21ST INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED17
21-25 AUGUST 2017, THE UNIVERSITY OF BRITISH COLUMBIA, VANCOUVER, CANADA

A METHOD FOR THE EXPERT-BASED IDENTIFICATION OF ENGINEERING CHANGE PROPAGATION

Kattner, Niklas; Mehlstäubl, Jan; Becerril, Lucia; Hollauer, Christoph; Weidmann, Dominik; Lindemann, Udo
Technical University of Munich, Germany

Abstract

Handling engineering changes effectively becomes more and more important for the performance of the overall design process. Therefore, it is crucial to assess the propagation of engineering changes precisely to prevent unexpected workload. However, the detailed analysis of technical systems to predict possible change propagation is often time consuming. Furthermore, the effort needed for the analysis highly increases with a growing number of system elements. Hence, methods for the application in the industry have to be both precise in the prediction of change propagation and extremely time efficient. This paper therefore presents a new approach to identify change propagation by combining a matrix-based approach of modeling interrelations with the implicit available knowledge of an expert familiar with the system. Additionally, the system understanding is expanded by introducing artefacts. It remedies the problem of having preliminary development results, e.g. feature specification or preliminary part lists, which also have to be considered in the change propagation analysis. Finally, the approach is evaluated within an industrial use case.

Keywords: Design management, Design methodology, Project management, Engineering change management, Change prediction

Contact:
Niklas Kattner
Technical University of Munich
Institute of Product Development
Germany
niklas.kattner@pe.mw.tum.de

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 21st International Conference on Engineering Design (ICED17), Vol. 4: Design Methods and Tools, Vancouver, Canada, 21.-25.08.2017.

1 INTRODUCTION

Engineering changes were and still are an important part in the design of new products (Jarratt et al., 2005; Maier and Langer, 2011), mainly because they are especially required for error correction and adjustment to new requirements (Lindemann and Reichwald, 1998; Eckert et al., 2004). Due to reasons of cost-effectiveness resulting from the increasing competition on the market, more and more modifications are made rather than new developments and therefore more variants are offered (Pikosz and Malmqvist, 1998). These modifications and variants are designed by means of engineering changes (Clarkson et al., 2001; Rutka et al., 2006). Thus, the number and relevance of engineering changes are also increasing. One of the challenges in dealing with engineering changes is the propagation of them and its resulting impact (Clarkson et al., 2004). Because of the dependencies between the individual elements of a system, changes can propagate and cause further changes (Clarkson et al., 2004). This propagation does not only occur between components, but also for example between components and requirements, or requirements and functions (Mocko et al., 2007). The different elements and dependencies make it difficult to trace change propagation. In addition to the impact within the product, the effects on all other outputs of the design process, such as system descriptions and drawings, must be considered (Pikosz and Malmqvist, 1998). Many of these outputs are provided to customers or used for future design processes.

There are several different methods and approaches to tackle the impact of engineering changes. Two important examples are the Change Prediction Method by Clarkson et al. (2004) and the Change Propagation Analysis method by Rutka et al. (2006). Most of these approaches attempt to convey a quick and reliable first impression about the magnitude of change impact. However, they are extremely laborious when defining and maintaining dependencies and they describe no procedure for the exact determination of the change impact. A precise determination of the propagation and impact of change is necessary for companies as only this enables the scheduled implementation of the engineering change and the consistency of all outputs in the design process. To be economically, the time required for the determination of change effects should be kept as low as possible. For these reasons, most scientific approaches are hardly used in industrial practice. In companies, the change impact is usually estimated by the experience and knowledge of the designers, who do not follow a specific procedure for the impact analysis (Tang et al., 2009).

This paper presents a new approach to determine the impact of engineering changes. In addition to the exact determination, the method is intended to facilitate the identification of the change impact and is executable within a reasonable time. As engineering changes are usually carried out by designers, knowledge about the system is assumed.

2 BACKGROUND

The previous methods and approaches to predict engineering change propagation and the resulting impact can be divided into two categories. These categories are matrix-based and model-based. Clarkson et al. (2004) form with their Change Prediction Method (CPM) the basis of the matrix-based approaches. The CPM is a mathematical method, which enables to predict the risk of change propagation within component level. They determine dependencies between components and define the direct likelihood and impact between two components. With this and the indirect propagation paths, they calculate the combined likelihood, impact and risk. Flanagan et al. (2003) developed a model to trace change propagation within two domains. For this, they consider the interaction between the components and functions and defines features as pairs of components and functions. Rutka et al. (2006) extend the CPM from Clarkson et al. (2004) into their Change Propagation Analysis (CPA) method. The CPA enables designers to determine the change impact for previously defined types of change. In this method, they define items, which can be elements from any domain. For every item, they determine all possible types of change. Ariyo O. et al. (2007) present an algorithm, which allows to assess change impact on different level of granularity. Based on the dependencies between the elements of the highest level of granularity, the dependencies between the elements of smaller level of granularity are calculated. By means of the CPM, the dependencies between the elements of the highest level of granularity are calculated. Koh et al. (2009, 2012) attempt to make direct conclusions from change options to requirements and include direct and indirect effects of change within the product architecture. For this, they define the domains product components, change options and requirements. For each pair of domains, they determine

different dependencies. Bauer et al. (2015) consider the reasons for the individual changes to draw conclusions about change impact and the behavior of product architecture. In their approach, any number of domains with different dependencies can be defined.
One of the most cited model-based approaches is the Change Favorable Representation (C-FAR) method by Cohen et al. (2000). Cohen et al. (2000) attempt to detect the possible change impact. For this, they use an EXPRESS model, which includes entities and their dependencies. The dependencies between two entities are expressed with a C-FAR matrix. Ollinger and Stahovich (2001) developed the RedesignIT tool. This tool uses a qualitative devise model to enable an early assessment of the change impact. In the qualitative devise model, they define so-called quantities, which can be properties of the components as well as device operations. Target is to generate and evaluate all possible redesign plans. The last approach mentioned in this paper is the Change Impact and Risk Analysis (CIRA) by Conrad et al. (2007). It enables the analysis and valuation of change impact. For this, they define characteristics and properties.
In summary, there are existing a number of approaches dealing with engineering change propagation. They are either matrix-based or model-based. They also pursue different goals and consider various domains.

3 MOTIVATION FOR A CHANGE PROPAGATION ANALYSIS

Within the framework of this work, designers of an industrial partner were interviewed. In these interviews, the designers were asked about their handling of engineering change propagation. Furthermore, a few approaches of the existing literature were presented. In this context, the need for a methodical approach to determine the impact of change became apparent and objectives and requirements for such a method were clarified. The primary objective of companies is to keep all outputs of their design process consistent and therefore, the impact of change must be accurately determined and documented. To be economical, this should be done over a short period of time. A change control board uses the results of an impact analysis to decide on the approval of a change and to provide a basis for estimating the possible costs and duration of the implementation. In addition to this, the results will show users all effects to consider during the implementation. Engineering changes are often carried out by designers themselves and for this reason, a procedure can be devised, which requires knowledge about the system and involves the user actively.
The requirements of the industry for such a method have not been fully satisfied by previous approaches. So that such a method is used in companies, the implementation and application must not take too much time. For example, if the CPM of Clarkson et al. (2004) is used in industry, first dependencies must be determined. Afterwards direct likelihood and impact for these dependencies as well as combined likelihood, impact and risk must be specified. In addition to the creation of the matrix, revisions are required after every change because dependencies and elements in the matrix might be changed. The same problems arise with the other approaches mentioned in Section 2, which also require an enormous effort. Another difficulty is that none of the examined methods describes a procedure to determine the exact impact of change. Rutka et al. (2006) indicate a change propagation simulation in which the type of change is selected and all affected elements are identified. However, this is only possible for changes that were previously defined. In practice this is not always feasible as it is not possible for a complex system to predict all potential changes before they occur. The other methods presented in section two can only be used to estimate the magnitude of change impact. However, they cannot be used to determine which specific elements are affected. In these approaches, only the dependencies are identified and further characterized and the type of change determines whether there is a change propagation or not.
In summary, the previous approaches take a lot of effort to determine and maintain the dependencies between elements. Considering complex systems with many elements, this results in either a very small subsystem which can be analyzed or a huge amount of time needed to apply the methods in an industrial environment.

4 IDENTIFICATION OF ENGINEERING CHANGE PROPAGATION

The following section presents the developed method, as well as a closer look on the evaluation at the industrial partner. Therefore, some important terms must be defined. The method in this work considers the impact of engineering changes within a defined system. A system consists of elements and the relations between them. These elements are delimited from their environment by a system boundary

(Lindemann, 2009). The elements of the system can have any degree of detail and can derive from different domains. Domains are views or perspectives of a system (Rutka et al., 2006). Additionally, in this work element types were defined. These are umbrella terms for the different types of elements on the different levels of detail.
Figure 1 shows the relations between domains, element types and elements. In this figure a system is considered from three different domains. These domains are the component view, the functional view and the requirements view. Within these domains there are different element types to describe the system from this perspective on different levels of detail. In a component view the element types could be parts, assemblies and modules. Elements always belong to an element type. For example, a screw would be an element of the element type parts and the individual parts are part of the component view. Each system can contain any number of domains and a domain can contain any number of detail levels or element types and elements. Additionally, element from different domains can have interrelations across the domain boarders.

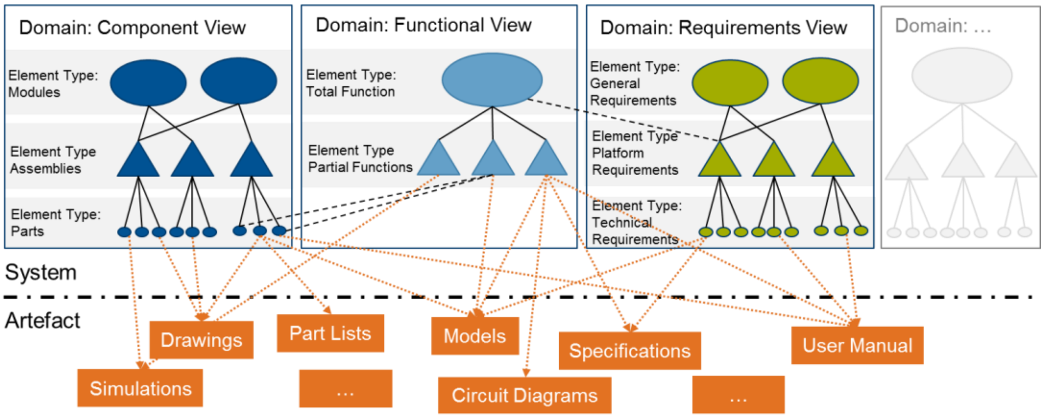

Figure 1. Domains, element types, elements, artefacts and its interrelations

4.1 Artefacts as output of the engineering design process

In existing literature, the main perspective on the propagation of technical changes is the identification of interrelations between components, functions and requirements. However, this is not fully sufficient for the requirements of the industry because many other outputs, such as system descriptions, drawings and simulations, are produced during the design process. Many of these outputs are supplied to the customers, or are reused for modifications of existing products. For this reason, they must be consistent after a change. For example, in software engineering the word artefact is used for documents and programs as well as all of its preliminary results (Stammel, 2015). Hence, to take all outputs of the design process into account, the term artefact is introduced in this method. For the understanding of our research results, we define artefact as followed:

"An artefact can represent any result or preliminary result developed during the product creation cycle."

However, artefacts do not represent an own domain because they contain elements of other domains or are derived from the characteristics of them. The contents of artefacts are not independent from each other because elements can exist in several artefacts and an artefact can also contain elements from several domains. It is also possible to define a whole domain as an artefact. For example, a product represents a system from a component view, but it is also an output of the design process and can be defined as an artefact. In figure 1, the relations between artefacts and elements are visualized. These relations are explained below and the need for the definition of artefacts is discussed.
For example, the elements "Piston", "Cylinder" and "Engine Housing" could belong to the element type "Parts". In an artefact "Drawing: Piston" is the element "Piston" included. In an artefact "Drawing: Engine" is the element "Piston" present, but also the "Cylinder" and the "Engine Housing". An artefact, which contains elements of several domains, is the "Functional Specification". It contains requirements and the functions with which the requirements should be implemented. For example, the requirement "High Compression" could be implemented by the function "Compress Mixture". In figure 1 it can also be seen, that artefacts do not have any dependencies to other artefacts and so changes cannot propagate directly between two artefacts. Change propagation only occurs, if elements are changed. If a change is

made to the artefact "Drawing: Piston", this can only propagate, if the element "Piston" is changed. Without the introduction of artefacts as design outputs, the effort to determine the impact increases. This results from the fact, that artefacts do not contain new information, but are derived from elements of other domains and their relations. Thus, propagation paths are shown several times. This is illustrated in the following section by an example. During a change of the element "Piston" direct impact on the "Cylinder", the "Engine Housing" and the artefacts "Drawing: Piston" and "Drawing: Motor" are determined. Subsequently, the impact from the change to the "Drawing: Engine" is examined. Here the "Cylinder" and the "Engine Housing" are identified as affected by the change. This example demonstrates that artefacts do not provide new information about the system.

4.2 Expert-based change propagation analysis (ECA)

By utilizing artefacts introduced in the previous chapter, we remedy the problem of having way more development results during the development phase than just requirements, functions and components. However, the method has to be efficient in the application within an industrial context. Therefore, we focused on modelling dependencies between element types instead of elements. Basically, this results in a reduction of the effort needed to create the dependency matrix. Since the approach focusses on an industrial context, the reduced level of detail of the matrices can be compensated by a system expert applying the method. A system expert is a person with knowledge about the considered system. Hence, an expert with a good understanding of the system will be both faster and more precise in identifying the actual change propagation. We decided to use matrices in our method, because they allow to display a multitude of elements with their dependencies.

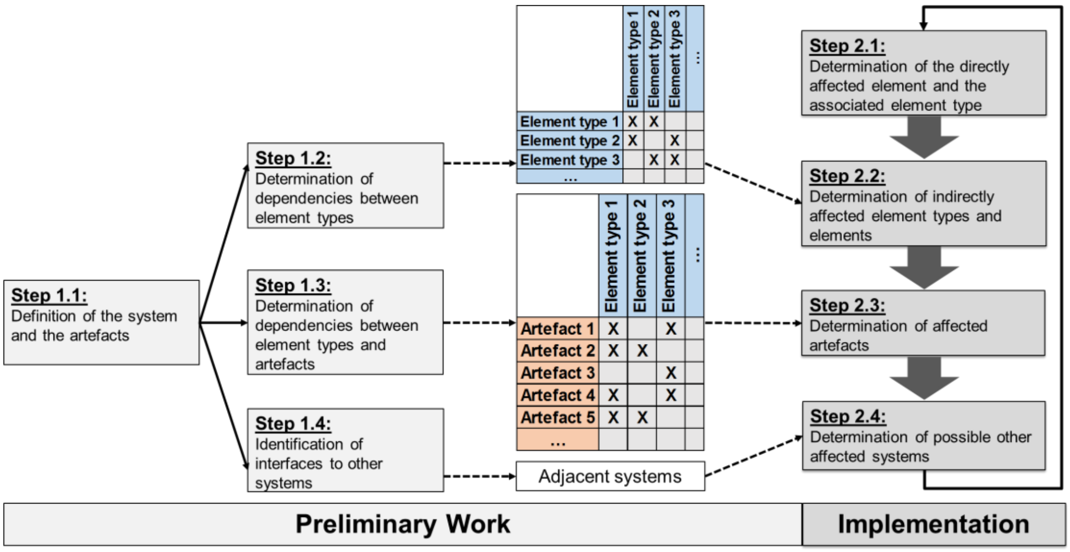

Figure 2. Expert driven change propagation analysis

The procedure of the overall method shows figure 2. The overall method consists of two major phases, which can be further divided into four separated steps. The first of the two major phases is the preliminary work. In this phase, the system and the artefacts are defined. Subsequently, the dependencies between element types and those between artefacts and element types are determined in matrices. In the last step of the preliminary work, the interfaces to adjacent systems are identified. The second major phase is the implementation. In the first step of this phase, the direct affected element and its belonging element type is identified. Subsequently, all indirectly affected element types and elements are determined with an iterative process. Thereby the knowledge of the expert is needed and the defined dependencies between element types serve as an aid. To determine the affected artefact in the next step, also the expert knowledge and the defined dependencies between artefacts and element types are used. Finally, the interfaces to adjacent systems are examined. In step 2 and 3 of the preliminary work, it is possible to scale the effort. Instead of element types, the dependencies between individual elements and those between artefacts and elements can be determined. However, the scaling level of the element types makes it possible to carry out these steps in a very short time, because only a few dependencies have to be determined. The lack of detail is redressed by the expert's knowledge of the system. In the following chapter the method is described only for the scaling level of the element types.

4.2.1 Preliminary Work

Before an expert can start with the actual change propagation analysis, the system of interest has to be detailed. The preliminary work has to be done once to define the relevant domains, element types and artefacts. However, whenever new artefacts emerge or the perspective in the system changes, the preliminary work must be checked for consistency.

Step 1.1: Definition of the system and the artefacts

At first, the system and all artefacts describing the system must be defined. The system consists of domains, element types and elements. In this method, any number of domains can be considered. Furthermore, a domain can include any number of element types and therefore, any number of detail levels. To reduce the complexity in defining the system, a hierarchical model with element types should be created for each domain. In order to recognize all effects, the considered system must be comprehensively covered with all domains and element types.

Step 1.2: Determination of dependencies between element types

In this step the dependencies between element types are determined. An element type depends on another element type, if a change on an element of the first element type can influence an element of the second element type. It is important to consider only direct dependencies. During the evaluation, it was discovered that for experienced developers it is difficult to restrict themselves on direct dependencies. Unconsciously, they include already indirect dependencies. However, this does not make sense and makes it difficult to trace the change propagation during the implementation. The direct dependencies are captured binary in an Multi Domain Matrix (MDM).

Step 1.3: Determination of dependencies between element types and artefacts

In order to be able to infer from element types to artefacts, the dependencies between them are determined in this step. A relation between an element type and an artefact exist, when elements of this element type are available in the artefact. The dependencies are captured binary in several Domain Mapping Matrices (DMMs).

Step 1.4: Identification of interfaces to other systems

In the last step interfaces to other systems are established. A connection to another system obtains, if the change of an element within the considered system can affect an element of the other system. For this reason, it is necessary to create a list of all adjacent systems. This should be supplemented by the exact elements associated with these systems.

4.2.2 Implementation

During the implementation of the method, the accurate impact of an engineering change within the defined system is determined. The implementation has to be repeated for each change and the results of the preliminary work serve as an aid.

Step 2.1: Determination of the directly affected element and the associated element type

At the beginning of the implementation, the element must be identified, which is directly affected by the change. The change to this element attempts to achieve the change target. For this affected element, the element type as well as the nature of change has to be determined. In this paper, the nature of change describes the change to an element.

In figure 3, this first step was carried out and documented in a simplified example of an air pump. Here the directly affected element is "Cylinder Pressure=8bar". This hast to be changed to "Cylinder Pressure=12bar" and it belongs to the element type "Technical Requirements".

Iterations	Element	Nature of change	Element type	Possible affected element types	Indeed affected element types	Element	Nature of change
Iteration 1	Cylinder Pressure=8bar	Cylinder Pressure=12bar	Technical Requirements				

Figure 3. Documentation form for the identification of directly affected elements

Step 2.2: Determination of indirectly affected element types and elements

In this step, all element types and elements that are indirectly affected by the change are determined iteratively. This is the most complex and comprehensive step, which also requires a certain degree of system knowledge. The procedure of this step is visualized in figure 4. Based on the directly affected element, its element type and the nature of change, the first iteration step is started. The direct dependencies of this element type to other element types are demonstrated with the MDM form step 1.2. The dependent element types may be affected. The user of the method decides with his expert knowledge, which of these element types are indeed affected. For this purpose, the individual elements of the element type must be viewed by the user individually. An element type is indeed affected, if one or more elements are affected by the change in the element with which the iteration has begun. The affected elements and their nature of change are captured. These elements are starting point for the next iteration. For each of these new affected elements this whole step is repeated until there is no further impact identified and the change propagation is finished. Impact on elements, identified as affected already, must not be considered further.

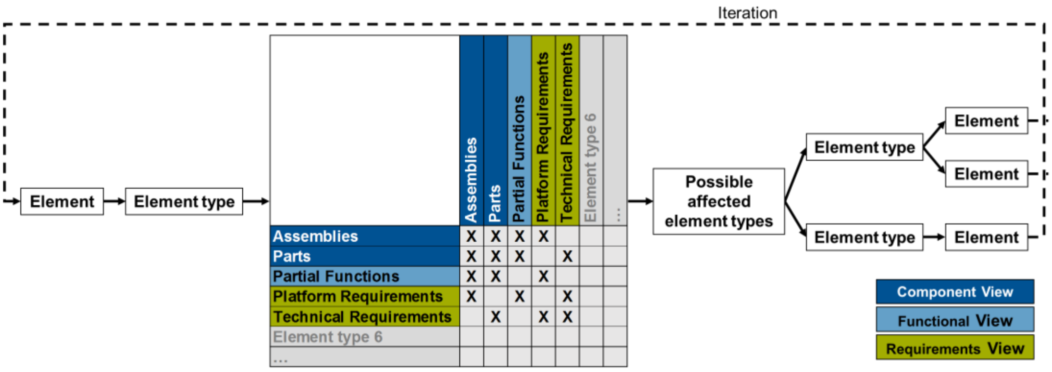

Figure 4. Approach to determine indirectly affected elements

This procedure is presented in the following with a simplified example. The results are documented in figure 5. For this example, the element "Cylinder Pressure=8bar" from element type "Technical Requirements" was already identified as directly affected in step 2.1. Based on this element and element type the first iteration starts. The possible affected element types are "Parts", "Platform Requirements" and "Technical Requirements". These element types were read out from the MDM in figure 4. Subsequently, the user investigates whether the change to "Cylinder Pressure=8bar" has an impact on elements of these element types. Within the element type "Parts" the element "Piston" is identified as affected. The diameter of this element must be increased from 10 to 12mm. Within the element types "Platform Requirements" and "Technical Requirements" no further impact is detected. Starting from the element "Piston", the next iteration step begins.

Iterations	Element	Nature of change	Element type	Possible affected element types	Indeed affected element types	Element	Nature of change
Iteration 1	Cylinder Pressure=8bar	Cylinder Pressure=12bar	Technical Requirements	Parts	no	Piston	ø from 10 to 12mm
				Platform Requirements	no	-	-
				Technical Requirements	no	-	-
Iteration 2	Piston	Ø from 10 to 12mm	Parts	Assemblies	no	-	-
				Parts	yes	Cylinder	ø from 10 to 12mm
				Partial Functions	no	-	-
				Technical Requirements	no	-	-
Iteration 3	Cylinder	Ø from 10 to 12mm	Parts	Assemblies	no	-	-
				Parts	no	-	-
				Partial Functions	no	-	-
				Technical Requirements	no	-	-

Figure 5. Documentation form extended with indirectly affected elements

In the second iteration step, the row of the element type "Parts" is read out. "Assemblies", "Parts", "Partial Functions" and "Technical Requirements" are determined as possible affected. On closer inspection, only impact on the element "Cylinder" of the element type "Parts" was determined. Starting with this element the third iteration step begins. In this step, also the row of the element type "Parts" is read out. This time, no further impact is emanated and this step is finished.

Step 2.3: Determination of affected artefacts
In this step, all affected artefacts are determined. The procedure of this step is visualized in figure 6. First, all columns of the affected element types of step 2.2 are selected in the DMMs. The user has to check whether the selected artefacts are indeed affected. If one of the affected elements from step 2.2 is in an artefact, this artefact is indeed affected by the change. For each artefact, the affected elements and their nature of change must be documented.

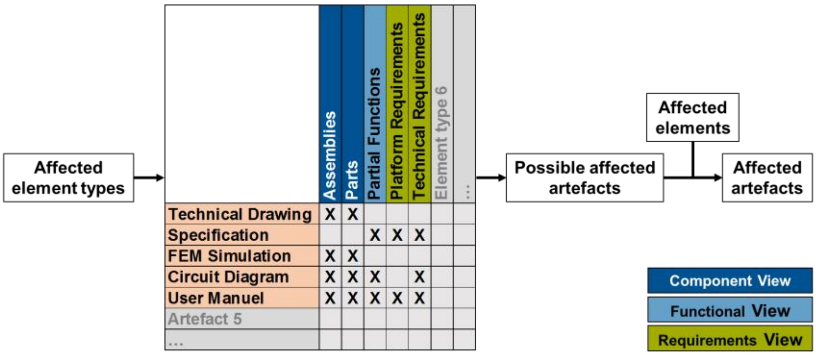

Figure 6. Approach to determine the affected artefacts

In figure 7 a simplified example has been documented. In step 2.2 the element types "Technical Requirements" and "Parts" were already determined as affected. Therefore, in this step first the columns of these element types must be read out from the DMMs. These DMMs are presented in figure 6. The possible affected artefacts are the "Technical Drawing", the "Specification", the "FEM Simulation", the "Circuit Diagram" and the "User Manuel" because they contain elements of the element types "Parts" and "Technical Requirements". After this, the user must check, if there are affected elements in the possible affected artefacts. These affected elements were determined in step 2.2. If this is the case, these elements must be changed within the artefact. The "Technical Drawing" and the "FEM Simulation" contain the "Piston" and the "Cylinder". The element "Cylinder Pressure=8bar" is available in the "Specification". The "Piston", the "Cylinder" and the "Cylinder Pressure=8bar" are in the "User Manuel". The "Circuit Diagram" is not affected by the change because it contains none of the affected elements.

Affected element types	Possible affected artefacts	Indeed affetced artefacts	Elements to be changed in the artefact
Technical Requirements Parts	Technical Drawing	yes	Piston, Cylinder
	Specification	yes	Cylinder Pressure=8bar
	FEM Simulation	yes	Piston, Cylinder
	Circuit Diagram	no	-
	User Manuel	yes	Piston, Cylinder, Cylinder Pressure=8bar

Figure 7. List of all artefacts affected by the change request

Step 2.4: Determination of possible other affected systems
In the last step, the system boundary is examined. For this purpose, the adjacent elements must be compared to the affected elements. If there is an accordance, a new change request must be made to the adjacent system. Beyond the system boundary, only new change requests are created and forwarded to the responsible persons.

4.3 Evaluation

Since the complexity of technical systems increase rapidly, companies try to remedy the increased difficulty to handle such systems by introducing system models representing these systems. The objective of the models and its underlying syntax is to reduce the complexity by introducing various different representations of a technical system, e.g. component view, requirements view or feature view. Since these models are representations of the technical system, they can be a viable support for the identification of possible change propagations. In order to test the feasibility, the presented method was evaluated by an industrial partner by utilizing available system models of several domains.
The company is a supply firm for the mobility sector in Germany and provided a subsystem divided into 78 main functions. The impact analysis was applied to one of these main functions. The system consists

of four domains, nine element types and 350 elements, which where all previously defined by the industrial partner. Within this extensive system, it is difficult to trace change propagation without a methodical procedure. Moreover, the system contains too many elements to apply one of the exiting approaches of the literature. For example Clarkson et al. (2004) needed about 20 hours to create their model for a system with 19 elements. In order to keep all artefacts consistent, the complete system with all domains and element types was considered in the impact analysis. Additionally, all relevant artefacts were captured. These included, among other things, model diagrams, system descriptions and pneumatic schematics. After defining the interrelations between element types and artefacts, six adjacent systems and five different elements in direct contact with the relevant systems were identified in the last step of the preliminary work. This step is particularly important for the system under consideration since all 78 main functions have to be consistent after a change.

The method was subsequently evaluated with several engineers of the industrial partner, which had to carry out an impact analysis for a change independently. The dependency matrices were created in advance and made available to the engineers. Thereafter, they provided feedback to the requirements and their experience. In order to test the impact of system knowledge on the method, the engineers had different knowledge of the considered system. An important issue for a company is the time required to carry out the impact analysis. This time has turned out very different, because the knowledge and experience of the users varied. The experts with system knowledge conducted the analysis much faster than the one without system knowledge. They confirmed that the method reduces the required time to determine the change impact. However, people with insufficient knowledge need much more time to apply the approach in our setting. On the whole, the involved persons where satisfied with the results of the method emphasising the low amount of time for the application as well as the support in identifying the exact elements affected by the change.

5 DISCUSSION

There are three main issues with change prediction approaches when applied in an industrial environment. First, the impact on the final product is often the main focus of the methods. However, it is important for companies to determine the impact on all outputs of the design process, especially preliminary results like simulations or feature specifications. These outputs must still be consistent after a change as they are often reused or delivered to customers. This issue is addressed in the developed method by the introduction of artefacts. The second constraint is the time required to determine the dependencies. Due to complex systems with a high number of elements and the detailed characterization of the dependencies between them, the time required for the application of existing approaches highly increases with an increasing amount of system elements. By considering element types instead of elements and using only binary dependencies, this time is extremely reduced. Furthermore, the lack of detail is then remedied by using expert knowledge to identify affected elements. Additionally, the approaches often describe a procedure to predict the magnitude of the change impact. If the exact propagation path is needed, they often fall short. Since the introduced method uses expert knowledge, a more precise identification of the actual change impact is possible.

A restriction of the method results from the required knowledge about the system and the artefacts. First, the affected elements have to be determined within the element types and afterwards the artefacts have to be investigated. Dependencies can be described in any detail, but the propagation of change must always be decided in a case-specific manner.

6 CONCLUSION AND OUTLOOK

Within the scope of this research, a methodical approach to determine the impact of engineering changes is introduced. The method is specifically tailored to the requirements and constraints of an application in an industrial context. The requirements and constraints were clarified with the help of an industrial partner and compared with existing approaches to predict change propagation. An additional opportunity of the method is the capture of change impact in all outputs of the design process by introducing artefacts. The term of the artefacts allows a clear and targeted trace of change impact within all preliminary results of the design process. The method developed in this paper allows the accurate determination of change impact in a short period of time. It guides the user through the procedure and reveals possible impacts. However, to achieve the efficiency in the application, an expert has to decide individually, whether there occurs a propagation of a change or not. The accurate determination of change impact allows to keep all outputs of the design process consistent. The ability to define any number of domains and levels of detail makes the method applicable to all systems which has to be further evaluated. However, the method has to be further tested to refine the definition of domains in various industries. Within the evaluation of this paper, the method was applied to identify changes within several system models. Thus, an application of the method in smaller companies without these models could reveal improvements in the artefact definition phase. In addition to the outputs of the design process, effects on organizational units and other processes in the company could be investigated and integrated into the method. The method can also be implemented in a tool to support the user and make the impact analyses more economically.

REFERENCES

Ariyo Owolabi, O., René, K., Eckert Claudia, M. and John, C.P. (2007), "Predicting change propagation on different levels of granularity: an algorithmic view", *Guidelines for a Decision Support Method Adapted to NPD Processes.*

Bauer, W., Nepomuk, C., Lindemann, U. and Maurer, M. (2015), "Domain-Spanning Change Propagation in Changing Technical Systems", *Complex Systems Design & Management Asia*, p. 111. https://doi.org/10.1007/978-3-319-12544-2_9

Clarkson, P.J., Simons, C. and Eckert, C. (2001), "Change prediction for product redesign", *Int Conf Eng Des (ICED'01),* Glasgow, UK.

Clarkson, P.J., Simons, C. and Eckert, C. (2004), "Predicting change propagation in complex design", *Journal of Mechanical Design*, Vol. 126 No. 5, pp. 788–797.

Cohen, T., Navathe, S.B. and Fulton, R.E. (2000), "C-FAR, change favorable representation", *Computer-aided design*, Vol. 32 No. 5, pp. 321–338. http://dx.doi.org/10.1016/S0010-4485(00)00015-4

Conrad, J., Deubel, T., Köhler, C., Wanke, S. and Weber, C. (2007), "Change Impact and Risk Analysis (CIRA)–Combining the Cpm/Pdd Theory and Fmea-Methodology for An Improved Engineering Change Management*", Guidelines for a Decision Support Method Adapted to NPD Processes.*

Eckert, C., Clarkson, P.J. and Zanker, W. (2004), "Change and customisation in complex engineering domains", *Research in engineering design*, Vol. 15 No. 1, pp. 1–21.

Flanagan, T.L., Eckert, C.M., Eger, T., Smith, J. and Clarkson, P.J. (2003), "A functional analysis of change propagation", *DS 31: Proceedings of ICED 03, the 14th International Conference on Engineering Design*, Stockholm. 2003.

Jarratt, T., Clarkson, J. and Eckert, C. (2005), "Design process improvement", *Engineering change*, Springer, pp. 262-285.

Koh, E C Y, Caldwell, N H M, Clarkson, P J (2009), "Using a Matrix-based Approach to Model Change Propagation", *DSM 2009: Proceedings of the 11th International DSM Conference*, pp. 271–284.

Koh, E.C.Y., Caldwell, N.H.M. and Clarkson, P.J. (2012), "A method to assess the effects of engineering change propagation", *Research in engineering design*, Vol. 23 No. 4, pp. 329–351.

Lindemann, U. (2009), *Methodische Entwicklung technischer Produkte: Methoden flexibel und situationsgerecht anwenden*, VDI-Book, 3.th ed., Springer-Verlag Berlin Heidelberg, Berlin, Heidelberg.

Lindemann, U. and Reichwald, R. (1998), *Integriertes Änderungsmanagement*, Springer Berlin Heidelberg, Berlin, Heidelberg.

Maier, A. and Langer, S. (2011), "*Engineering change management report 2011: survey results on causes and effects, current practice, problems, and strategies in Denmark"*. Technical University of Denmark (DTU).

Mocko, G.M., Fadel, G.M., Summers, J.D., Maier, J. and Ezhilan, T. (2007), "A systematic method for modelling and analysing conceptual design information", *DSM 2007: Proceedings of the 9th International DSM Conference*, Munich, Germany, 16.-18.10. 2007.

Ollinger, G.A. and Stahovich, T.F. (2001), "RedesignIT-A constraint-based tool for managing design changes", *Proc. ASME International Design Engineering Technical Conferences, Design Theory and Methodology*.

Pikosz, P. and Malmqvist, J. (1998), "A comparative study of engineering change management in three Swedish engineering companies", *Proc. ASME International Design Engineering Technical Conferences, Design Theory and Methodology.*

Rutka, A., Guenov, M.D., Lemmens, Y., Schmidt-Schäffer, T., Coleman, P. and Riviere, A. (2006), "Methods for engineering change propagation analysis*". 25th Congress of the International Council of the Aeronautical Sciences*, Hamburg, Germany. 2006.

Stammel, J.J. (2015), "*Architekturbasierte Bewertung und Planung von Änderungsanfragen*", Dissertation, Institut für Informatik, Karlsruher Institut für Technologie (KIT), Karlsruhe, 2015.

Tang, D., Zhu, R., Tang, J., Xu, R. and He, R. (2009), "*Product design knowledge management based on design structure matrix*", Advanced Engineering Informatics, Vol. 24 No. 2, pp. 159–166. design structure matrix", Advanced Engineering Informatics, Vol. 24 No. 2, pp. 159–166.