
2. Inference Engine (IE): it is the software that actually performs reasoning function [Power 2000]. It contains a Data Mining or Self-learning Module in order to "learn" useful knowledge and a Reasoning Module for reasonable decision support depending on the strategy and situation. It draws its conclusion from the knowledge represented in the knowledge-base.
3. User Interface: it allows the interaction between decision makers and the system.

A generic KDDSS architecture could be seen below.

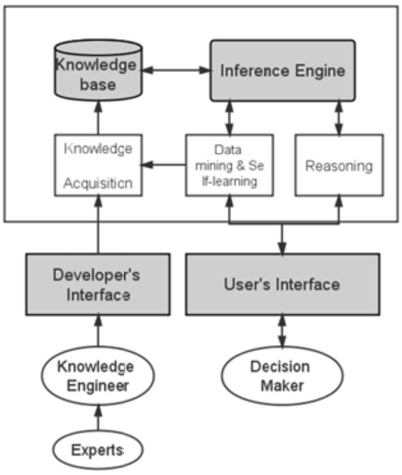

Figure 2. Proposed KDDSS Architecture

3.1 Knowledge-Base Structure

Knowledge-base is the core part of the KDDSS. Based on previous works in Knowledge-base design [Arian and Pheng 2006], [Kohn et al. 2013] the authors propose a Knowledge-base structure seen in Figure 3.

The proposed Knowledge-base consists of 3 layers. The first layer is the **Fact** layer. Facts are acquired during the implementation of the DSS. They are the raw material to be used in the Knowledge-base, without which no information can be gained by the system. Therefore, facts layer lays the foundation for other layers in the Knowledge-base. The second layer is the **Rule** layer which contains the rules on how to implement the facts. The third layer is the **Strategy** layer. Here the strategy is bidirectional. On the one hand, it gives suggestions to the user of the Knowledge-base which is not necessarily a human being and could also be another component of the system. On the other hand, it controls the second layer, or acts as "the rule of the rule". Such strategies are defined by the company's strategy during the Knowledge Acquisition and analysis. An example of how each layer could look like is provided in the next chapter in more detail.

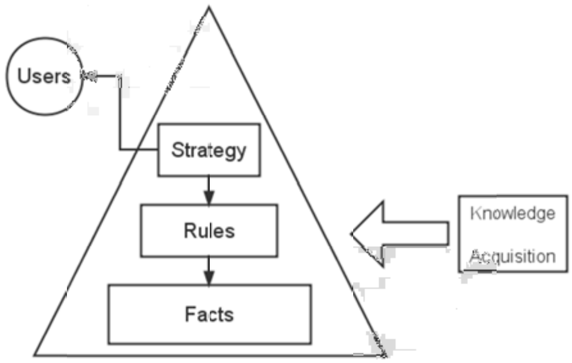

Figure 3. Proposed KB Structure

4. Decision Support System for Engineering Change Management

Based on previously introduced ECM generic processes, and specifically considering the fact that a DSS could ease the impact evaluation phase by utilizing the historical EC data, the authors propose the following ECM process which consists of more detailed steps and the associations with the DSS.

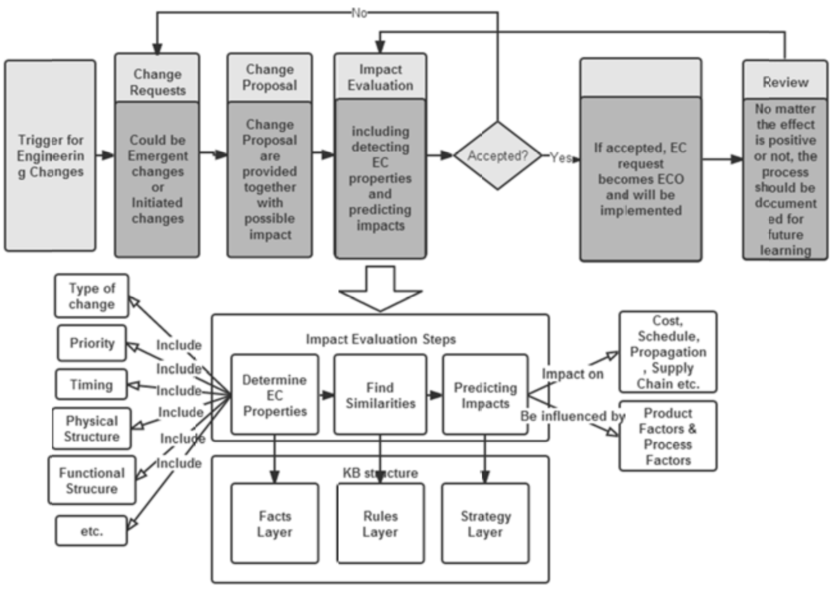

Figure 4. ECM Process Suitable for DSS

Compared to previous processes, the authors divide the Impact Evaluation phase further into 3 steps: **Identification of EC properties, Finding similarities –** compare the identified properties with the properties of past ECs and recognizing the most similar changes with respect to the properties; as well as **Predicting impacts –** utilizing the identified similarities and the factors that might influence the impact of changes. In Figure 4, the connection between the ECM process and KB structure is further illustrated. **EC properties** are the raw data of an EC and therefore go to the **Facts layer**. As **finding similarities** step uses the properties to find relationships, it is refered to as the **rules layer**. The **prediction of impacts** can be realized by the **strategy layer.** The strategy layer uses the similarities found by the rules layer, and by utilizing the expert knowledge rules and machine learning algorithms predict the impacts.

In order to further describe the DSS structure, a detailed information flow is carried out for the realization and a more graphic description is also given in figure 8. The steps are textually described below with the help of a simple example from the computer industry. It is worth mentioning that due to the conceptuality of the work at this stage, the exact nature of some attributes and how they are quantified are still to be realized and therefore here parameters are used instead of real values.

1. An EC proposal can be given to the DSS by the decision maker through the User's Interface. e.g. A computer company wants to replace a 7200rpm Hard Disk Drive (HDD) by a Solid State Drive (SSD) in a notebook product.
2. The first step during the evaluation is to determine the properties of the proposed change. As discussed above. Such properties could be reason, process factors (when and where does the change happen), priority, complexity (functional complexity), architecture (physical complexity), etc. These are just possible examples of attributes and of course in practice there will be more factors. All those properties can be entered by the user through the user interface or depending on the target company's information management infrastructure could be extracted easily from available databases. The following is an example:

Reason	Where	Priority	Complexity	Architecture	Etc..
Better performance	Design phase	Convenience	C	A	

Figure 5. Current EC Attributes

3. According to the attributes, the relevant past EC data are sorted out from the database; in the Rules level of KB and properties of the proposed EC and the past ECs are compared. Also the impacts of the past ECs caused by the attributes are sorted out together. For example two similar past ECs are found:

Reason	Where	Priority	Complexity	Architecture	Etc..
Better performance	Design phase	Convenience	C1	A1	

Similarity	Cost	Lead Time	Functional Impact	Physical Impact	Safety Impact	Etc
80%	+140 Euro	+40 min	FI 1 FI 2 ...	PI 1 PI 2 ...	SI 1 SI 2 ...	

Reason	Where	Priority	Complexity	Architecture	Etc..
Better performance	Design phase	Convenience	C1'	A1'	

Similarity	Cost	Lead Time	Functional Impact	Physical Impact	Safety Impact	Etc
90%	+170 Euro	+30 min	FI 1' FI 2' ...	PI 1' PI 2' ...	SI 1' SI 2' ...	

Figure 6. Past ECs 5400rpm-7200rpm notebook (left) and 7200rpm-SSD desktop (right)

LEFT: There used to be a replacement of a 5400rpm HDD by a 7200rpm HDD in a notebook product. Due to the similar attributes and the fact that, same as the current change, the change happens on a notebook product, this past EC is 80% similar to the current change. This past ECs eventually lead to 140 euro increase in cost, 40 minutes increase in lead time, and there are a few functional impacts, physical impact and safety impact. Here we assume that FI 1, PI 1,and SI 1 are mainly caused by the properties of the notebook itself; FI 2, PI 2 and SI 2 are caused by the replacement of HDD e.g. higher rpm.

RIGHT: There used to be a replacement of a 7200rpm HDD by a SSD in a desktop product. Due to the similar attributes and the fact that, same as the current change, the change is a replacement to SSD, this past EC is 90% similar to the current change. This past ECs eventually lead to 170 euro increase in cost, 30 minutes increase in lead time, and here we again assume that FI 1', PI 1',and SI 1' are mainly caused by the properties of the desktop itself; FI 2', PI 2' and SI 2' are caused by the replacement to SDD e.g. lower rpm.

4. According to the inter-relationship between the impacts and attributes of past ECs and the similarities between the proposed EC and the past ECs, together with the help of the expert knowledge stored as rules in KB and reasoning module, the possible impacts of the proposed EC are predicted. In the given example, with the help of machine learning algorithms, the system can give numerical estimations based on probabilities, e.g. with a 90% possibility the

cost will increases by 200 Euros and the lead time will go up by 45 minutes. As a result, we will have:

Reason	Where	Priority	Complexity	Architecture	Etc..
Better performance	Design phase	Convenience	C	A	

Probability	Cost	Lead Time	Functional Impact	Physical Impact	Safety Impact	Etc
90%	+200 Euro	+45 min	FI 1 FI 2' ...	PI 1 PI 2' ...	SI 1 SI 2' ...	

Figure 7. Impact prediction of current EC

5. The predicted impacts are illustrated to the decision maker. The decision maker decides if the EC proposal is accepted according to the predicted impact from DSS. If yes, EC proposal becomes EC Order and will be implemented.
6. After the implementation, a review shall be done in order to evaluate the correctness, effectiveness and efficiency of the decision making process. Through the machine learning module of the Inference Engine, such feedback is "learned" and stored in the database. In addition, users can also add text written feedbacks to the current EC process in order to present the tacit knowledge for the future use.

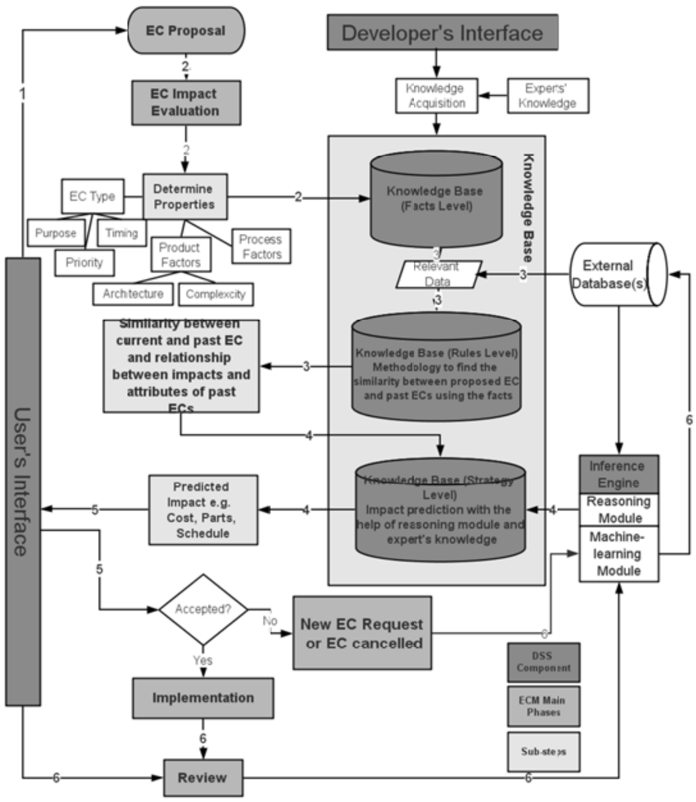

Figure 8. Proposed Information Flow

5. Conclusion and Furture Work

Due to the understood potential for improving the efficiency of the ECM process in time and quality of decisions, in this paper a high level architecture of a possible Decision Support System to assist the decision making in Engineering Change Management process was proposed. Accordingly, the concept is believed to be capable of providing the following advantages:

- Consideration of various variables and utilizing the historical data for solving the current problems in hand
- Storage and reuse of huge amount of knowledge (also tacit) from different experts. This is specifically important in long life-cycle products such as aerospace industry or nuclear power plants where the knowledge of experts after their retirement shall not be lost.
- Customizability of the analysis with respect to the strategic objectives of the organization.
- Dynamic update of knowledge by utilizing the self-learning module.

The authors would like to specifically note that, the accuracy of the system is dependent on the quality and quantity of the data. Thus, in long lifecycle and complex products which incorporate many changes annually, this system will work the best and improves over time.

Although the motivation for such a tool is completely realized, the concept presented in this paper is limited to a theoretical proposal. Therefore, more in-depth research is needed in order to fill the knowledge gap and further examine the feasibility of the proposal and its application in practice. Accordingly, the authors identified the following steps towards finalizing the concept:

- Further research on EC attributes and important impacts as well as their value types in practice
- Research on already existing data sources in organizations and their filtering approaches
- Defining the rules and knowledge model to capture the knowledge from experts
- Development of the Knowledge-base
- Identifying the suitable algorithms for the machine learning module
- Optimization of the User Interface in order to ensure the simple interaction between the system and decision maker
- Identification of ECM Key Performance Indicators in order to enhance the model with KM Evaluation Framework for better ratinga
- Practical implementation and validation of the model

Acknowledgements

The authors would like to thank the EU FP7 Marie Curie Project "PURESAFE" for the funding for this project.

References

Acar, B. S., Benendetto-Neto, H., Wright, I. C., "Design change: problem or opportunity", Proceedings of engineering design Conference '98, Brunel University, UK, Professional Engineering Pulishing, Bury St. Edmunds, 1998, pp. 445-454.

Arian, F. M., Pheng, L. S., "Framework For Developing A Knowledge-based Decision Support System For Management Of Variation Orders For Institutional Buildings", ITcon, Vol. 11, 2006, pp. 285.

Clark, K. B., Fujimoto, T., "Organization and management in the world auto industry", Harvard Business School Press, Boston, 1991.

Clarkson, P. J., Simons, C., Eckert, C. M., 'Predicting change propagation in complex design', Proceeding of ASME design engineering technical conferences, Pittsburgh, USA, CD-ROM, paper no. DETC2001/DTM-21698, 2001.

DiPrima, M. R., "Engineering change control and implementation considerations", Prod. Inventory Manag., Vol 23, No. 1, 1982, pp. 81-87.

Eckert, C. M., Wyatt, D. F., Clarkson, P. J., "The elusive act of synthesis: creativity in conceptual design of complex engineering products", Proceedings of the 7th ACM conference on creativity and cognition, Berkeley, California, USA, 2009.

Fricke, E., Gebhard, B., Negele, H., Igenbergs, E., "Coping with changes: causes, findings and strategies", Syst. Eng., Vol. 3, No. 4, 2000, pp. 169-179.

Gebus, S., "Knowledge-based Decision Support Systems For Production Optimization and Quality Improvement in The Electronics Industry", Oulu University Press, 2006.

Jarratt, T. A. W., Eckert, C. M., Caldwell, N. H. M., Clarkson, P. J., "Engineering change: an overview and perspective on the literature", Research in Engineering Design, Vol. 22, 2010, pp. 103-124.

Jarratt, T. A. W., Eckert, C. M., Clarkson, P. J. "Design process improvement" – Engineering Change, P.J. Clarkson, C.M. Eckert (eds.), Springer, New York, 2004a.

Jarratt, T. A. W., Eckert, C. M., Clarkson, P. J., "Development of a product model to support engineering change management", Proceedings of TMCE 2004, Lausanne, Switzerland, 2004b.

Kocar V., Agunduz A., "ADVICE: a virtual environment for engineering change management. Comput Ind 61(1), 2010, pp. 15-28.

Kohn, A., Lindemann, U., Maurer, M., "Knowledge base for supporting the handling of product models in engineering design", International Conference on Engineering Design, 2013.

Niknam, M., Bonnal, P., Ovtcharova, J., "Configuration Management Maturity in Scientific Facilities", International Journal of Advanced Robotics, to be published on Special issue on Robotics and Systems Engineering in Scientific Facilities, ISSN 1729-8814, 2013.

Niknam, M., Ovtcharova, J., "Towards Higher Configuration Management Maturity", Product Lifecycle Management: PLM for Society, IFIP WG5.1, 10th International Conference PLM13, book edited by Alain Bernard, Louis Rivest, Debasish Dutta, IFIP Advances in Information and Communication Technology, Springer Verlag, ISBN - 978-3-642-41501-2, 2013, pp. 396-405.

Ollinger, G. A., Stahovich, T. F., "ReDesignIT" - A constraint-based tool for managing design changes', Proceedings of ASME design engineering technical conferences, Pittsburgh, USA, CD-ROM, paper no. DETC2001/DTM-21702, 2001.

Pikosz, P., Malmqvist, J., "A comparative study of engineering change management in three Swedish engineering companies", Proceedings of ASME design engineering technical conference, Atlanta, GA, USA 1998.

Power, D. J., "Decision Support System Hyperbook", accessed on 15 October 2013, http://dssresources.com/subscriber/password/dssbookhypertext , 2000.

Reidelbach, M. A. "Engineering change management for long lead-time production environments", Prod. Inventory Manag. J., Vol. 32, No. 2, 1991, pp. 84-88.

Ring, J., Fricke, E., "Rapid evolutions of all your systems - Problems or opportunity", Proceedings of the IEEE17th Digital Avoinics Systems Conference, Seattle, 1998.

Riviere, A., DaCunha, C., Tollenaere, M., "Performances in Engineering Changes Management", proc. of the 4th International Conference on Integrated Design and Manufacturing in Mechanical Engineering, Clemont-ferrand, France, 2002.

Sage, A. P., "Decision support system engineering", Wiley-Interscience, 1991.

Terwiesch, C., Loch, C. H., "Managing the process of engineering change orders: the case of the climate control system in automobile development", J Prod Innov Manag, Vol. 16, No. 2, 1999, pp. 160-172.

Ulrich, K. T., "The role of product architecture in the manufacturing firm", Res. Policy, Vol. 24, 1995, pp. 419-440.

Masoud Niknam, PMP, Research Associate
IMI, Karlsruhe Institute for Technology
Zirkel 2, Geb 20.20. 76131 Karlsruhe.
Telephone: 0721 - 60846641
Email: masoud.niknam@kit.edu
URL: http://www.imi.kit.edu