Predicting Requirement Change Propagation Using Higher Order Design Structure Matrices: An Industry Case Study

This research examines the use of a higher order Design Structure Matrices (DSM) as a requirements change modelling tool to predict requirement change propagation in an industry case study of two large scale design projects. The case studies presented in this paper explore the use of a requirements change to predict engineering change propagation. Most design projects tend to evolve as it is an iterative process and, as a result, requirements will do the same. Changes in requirements can propagate to several other requirements of different sections of the system which may further lead to increases in the project cost and lead time. Thus, it is essential to predict the change propagation due to requirements as it enables the designers to foresee unanticipated changes and maximizes the probability for the project's success.

The studies revealed second order relationships, those relationships were intermediary requirements are needed to relate requirements, were influential in predicting requirement change propagation. Rarely was unforeseen propagation occurring in first order form, rather it was occurring in second order. Further, the studies revealed modelling requirements change can expose secondary relationships early in the engineering change definition process that could enhance the decision making process and, more specifically, augment the cost estimations. Designers and engineers are not able to intuitively predict changes in the second order form, especially for complex systems which may have hundreds or thousands of requirements. This introduces an interesting dynamic to propagating requirements that cannot be recognized by simple designer attentiveness to change; rather the use of change modelling tools is needed. A modelling tool, such as that proposed in this paper, can provide the designer insight as to the requirements which may be affected before approving an engineering change.

Keywords: Requirement change, Requirement change Management, Requirements engineering, Engineering changes, Engineering change management, emergent changes

1. Need for managing requirement change

Requirements play a critical role within any design process [1,2,3,4,5] as they are one of the initial documents needed in the design process and are maintained throughout the process to ensure project completeness. Requirements define what stakeholders such as users, customers, suppliers, developers, and manufacturers need and how each is satisfied [5]. Thus, one of the initial steps in the design process is to correctly identify and specify the system's requirements because their use and proper maintenance is crucial to the success and efficiency to any design project [6].

Most design projects tend to evolve as it is an iterative process and, as a result, requirements will do the same [7,8,9]. This evolution includes a change in the requirement's structure, abstraction, and quantity over time as a design project progresses from its conceptual to detailed stages. This requirement change is expected to occur within any stage of the product life cycle and may cause undesired uncertainty and complexity within the design process [10]. Requirements change may result in the underlying design targets to diminish or, in some instances may cause project failure [7,11]. Many of the costs involved with managing requirements are a result of change that occurs and the lack of preparation for such change earlier. It has been recognized that requirement change and its management can be expensive and time consuming [12,13]. The greatest proportion affecting requirement costs can be traced to requirement change management [14,13].

Changes in requirements can propagate to several other requirements of different components, subsystems, or sections of the system which may further lead to increases in the project cost and lead time. Thus, it is essential to predict the change propagation due to requirements as it enables the designers to foresee unanticipated changes and maximizes the probability for the project's success. Hence, this study examines the use of the Design Structure Matrices (DSM) as a requirements change modelling tool to predict requirement change propagation in an industry case study of two large scale design projects.

This study will also explore different types of requirement relationships in developing a DSM and each requirement relationship will be evaluated for its ability to propagate requirements. In performing this study, the following research questions are addressed:

(1) Can a higher order DSM be used to predict change propagation due to requirement change?

(2) How do different types of requirement relationships affect the ability to propagate requirement change?

In addressing these questions, two industry projects examining requirement change will be presented where a case study research method is used in addressing these research questions. This is performed through a retrospective analysis of two design projects in an industrial corporation. The objectives of this study align closer with those of case study analysis than user studies in which patterns are sought that might be suggestive and foundations for subsequent experimental studies [15,16,17].

2. Requirements research

The Rational Unified Process defines a requirement as “a condition or capability to which a system must conform; either derived directly from user needs, or stated in a contract, standard, specification, or other formally imposed document” [18]. The end goal of requirements elicitation is statements which identify critical attributes, characteristics, capabilities, or functions of the design [19]. This is used within the design process as a guide for specifying what the system must accomplish.

Requirements play an important role as they are the fundamental information required throughout the design process [1,2,3,4,5]. It is widely accepted that requirements development is an integral part of the design process as they often are necessary inputs to different design tools and methods, such as QFD, pair wise comparison, and decision matrix. A requirement list not only reflects the initial position but, since it is continually reviewed, serves as an up to date working document of the design process [1]. As a result, ensuring the requirements elicited are correct, complete, and properly managed is instrumental to any design process.

2.1. Requirements change

Design is a complex and dynamic process [20,21,22,23]. A requirement document is not fixed with respect to the data it contains and is used by multiple individuals over the span of a project. Research within the field of requirement change assists in understanding how requirements and their change shape the design process [24]. Throughout the design process, it has been shown that more than half of a system's requirements will change before completion [25,26]. This is partially due to the inaccuracies subjected to requirements during their elicitation, interpretation, and management [27].

Depending on the type of project at hand, requirements may change internal or external to a project [10,28,29]. For example, an internal requirement change may occur when a subsystem requires greater packaging space and results in a design change, which subsequently leads to a change in requirements. An external change may occur when government regulations change on a vehicle safety standard. Changes may be initiated by an engineering redesign, the customer's ever changing needs, competition, or the need for internal improvement [30,31]. Additionally, changes in the understanding of the problem or internal effects such as budget considerations can also cause requirement changes [32].

Requirement changes also introduce negative consequences such as increased complexity [33,34], potential data loss [35], and cost and time wasted [27]. A designer could save time and money if it were possible to make a quick, yet accurate, assessment about the overall effects of a design or requirement change before making a commitment to implementing the change [36]. The result of requirement changes introduces many challenges that become barriers for successfully completing a design project that accurately meets the needs of the client. Thus, a need for a requirements management tool to mitigating the negative consequences of requirement change propagation is needed [13,37].

While the occurrence of requirement change has been recognized, the managing and modelling of this change has not been thoroughly researched [7,38,39]. Great difficulty is involved with managing requirements change, specifically with how this change can be modelled. This is due to their evolving and dynamic nature and the many characteristics that could be modelled, including change type, requirement relationship, and impact of change.

2.2. Requirement change propagation in design

Change propagation is a process in which a change to one element of a design results in additional changes either within or different parts of the design when otherwise this change would not have been required [40,28]. In many instances, the change initiator is not aware of the propagation consequence of the change [22]. Change propagation research stems from studies performed in change management, engineering design, product development, complexity, graph theory, and design for flexibility [41,42]; however, none target the use of requirements as a means for managing change.

Relevant methods for predicting change propagation in design have appeared, primarily in the field of software engineering [43,44]. These methods decompose a program into pieces that are then linked in a propagation graph. Within mechanical design, these pieces might be subsystems or specific components. However, this method of breaking down a system is not detailed as one component or "piece" may consist of dozens or hundreds of requirements. Nonetheless, the technique of decomposing a system down into pieces aims to highlight where subsequent, immediate changes might be necessary, presupposing that the subsystems are generally independent. In software redesign programming variables are relied upon to indicate changes. They are not appropriate for mechanical design where the parametric links between parts may be less explicit or the prediction of change involves more than one step [10]. Furthermore, predicting change in complex systems such as automobiles is difficult as the consequences of change are often hard to predict, especially when device subsystems cross boundaries [36].

Requirement changes are one of the reasons for engineering changes (ECs) where ECs are defined in multiple ways by different researchers. However, in this research the following definition is used [45]: "An engineering change is an alteration made to parts, from embodiment design stage to production stage of the product life cycle, in its form or fit or function, drawing or software that has already been released. The change can be of any size or type, can involve any number of people, and can take any length of time." These changes occur when companies request changes to products, documents, components, manufactured or purchased parts, processes, or even supplies [46]. Hence, research related to engineering change has been reviewed for different change propagation models. The majority of the work performed in engineering change has been to define and characterize engineering change propagation [40].

2.2.1. IBM DOORS

Though few, there are possible approaches for monitoring requirements change. IBM Rational DOORS is a Requirements Management tool that allows users to input different sets of requirements within its database [47]. Data is stored in Rational DOORS through modules, which list a specific requirement list. For larger projects, it is beneficial to have multiple modules to accommodate all the requirements needed such as architectural, system, and user requirements, so that they may be appropriately classified. Rational DOORS is able to develop relationships between requirements, however it has some shortcomings in its inability to differentiate requirement relationship types. For instance, two requirements may possess a function to function relationship, both serving the same function such as converting energy, while another two requirements may have a component to competent relationship, both relating to a specific component. Within Rational DOORS this is noted as a requirement relationship. No type or weighting or relationship exists, an aspect this paper has recognized the need for.

2.2.2. Change FAvorable Representation (C-FAR)

Another change propagation tool is the Change FAvorable Representation (C-FAR), a new and different methodology of representing design information so that changes can be dynamically anticipated and evaluated [48]. Fundamentally, C-FAR uses attributes, elements and relationships as the foundation for its model. It uses a schema that defines the main entities, relations that describe the connectivity between entities; and attributes, which describe the entities [48,10].

C-FAR is effective in computationally measuring the affect of one attribute to another using its matrix relationship. However, C-FAR uses an existing product information model to facilitate change representation, propagation, and qualitative evaluation. Unfortunately, product information may not always be available during the design process. Requirements, however, are an initial document generated and will always be available. While it may constantly change, its existence is nonetheless evident

2.2.3. Higher order DSMs

Change propagation has been predicted using a Design Structure Matrix (DSM) in complex systems termed change prediction model (CPM) [28]. CPM uses the probability of change in a subsystem area on others as elicited by experienced engineers. Although this model is capable of predicting the likelihood of changes in other subsystem areas, it is not to the resolution of engineering requirements, as it investigates specific subsystem areas, whereas requirements may be able to identify the specific features of the component. Therefore, a more systematic approach from the requirements paradigm is sought for which higher order DSMs are explored for its ability to predict change propagation through requirements.

This work was motivated by Delta DSMs by Giffin [28] which illustrates how the actual changes differed from the baseline DSM developed. This paper makes use of DSMs differently as a DSM is used here by viewing how change in one element of a DSM can propagate throughout a system. This propagation is illustrated through a higher order DSM. DSMs have been used to model change propagation before [49,50,51,45,52], however it has yet to be performed formally through the use of requirements. The advantage of using requirements is it does not depend on component architecture, allowing designers to use it early in the conceptual design phase.

2.3. Modelling change through higher order DSMs

One of the few tools available to model change is a higher order DSM, derived from a Design Structure Matrix (DSM). A DSM is used to develop relationships between subsystems, components, or requirements [53,54,55]. In the case of this study, where requirements are related, the DSM functions by listing the requirements on both axis and highlighting all cells where the requirement on a row is related to a requirement on a column. DSM can assist the designer by providing them with a means for modelling, visualizing, and clustering relationships between design elements, such as requirements. DSMs provide a tool for identifying the parts of a product or design and the parametric relationships between them [56,57]. DSMs used in the study presented in this paper are shown in Figure 6, Figure 8, and Figure 9. Each cell in the matrix may contain a numerical or binary representation of the link between one in a row to another column heading [10]. A DSM may not necessarily be symmetric based on the directionality of the relationships. For instance, requirement A may influence requirement B, but not vice versa, causing asymmetry. In the DSMs presented in this paper, a cell highlighted in green indicates a relationship exists between requirements.

A DSM here can be considered as a zeroth order DSM as it serves as a baseline matrix. In order to create a map of how changes affected the system, a higher order is created. A higher order DSM may be that of a first, second, or third order depending on the complexity, population, and coupling of the requirements. An example of a higher order DSM is shown in Figure 10 and Figure 11. While a higher order DSM is capable of propagating requirement changes, its current limitation is it cannot predict the addition of a requirement. The inability to predict requirement additions is due to the lack of relationship with a requirement that has yet to exist. For example, a requirement regulating the use of an electric motor may change to adapt an internal combustion engine instead. This propagation may lead to subsequent changes; however, the addition of a requirement to sanction exhaust emissions cannot be predicted. A higher order DSM may be interpreted in multiple manners as the propagation modelled may be in multiple orders. The example shown in the following section illustrated how relationships are identified through multiple orders.

2.4. Example of modelling change through a higher Order DSM

First order relationships are those which a requirement is directly related to another requirement. These are highly dependent on how relations are formed. For example, a requirement may be related to another requirement because they deal with the same component, or they share a similar function. As seen in Figure 1, a DSM is used to represent the relations between five requirements, A through E. All cells highlight in green indicate a relationship between requirements, such as that between C and E. The original DSM may be considered a zeroth order relationship matrix. If a change is made to requirement E, all immediate relations are highlighted in red, as seen in Figure 2. These relationships are terms first order relationships because of their direct relation with the requirement changed. As seen, requirement E has one first order relation, being with requirement C. Second order relations are those illustrated in Figure 3 where due to the potential propagation to requirement C, all requirements related to requirement C are highlighted. This indicates that requirement D is second order related to the requirement E, where requirement C acts as the mediator. All the higher order DSMs used in this paper will use red and yellow to indicate first and second order relationships respectively.

	Req A	Req B	Req C	Req D	Req E
Req A	🟦	🟩		🟩	
Req B	🟩	🟦			
Req C			🟦	🟩	🟩
Req D	🟩		🟩	🟦	
Req E			🟩		🟦

Figure 1: Example baseline DSM (zeroth order)

	Req A	Req B	Req C	Req D	Req E
Req A	🟦	🟩		🟩	
Req B	🟩	🟦			
Req C			🟦	🟩	🟥
Req D	🟩		🟩	🟦	
Req E			🟥		🟥

Figure 2: Example First Order DSM

	Req A	Req B	Req C	Req D	Req E
Req A	🟦	🟩		🟩	
Req B	🟩	🟦			
Req C			🟦	🟨	🟥
Req D	🟩		🟨	🟦	
Req E			🟥		🟥

Figure 3: Example Second Order DSM

3. Industry case study catalyst

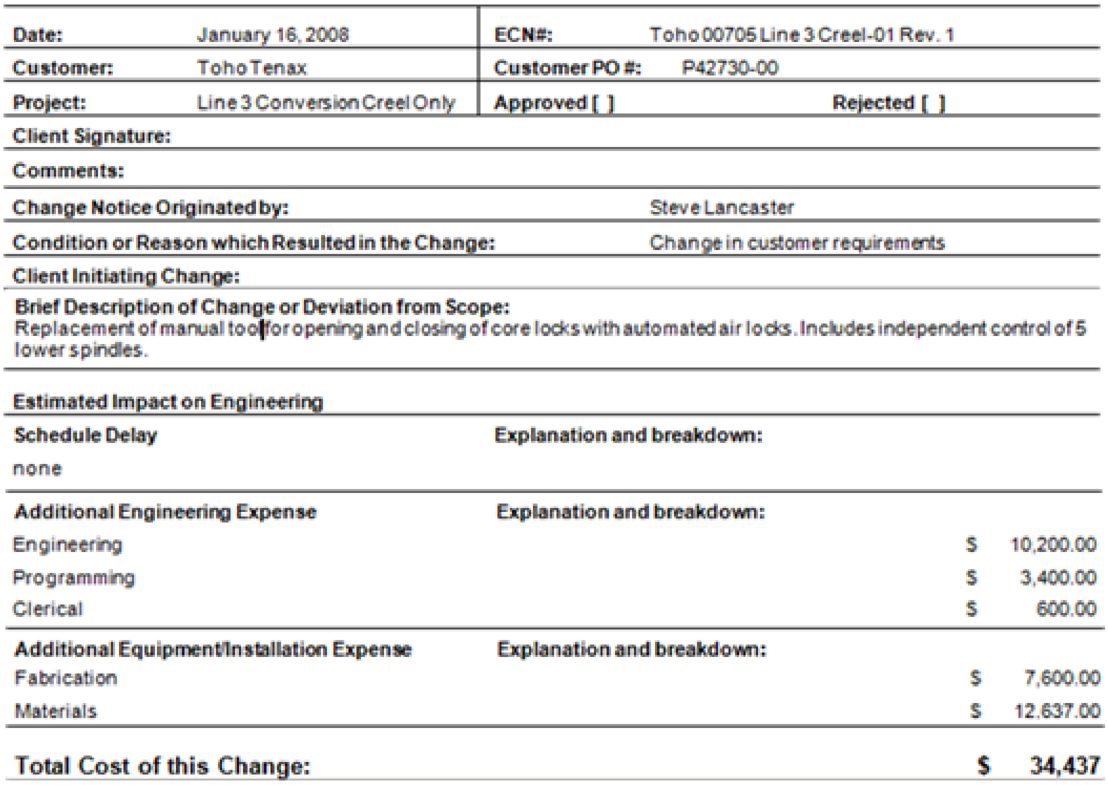
This study examines the potential to predict requirement change propagation through the use of a higher order DSM in two industry design projects. Initially, a case study of their data management system was performed to view how requirements are managed within their system and design process [35]. The corporation provided the authors full access to their design data, which included requirements and engineering changes. The results indicated that requirements were neglected and not properly used, validated, and reviewed. Specifically, requirements were provided by a client and never examined after project initiation. A subsequent case study project, motivated by these findings, was performed to analyze requirements propagation in an effort to predict engineering change [27]. The results of this second historical case study indicated engineering change may be predicted through use of requirements change.

The engineering corporation used in the study is located in Greenville, South Carolina. It is a 60,000 sq. ft. manufacturing facility developing automation solutions. The life cycle of the products they manufacture range from 10 to 20 years. The corporation performs its own fabrication and assembly with non-automated manufacturing systems, employing over 60 associates including engineers, project managers and business managers. The number of associates involved and their role will vary depending on the size and scope of the project. On average, fifteen associates will be involved with a project with each associates working on multiple projects in parallel.

All data pertaining to engineering change was localized within engineering change notifications (ECNs) forms documented by the corporation. There is external information located in emails between the client and corporation where discussion of the change takes place. When a change is initialized, the corporation collects this information from the client and summarizes it in an ECN. This ECN form is exchanged and negotiated with the client until a final change is approved by both parties. The changes which took place in this study were all initiated by the customer, as the customer required a specific change and requested the cost and time delays associated with the change.

Changes in requirements can provide an indication of change propagation. In order to study change propagation, the documents which pertain to engineering change are sought. The key document within the corporation files pertaining to engineering change are the ECNs. The specific detail that is analyzed within ECN is the cause of each change. The ECN form will be detailed to describe the information it contains and its relevance to this research after a brief description of the engineering change (EC) process within the corporation.

A change is initiated when a manager within the corporation or the client identifies a change required in the system. For example, the client may wish for their manufacturing equipment to carry more pallets on its line. The change starts through an exchange of conversation between the corporation and the client. At this time, an ECN form (Figure 4) is documented which details the change information and all related monetary and time delays. If this is approved by the client, a permanent design is developed to address the change in the final design of the system. The ECNs must be documented by the project or operations manager of the project who is also responsible for contacting the client to ensure ECN completeness and approval. The initiator of the ECN is the only associate allowed to make any further changes to it before approval. ECNs are stored within the data system of the corporation where they are accessible to all associates for viewing.



Date:	January 16, 2008	**ECN#:**	Toho 00705 Line 3 Creel-01 Rev. 1
Customer:	Toho Tenax	**Customer PO #:**	P42730-00
Project:	Line 3 Conversion Creel Only	**Approved [ ]**	**Rejected [ ]**
Client Signature:			
Comments:			
Change Notice Originated by:		Steve Lancaster	
Condition or Reason which Resulted in the Change:		Change in customer requirements	
Client Initiating Change:			
Brief Description of Change or Deviation from Scope: Replacement of manual tool for opening and closing of core locks with automated air locks. Includes independent control of 5 lower spindles.			
Estimated Impact on Engineering			
Schedule Delay		**Explanation and breakdown:**	
none			
Additional Engineering Expense		**Explanation and breakdown:**	
Engineering			$ 10,200.00
Programming			$ 3,400.00
Clerical			$ 600.00
Additional Equipment/Installation Expense		**Explanation and breakdown:**	
Fabrication			$ 7,600.00
Materials			$ 12,637.00
Total Cost of this Change:			**$ 34,437**

Figure 4: Corporation’s Engineering Change Notification (ECN) Form

The ECN form contains pertinent information such as the initiation date of the ECN, the change originator (client or customer), the specific project, a unique ECN tracking number, the condition or reason for change, and a status of approval or rejection. In many instances it was found that a “change in customer requirements” was the condition or reason for change. However, the specific requirement changed is not specified. An ECN does not require approval to be documented as many ECNs are initiated but never implemented. A textual description of the change explains the specifics of the change and what must be done to satisfy the ECN so it may be addressed by the engineers. The “Impact on Engineering” section highlights the delays and expenses that will result from this ECN. Delays may result from additional time needed to complete or revise the affected subsystem or the addition of a new component. In addition to delays, anticipated engineering expenses incurred may include addition costs from changes in manufacturing, assembling, clerical, or programming. A similar category is the equipment and installation expenses which details additional expenses such as material or fabrication. Each ECN indicates a total cost for the requested change. This cost of change is based on the requested change and what the engineers anticipate will change to other components or subsystems as a result. When the corporation completes this form, it is sent to the client so it may be approved before any changes can be implemented. By approving, the client is willing to absorb the delays and pay additional expenses due to the change. All ECNs were categorized as "pending changes" for those changes awaiting approval and "approved changes" for those changes approved by the client and automation corporation.

The ECN is analyzed to identify which requirements this change affects. This required viewing the client requirements and identifying where a requirement may be influenced by this type of engineering change. The author of the paper familiarized himself with the requirements as to correlate each ECN to a requirement or set of requirements. The requirements are written in a hierarchical format making the identification of relevant requirements convenient. For example, a change relating to wiring would be located within the electrical controls grouping of the requirements document; however this was not the case for all ECNs. After completing an initial DSM, illustrating the relationships between all requirements, a higher order DSM is created for each of the affected requirements. This is performed to view if subsequent ECNs can be predicted through the requirement propagation indicated in the higher order DSMs.

A typical change propagation that occurred at the corporation is illustrated in Figure 5 along with the proposed method to predict change propagation. Currently there were instances where change propagation did occur unnoticed and an ECN is implemented at a later time. In the proposed method, the higher order DSMs are used to predict the subsequent changes that may occur. It is important to note that not all ECNs are due to propagation, some may come in the form of a change independent of any prior changes [45,58].

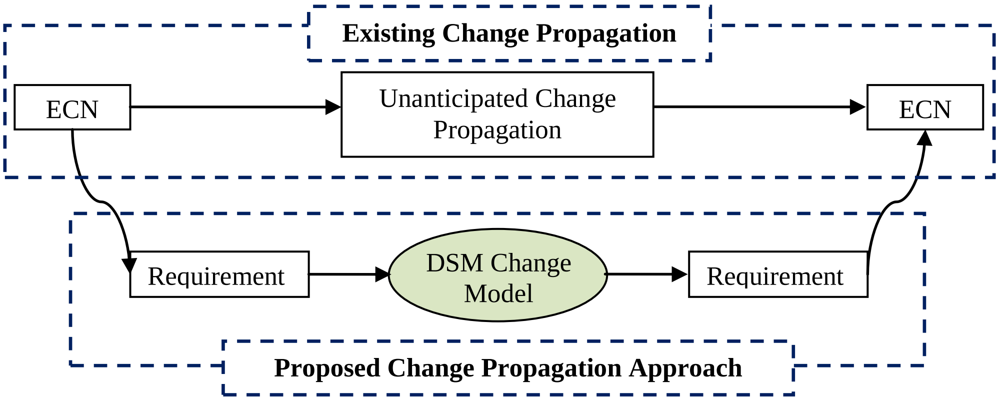

Existing Change Propagation
ECN
Unanticipated Change Propagation
ECN
Requirement
DSM Change Model
Requirement
Proposed Change Propagation Approach

Figure 5: Proposed ECN Prediction Method

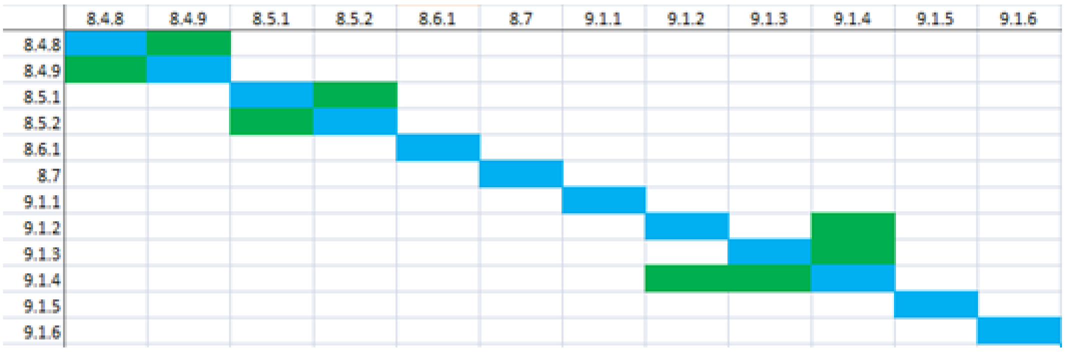
If the ECN results in a change in requirements, these requirements are processed through the change model. This model includes propagating the requirements using a higher order DSM for each affected requirement to predict subsequent changes. From this higher order DSM the potential requirement(s) which may change as a result of an initial requirement change is identified. As this study is of a historical nature, subsequent ECNs are analyzed to view if they have affected requirements which were predicted by the higher order DSM.

The DSM relationship categorized between requirements is of great significance in predicting the propagation of requirements. The initial study used requirements relationships based on their syntactical subject, or in most cases, their component. A syntactical relationship is initially explored as it could be easily automated in future applications. An example of this automation through linguistic analysis of requirements is developed by Lamar and colleagues [59,60] which automations the extraction of a subject from a requirement. The requirement relationships used in the second study presented in this paper make use of keywords.

The scope of this study is that of contractual engineering firms. This study does not explore requirements propagation in innovative product development or design firms. Rather, the requirements documents used here are well defined and specific. An assumption made here is that all requirement changes can be propagated.

4. Subject based case study

The project duration of the initial study spanned approximately fifteen month and included fifteen managers, engineers, and business associates [27]. The engineers were responsible for working on a specific subsystem of the product. The project client provided the corporation a contract incorporating 160 requirements. A DSM was developed for the requirements based on their subject or component/system relationship. A small segment of the DSM is shown in Figure 6. The requirements are listed in both columns and rows. In this DSM, the relationships are bidirectional which creates a symmetric DSM because of the shared subject between related requirements.



	8.4.8	8.4.9	8.5.1	8.5.2	8.6.1	8.7	9.1.1	9.1.2	9.1.3	9.1.4	9.1.5	9.1.6
8.4.8	🟦	🟩										
8.4.9	🟩	🟦										
8.5.1			🟦	🟩								
8.5.2			🟩	🟦								
8.6.1					🟦							
8.7						🟦						
9.1.1							🟦					
9.1.2								🟦		🟩		
9.1.3									🟦	🟩		
9.1.4								🟩	🟩	🟦		
9.1.5											🟦	
9.1.6												🟦

Figure 6: Small Segment of DSM for Subject Based Study

Three approved ECNs were recorded within the system, though only two are analyzed due to missing information within one. The changes did not specifically state a requirement change; rather they requested a change in the design of component, subsystem or a specific design approach. Thus, each ECN was analyzed to extract affected requirements to create a higher order DSM. From this model, the requirements that were potentially affected by the changed requirements are illustrated through the relationships. For example, in Figure 6, if requirement 8.5.2 is changed, then requirement 8.5.1 may be affected as they are directly related, causing a first order relationship.

Following this approach, ECN-03, filed on 2008.10.02 specified the need to "fabricate additional combs" which is associated with requirements 9.3.9 and 9.3.10. The affected requirements stated:

> *9.3.9: A **Yarn Comb** for (22) ends shall be provided for each layer of bobbins.*
> *9.3.10: A **Yarn Comb** for (220) ends shall be provided for each of the two (2) PAN sheets.*

The combined results of requirements 9.3.9 and 9.3.10 higher order DSMs revealed there were 14 first order requirement relations through 27 relations. In the second order, there were 43 requirement relations through 247 relations. Based on these propagation results, first and second order relationships propagated to 9% and 24% of the total system requirements, respectively. This meant at the second order, the changes to the requirements affected in ECN-03 could propagate to 24% of the requirements document.

ECN-04, approved 2008.11.07, specified the need to "install and fabricate mounting brackets for the combs." ECN-04 is analyzed for its affected requirements and whether these requirements were related to the requirements affected in ECN-03. In interviewing the engineers, the mounting of the combs was not considered at the time of the change in the number of combs (ECN-03). The higher order DSM developed for the requirements affected in ECN-03 (requirements 9.3.9 and 9.3.10) and the requirements related to the mounting of the combs are found to be related through a second order relation. In the interviews, the engineers retrospectively agreed that the need for mounting brackets was difficult to foresee during ECN-03. This was supported by the lack of a first order relation between the combs and mounting bracket.

Moreover, the engineers agreed that the capability to expose these secondary relationships early in the ECN definition process could enhance the decision making process and, more specifically, augment the cost estimations. Through use of a modelling tool such as the higher order DSM, this change could have been properly addressed during the initial change.

5. Keyword based requirement change study

While the initial study demonstrated that a subject relationship between requirements could predict requirements that might be affected by the changes, the returned subset of potential requirements affected was approximately a quarter of all requirements. Thus, a refinement of the relationship types between requirements is explored in a second, larger scope case study. This project included 214 total requirements, approximately a third more than that of the subject based study. A list of ECNs were collected and used to identify associated, affected requirements. Similar to the subject based study, the ECNs simply stated "change in customer requirements" without specifically stating which requirement or set of requirements changed. It should be noted that after communicating the results of the initial study, the corporation has started to incorporate explicit identification of affected requirements in their new ECNs. To propagate the requirement change, a different type of relationship was developed in this study based on keywords. This provided a different perspective on the requirement relationships and the ability to propagate changes.

5.1. Selection of Keywords and Requirement Relationships

Keywords were selected by reviewing and interpreting the semantics of the requirements rather than simply the syntactical information. This was performed by studying the requirements document and understanding how each requirement specifically affected the system design. This required the authors review the requirements document three times before keywords could be elicited. After which the document was reviewed again and each requirement was tagged with five keywords which the author felt were relevant to the requirement and the overall system. While the selection of keywords is subjective, a set of common words were identified after investigating the requirements. However, some requirements consisted of keywords that resulted in minimal relationships. While out of scope of this paper, it is hypothesized that a controlled vocabulary may be developed for each project with a large intersecting portion of the vocabulary spanning multiple projects. To illustrate the keyword elicitation, consider the following requirement:

> *2.1.12: Vibration dampening level pads will be provided with a +/- 2-inch height adjustment capability.*

This requirements states all vibration dampening pads must be able to provide specific adjustability. The keywords driving this requirement are: *vibration, level pads, dampening, height, and adjustment*. These keywords were selected because any system which experienced vibration could require dampening pads and this requirement affects such systems. Further, level pads were selected as any changes which may occur to the pads themselves may affect this requirement. For example, level pads may be purchased which are not able to provide the targeted height adjustment. Dampening was selected as a keyword addressing the working principle of the level pads as there may be other dampening mechanism which relate to this due to their shared objectives. Height is selected as a keyword because of its overall dimensional affects on the system. Adjustment is selected as a keyword because it was important this system afford adjustability to satisfy the requirement.

Each requirement was tagged with five keywords. This number of keywords was arbitrarily selected as a relatively high number of keywords. A total 1070 keywords

(407 unique) were elicited from the requirements. Investigation of the keywords and their propagation sensitivity revealed that the minimum number of keywords needed to propagate to the appropriate requirements was three. This was performed to prohibit the saturation of relationships due to superfluous keyword relationships. As seen in Figure 7, in the number of keywords increased beyond three, there were minimal returns in terms of the number of relationships. A fourth keyword would increase the number of relationships by approximately 4% while a fifth keyword would increase 6% over that of three keywords.

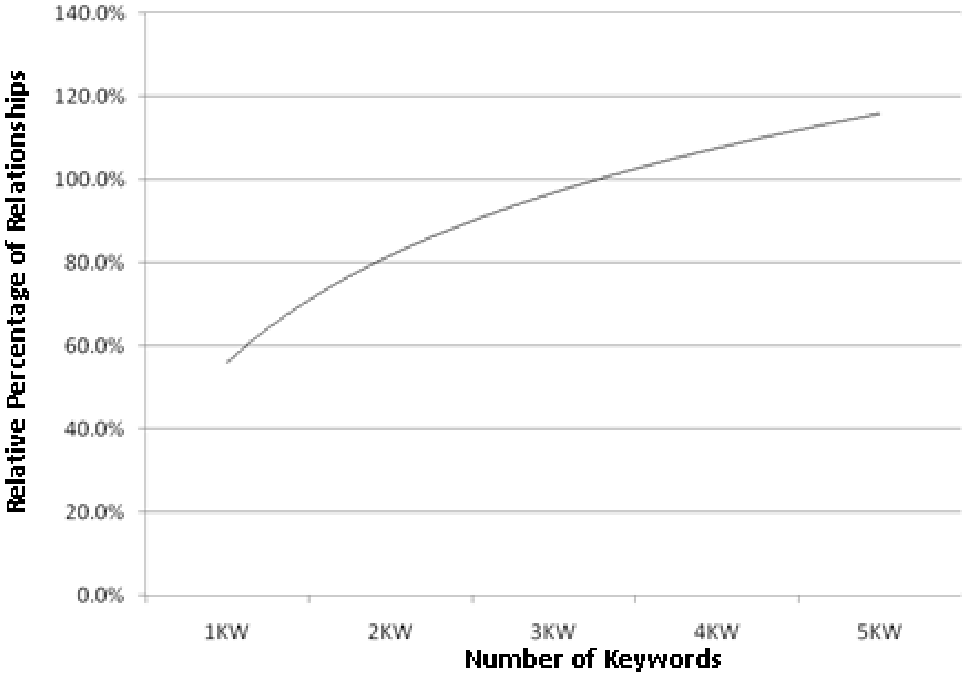


Figure 7: Percentage Number of Relations per Keyword w.r.t. Three Keywords

In developing relationships, requirement keywords were compared against the text of other requirements to identify if the text included those keywords. A requirement may only be related to another requirement if at least one its keywords exist within the text of the related requirement. For instance, requirement A is related to requirement B if any of the three keywords of requirement A were found in the text of requirement B. In this manner, the relationships are not bidirectional and the resulting DSM is asymmetric. For instance, the following requirements may have the keywords:

2.2.1: Tooling or fixtures switched during change over shall attach to a sub-plate in accordance with "single minute exchange die" (SMED) design philosophies.

Keywords: Tooling, Fixtures, Change over

2.9.14: Fragile Parts (Sensors, plastic parts, plastic gears etc..) or parts touching fragile parts (e.g. gear to gear assembly) must be assembled with tooling incorporating force control (and/or spring loaded mechanisms) to prevent part damage during the assembly.

Keywords: Fragile, Touching, Force Control

These keywords cause a relationship from requirement 2.2.1 to 2.9.14 because requirement 2.9.14 has the word "tooling" within its requirement text. However, this relationship is not bidirectional as none of the keywords belonging to requirement 2.9.14 are located within the text of requirement 2.2.1.

5.2. Development of DSM

The DSM (Figure 8) was developed based on keyword based requirement relationships, where requirements may be related if there is a keyword match. An extract from the complete DSM is illustrated in Figure 9. It should be noted that the strength of the relationship is not addressed here as the relationship is simply binary; existing or not. While it may be noted that multiple keyword matches would suggest a stronger relationship, this is reserved for future investigation. Unlike the subject based study, this DSM is asymmetric.

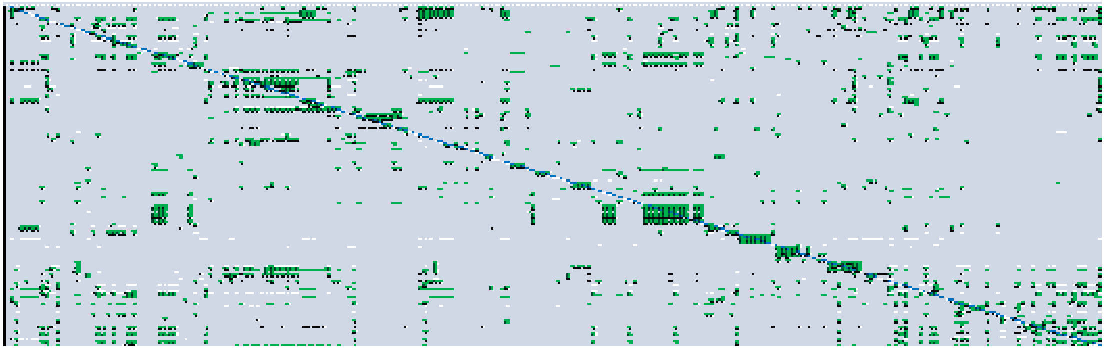
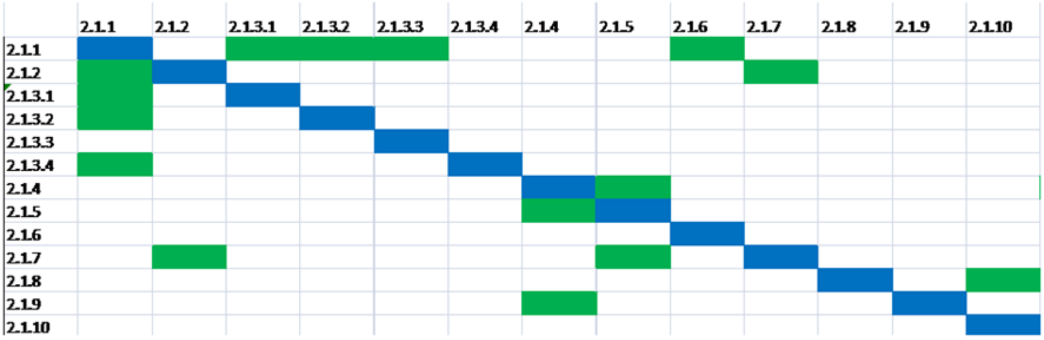


Figure 8: DSM for Keyword Based Study



	2.1.1	2.1.2	2.1.3.1	2.1.3.2	2.1.3.3	2.1.3.4	2.1.4	2.1.5	2.1.6	2.1.7	2.1.8	2.1.9	2.1.10
2.1.1	🟦		🟩	🟩	🟩				🟩				
2.1.2	🟩	🟦								🟩			
2.1.3.1	🟩		🟦										
2.1.3.2	🟩			🟦									
2.1.3.3					🟦								
2.1.3.4	🟩					🟦							
2.1.4							🟦	🟩					
2.1.5							🟩	🟦					
2.1.6									🟦				
2.1.7		🟩						🟩		🟦			
2.1.8											🟦		🟩
2.1.9							🟩					🟦	
2.1.10													🟦

Figure 9: Extract of DSM for Keyword Based Study

The DSM modelled 2,839 relationships between the 214 requirements. On average, each requirement had 13.3 relations with one requirement relating to 51 other requirements and other requirements being completely independent with zero relations.

5.3. ECNs for keyword based study

Each ECN requirement(s) was analyzed for its propagation on subsequent ECN requirements. For instance, ECN-01 was analyzed such that any propagation that may have resulted can be viewed and used to assist in propagating the subsequent requirements from ECN-07 and ECN-11. The ECNs were sorted based on their documentation date and their status. All the implemented ECNs, noted as "approved" were analyzed. A total of 16 ECNs are documented, of which six are approved. It is not documented why the remaining 10 were not approved and therefore they were not addressed in this study. However, it is important to note that the rejection was not due to change propagation. As before, directly changed requirements were extracted from each ECN. The date the change was initiated is also documented to track the initiation of the change. The reason for change in all cases was noted as a "change in customer requirements," however, once again, it is not explicitly stated which specific requirements were affected. A description of the change is detailed so this may be understood by those involved with the project. An expense is documented so the client understands the cost of the change before approval.

Of the six approved ECNs, only three ECNs were analyzed for the requirements they affected. This is due to the available information documented in the ECN to assist in identifying the changed requirement. While interviews may help expose the missing information, not all engineers associated with this project are still employed within the corporation. As such, these ECNs with insufficient documentation are removed from the study. Ultimately, ECN-01, ECN-07, and ECN-11 are studied to determine the predictability of requirement change based on the keyword built DSM. Identifying the affected, or changed, requirements was performed by viewing the initial set of requirements to identify which requirement dealt with the change noted in the ECN. For example, ECN-01 states "EGR attach station - Eliminated of 2 screw driver and torque arm". This requirement affected all requirements which deal with the EGR station. Further, any requirement pertaining to screw drivers and torque arms could also be considered as affected requirements. The affected requirements are noted and used in developing a higher order DSM. A list of the affected requirements for the three ECNs investigated is seen in Table 1. As seen in the table, some ECNs affected multiple requirements.

Table 1: Changed Requirements of Approved ECNs

Approved ECNs	**Date**	**Requirements Affected**
ECN-01	10-Jun-08	2.5.8 - 2.2.3 - 2.1.2 - 2.9.2 - 2.13.3 - 2.1.14
ECN-06	2-Sep-08	Insufficient Documentation
ECN-07	15-Aug-08	2.1.14 - 2.2.6
ECN-11	2-Sep-08	2.7
ECN-14	2-Sep-08	Insufficient Documentation
ECN-15	23-Jan-09	Insufficient Documentation

A higher order DSM, allowing the user to view the changes that propagate through the requirements based on keyword relations, was created for each of the requirements affected by an ECN. A total of eight higher order DSMs are developed to model the requirements affected from ECN-01 and ECN-07. A higher order DSM is not developed for the requirements affected in ECN-11 as it is the last ECN in the study and is not useful for predicting subsequent change. An example of one of the higher order DSMs populated is shown in Figure 10 and an extracted portion shown in Figure 11. This higher order DSM uses the requirement relationship in the original DSM to propagate the requirement changes. Using this change propagation, the requirements changed in subsequent ECNs will be analyzed to view if their propagation could have been predicted using this relationship and tool. All cells shaded in red indicate are first order propagation, while those shaded in yellow indicate a second order.

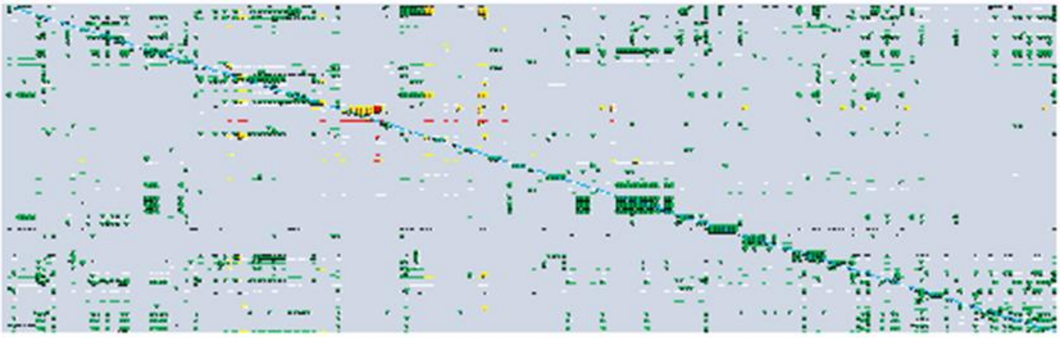
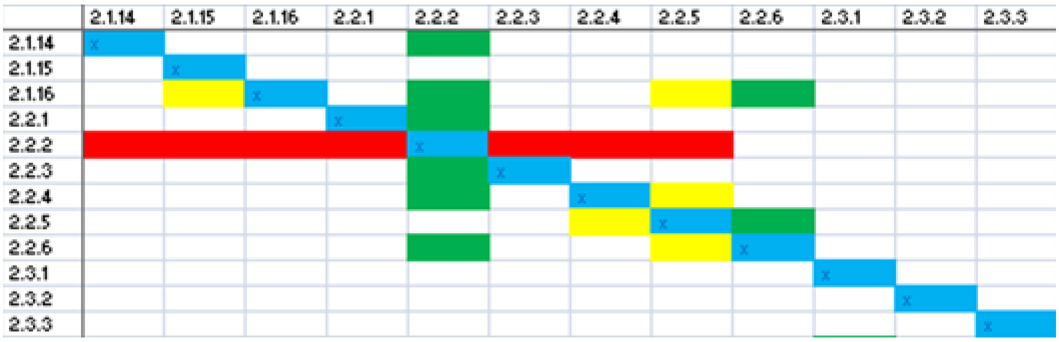


Figure 10: Example higher order DSM for Keyword Based Study



	2.1.14	2.1.15	2.1.16	2.2.1	2.2.2	2.2.3	2.2.4	2.2.5	2.2.6	2.3.1	2.3.2	2.3.3
2.1.14	🟦 x				🟩							
2.1.15		🟦 x										
2.1.16		🟨	🟦 x		🟩			🟨	🟩			
2.2.1				🟦 x	🟩							
2.2.2	🟥	🟥	🟥	🟥	🟦 x	🟥	🟥	🟥				
2.2.3					🟩	🟦 x						
2.2.4					🟩		🟦 x	🟨				
2.2.5							🟨	🟦 x	🟩			
2.2.6					🟩			🟨	🟦 x			
2.3.1										🟦 x		
2.3.2											🟦 x	
2.3.3												🟦 x

Figure 11: Extract of higher order DSM for Keyword Based Study

All three requirements, from both ECN-07 and ECN-11, could be predicted in this study as ECN-01 was the first change and this change could not have been predicted by a previous ECN. The ECNs analyzed were ECN-07 which comprised of changes to requirement 2.1.14 and 2.2.6 followed by ECN-11 which comprised of changes to requirement 2.7.

5.4. Requirement change propagation analysis

To highlight high potential requirements, those requirements possessing a great number of relationships, a ranking is defined for all relationships. This describes the relationship ranking compared to the remaining 213 requirements. This was noted as some requirements, due to their populous relationships, exaggerated the number of first and second order relationships it had with the requirements of interest. The ranking gives insight as to the strength of relationship, based on the number of relationships, compared to the other requirements.

As seen from the propagation results in

Table 2, many requirements shared a great number of relationships with the requirements affected by ECN-07 (requirement 2.1.14 and 2.2.6). The results indicated a relationship did exist as each of ECN-07's affected requirements could have been predicted through a previous ECN's affected requirement. The total number of relationships for each of the previous ECNs is shown in the first primary column of Table 2 and Table 3.

The first order relationship pathways are all first order path that a requirement has with related requirements. Since first order relations are a single pathway, each path is to an individual requirement. For instance, requirement 2.5.8 had 31 first order relations through 31 separate pathways. Second order pathways are those possible pathways of connection to second order relations. A requirement may be related to another requirement in the second order through multiple mediating requirements, increasing the number of second order relationship paths. For example, requirement 2.9.2 had 487 second order relationship pathways, of which 20 were to requirement 2.1.14, meaning there are twenty requirements which have a relation to both requirement 2.9.2 and 2.1.14. For this reason, a requirement may have more second order relations than there are number of total requirements.

5.4.1. ECN-07 – requirement 2.1.14

ECN-07's requirement 2.1.14 encompassed thirteen second order relationships with requirement 2.2.3, a requirement affected during ECN-01. Interestingly, ECN-07 encompassed a first order and twenty second order relationships with requirement 2.9.2. There exist no ranking for first order relations as this is binary and this relationship ranked 13th amongst all second order relationships for requirement 2.9.2. Examining ECN-07, it could be inferred the change of requirement 2.9.2 during ECN-01 influenced and propagated the change of requirement 2.1.14. Requirement 2.9.2 states:

> *Transport pallets shall be used to transport the product. Pallets will not be used as fixtures for critical operations. Client must approval deviations from this specification. If pallets are used as fixtures then each measurement must have a capability (Cpk) > 1.33. The measurement report has to be provided to client.*

It is important that this requirement was highlighted through the propagation as an immediate conflict is recognized. ECN-01 states "removed pallets and pallet return conveyor and replaced with fixed tooling nests." While the requirement states "pallets will not be used as fixtures," this change specifically states to place a fixed tooling nest on a pallet return conveyor while the requirement stated this should not occur. Nonetheless, reviewing the requirement affected in ECN-07, requirement 2.1.14 states:

> *The entire base plate where tooling and fixtures are mounted must be completely removable for each process in such a manner that a new base plate with new tooling can be interchanged.*

Again, it is seen that this requirement states that tooling and fixtures which are mounted must be completely removable. This is in direct conflict with ECN-01 as it called for the addition of a fixed tooling nest. This change was imminent as it directly conflicted with requirement 2.1.14. This was recognized by the higher order DSM as a critical requirement.

Table 2: ECN-07 Propagation Analysis

Total Relationship Pathways			**Relationships with ECN-07** ***Requirement 2.1.14***		
ECN-01	1st Order	2nd Order	1st Order	2nd Order	2nd Order Ranking
2.5.8	31	249	-	-	-
2.2.3	43	794	-	13	31
2.1.2	37	539	-	2	56
2.9.2	38	487	1	20	13
2.13.3	62	1163	-	3	93
2.1.14			N/A	N/A	-
Total Relationships Pathways			**Relationships with ECN-07** ***Requirement 2.2.6***		
ECN-01	1st Order	2nd Order	1st Order	2nd Order	2nd Order Ranking
2.5.8	31	249	-	-	-
2.2.3	43	794	-	15	32
2.1.2	37	539	1	29	6
2.9.2	38	487	1	40	3
2.13.3	62	1163	-	17	12
2.1.14	22	375	-	18	16

5.4.2. ECN-07 – requirement 2.2.6

The second ECN-07 affected requirement, requirement 2.2.6 states:

> *Supplier will design equipment for fast change over time (5 minutes or less total line change overtime) using quick change out tooling and fixtures.*

It could be inferred requirement 2.9.2 or 2.1.2 propagated to requirement 2.2.6 because of the number of first and second order relationships and their ranking. This requirement again relates to the design of tooling and fixtures, a set of components which changed during ECN-01. The second requirement from ECN-01 which shared high relation with requirement 2.2.6 was requirement 2.1.2. Requirement 2.1.2 states:

> *Tooling or fixtures switched during change over shall attach to a sub-plate in accordance with "single minute exchange die" (SMED) design philosophies*

Attaching a sub-plate to a fixture may have influenced the ability for the fast change over time stated in requirement 2.2.6. As ECN-01 influenced requirement 2.9.2

and 2.1.2, both of which address the use of fixtures and their attachment, the higher order DSM could have assisted in predicting an influence such change would have had on requirement 2.2.6 which relates to the change out of tooling and fixtures.

5.4.3. ECN-11 – requirement 2.7

Finally, ECN-11 is examined with the results illustrated in Table 3. Reviewing the final change, ECN-11 affected requirement 2.7 which pertains to station lights, stating:

> *Status lighting at every station must be mounted for good visibility (Top down: Red-Yellow-Green). Module Status Indicator lighting will be as follows:*
>
> - *Green Light (Solid): No Faults present on Module*
> - *Green (Flashing): Quality Check is switched off*
> - *Yellow (Solid): Manual Mode – Not in automatic*
> - *Yellow (Flashing): Parts Bin out of parts*
> - *Red and Yellow (Flashing): Station out of parts*
> - *Red (Solid): Station is faulted*
> - *Red (Flashing): Station Stopped - Part failed caused by % counter of faults*
> - *Green and Yellow and Red (Solid): Station is deactivated*
> - *Green (solid) Yellow (flashing): no faults, cycle time above limit*

In this ECN, requirement 2.7 was selected as the affected requirement because of its significance to the lighting over each station. This was important because ECN-11 states to make a change in which lights will be stacked at each station. This change was incorporated into the higher order DSM for analysis and it was found that the subsequent requirement change with the greatest influence was requirement 2.5.8, as seen in Table 3. Requirement 2.5.8 had a first order relation with requirement 2.7 and also had 28 second order relationships, more than any other requirement. Further, requirement 2.13.3 was also highly related to requirement 2.7 through second order relationships with 23 relationships. This is a potential indicator that the change to requirement 2.5.8 and 2.13.3 propagated to requirement 2.7. Requirements 2.5.8 and 2.13.3 respectively state:

> *Individual Module and/or system operations can be PLC controlled as long as the data transfer, collection, and management is PC based and does not slow down the speed of the system.*

> *If no part tracking (i.e. Pallet RF tag, etc) is used or if the last station (packaging) includes a process (e.g. screw driving) then process failures will require an operator intervention (e.g. a reset, password, or key switch) before the failed part can be removed from the station. The operator or equipment will remove non-conforming product at the point of failure to a lock box or equivalent quality device. The equipment must confirm the placement of the non-confirming material in the lock box before restarting. Furthermore the control has to count each failure in order to validate the number of bad parts in the box and the number of occurred failures. The lock box will have ergonomic access such as a chute or gravity conveyor. No part damage is allowed and has to be considered for the design of the conveyor (opportunity for potential rework). For End Of Line Testing stations only automated bad part handling is allowed.*

Initially examining those requirements directly, one does not immediately identify a relationship with the status lighting stations. Unlike the previous examples where the relationship was apparent, the strength of using a change modelling tool is realized in situations such as this. Further, it is difficult to consider this relationship as coincidental because of the high number of relationships and ranking of relationships between requirements 2.5.8 and 2.13.3 to their potential propagated requirement 2.7.

Table 3: ECN-11 Propagation Analysis

Total Relationships Pathways			**Relationships with ECN-11 Requirement 2.7**		
ECN-01	1st Order	2nd Order	1st Order	2nd Order	2nd Order Ranking
2.5.8	31	249	1	28	1
2.2.3	43	794	-	-	-
2.1.2	37	539	-	4	31
2.9.2	38	487	-	4	32
2.13.3	62	1163	1	23	7
2.1.14	22	375	-	-	-
ECN-07					
2.2.6	42	889	-	3	50

In subsequent review with the automation corporation, they recognize that there is perhaps a relation between these requirements, but readily admit that this relation is not apparent. For this reason, the corporation is considering adoption of the modelling strategy to help guide the engineers during the ECN proposal process so that accurate costs can be comprehensively predicted.

6. Discussion

Two studies were presented in this paper, both presenting a requirement change propagation tool exploring different types of requirement relationships. The first relationship technique made use of a syntactical subject based relationship. This resulted in a symmetric DSM and gave general insight as to the propagation of requirement and their use in predicting engineering change through requirement propagation. The first study indicated that change most often goes unnoticed during second order propagation. A subsequent study was performed to explore different types of requirement relationships, through keywords, that may affect the propagation of requirements and their ability to predict engineering change. The study was able to confirm that the propagation of requirements can be used to predict future EC requests. Using a requirements change modelling tool could have given the designers and engineers an indication of the potential propagation due to the initial requirement change. The use of a tool in this instance could have saved time lost between ECNs; prevented inaccurate estimation of change cost, and reduced the risk due to the uncertainty involved in approving changes.

It was found that second order relationships appear to be the key to predicting subsequent change requests to the requirements. Second order relationships are of great interest because most designers and engineers cannot predict the propagation of changes to such length. In interviews with the engineers, the first order subject based relationships were recognized as generally easy to predict. For example, if a change is made to the spindle on the machine, the machine spindle may affect the bearing holding the spindle. However, a second order relation arises when the bearing, in turn, has an effect on an adjacent snap ring. As a result, while the spindle is not directly related to the snap ring, there is a second order relationship there that may expose resulting snap ring changes derived from spindle changes. It is difficult for designers to intuitively predict changes in the second order form, especially for complex systems which may have hundreds or thousands of requirements. Rarely was unforeseen propagation occurring in first order form, rather it was occurring in second order, as seen in the results from both studies. This introduces an interesting dynamic to propagating requirements that cannot be recognized by simple designer attentiveness to change; rather the use of change modelling tools is needed.

The studies performed assisted in determining the important factors to consider during the propagation of requirements. Both projects were able to identify that propagation can be predicted through a requirements change modelling tool, in this study higher order DSMs are created. The limitations with using such a tool are the high number of potential propagated requirements and lack of distinction between the propagated requirements. Additionally, the differing studies assisted in determining the importance of relationship types. Relationship types have an impact on the DSM created, and as a result, any subsequent higher order DSMs developed.

In both studies, due to the relatively short list of requirements for a mechanical system, 160 and 214, a third order relationship was deemed inadequate. At the third order, nearly 95% of all the requirements are related, making it difficult to accurately predict changes.

A consideration when using such a system is the time and effort required to maintain the tool. An associate, usually the designer or engineer, at all times must maintain such a system to ensure all requirements, and their relationships are up to date. This requires time as requirements are continuously changing and new requirements are introduced thereby resulting in changes in relationships. However, the cost benefit of using and maintaining a requirements change modelling tool could prove to be financially sound as it may reduce losses resulting from mismanagement of changes. A specific example of this is seen in the initial study where the corporation lost nearly a month of time and thousands of dollars due to their inability to predict requirement change, based on interviews. The president of the corporation stated "this tool could have saved us a $100,000 because of unanticipated changes," indicating its industrial potential.

This tool is not able to predict how much a change may actually cost, however it leads designers and engineers to taking into considerations requirements otherwise neglected. While the higher order DSM does provide the designer insight as to the change which may occur due to propagation, all affected requirements may not necessarily change. A requirement may be affected by a change to a related requirement; however this does not always merit a subsequent requirement change. The key takeaways of this study are:

(3) A requirement change management tool can be used to propagate requirement change with a high degree of certainty.

(4) The importance of second order propagation within requirements is critical to the success of effectively predicting requirement change

(5) The recognized importance of requirement relationship types on the ability to propagate requirement change.

(6) The needed ability to weight, rank or narrow the list of propagated requirements into a list of high potential requirements the designer is able to use to assist in analyzing requirement change.

A contribution of this research is revealing how requirements are related through different mediums, in this study subjects and keywords are used. Though recognized as important, no comparison is provided between the two methods as it is out of scope of this paper. It is also important to note that the DSMs created here are intentionally not created by system experts who worked on the design. The aim of this tool is to be robust enough to operate with less experienced engineers and designers, making it usable by anyone. Research has been performed to view comparison between a requirement based DSM and an expert generated DSM and revealed those developed by experts have less relations between requirements, however additional relations do not adversely affect the approach [61]. It is recognized that the selection of keywords may is subjective and its repeatability and reproducibility must be analyzed.

7. Conclusions and future work

The need for a requirements modelling tool is critical in ensuring requirements are maintained and the consequences of requirement changes are properly assessed. This is apparent in the study presented in this paper as two studies were performed which explored the use of a requirements change modelling tool, specifically a higher order DSM, to predict engineering change propagation. It was determined from this study that second order change propagation are most likely to propagate and are difficult to foresee at the time of change. A modelling tool, such as that proposed in this paper, can provide the designer insight as to the requirements which may be affected before approving an engineering change.

In some instances, a requirement change could saturate nearly a quarter of all the remaining requirements through second order relationships, resulting in several false positive requirement change propagation scenarios. Managing dozens of potentially related requirements may not pose great difficulty, but scaling to hundreds or thousands of requirements quickly becomes too complex to manage without additional filters. A means of narrowing the high potential list of requirements is needed such as the rankings used in the keyword based study, weightings and impact factors. Further work is required in enhancing the tool to be capable of identifying the creation of new requirements, which currently cannot be performed. The addition of a requirement is considered here as a requirement change, as the requirement document does experience change in content.

Relationship weightings must be developed so not all requirements relationships are treated equally. Specifically, future work will be performed with both studies to view how weighing the changes based on their relationship will assist in narrowing the selection of requirements propagated. Weightings can be incorporated pre and post propagation. For instance, a pre-propagation weighting may include a weighting measuring the strength of relationship. A post propagation weighting may include weightings to measure the difference in propagation strength between first, second and third order propagation. A weighting could be applied to computationally calculate the possibility of occurrence through the relationship type, be it first or second order, and quantity of relationships. This will assist in determining the change propagation with a higher degree of certainty and decrease the time needed to evaluate change propagation. After differentiating between the requirements propagated, the next steps include ranking the propagated requirements to identify high risk requirements. Once weighting have been developed, the requirements highlighted as high potential will be intersected to view if the propagated requirements, which manifested into subsequent ECNs, could be identified. Different methods of relating the requirements to propagations will be implemented until a consistent method is found.

The projects investigated in both studies were completed and the authors retrospectively attempted to predict engineering propagation using the information provided. It is important this research is performed on live studies to ensure it is capable of assessing requirement change and the consequences involved, such as cost and time delay. Nonetheless, a historical analysis is needed to develop the framework for future studies on live projects where this work may be implemented

References

[1] G Pahl and W Beitz, *Engineering Design: A Systematic Approach*, 2nd ed.: Springer, 1998.

[2] Karl T Ulrich and Steven D Eppinger, *Product Design and Development*, 1st ed.: McGraw Hill, 1995.

[3] David G. Ullman, *The Mechanical Design Process*, 3rd ed.: McGraw-Hill, 2003.

[4] Kevin Otto and Kristin Wood, *Product Design*.: Prentice Hall, 2000.

[5] Elizabeth Hull, Ken Jackson, and Jeremy Dick, *Requirements Engineering*, 2nd ed.: British Library Cataloguing, 2005.

[6] Zhen Yu Chen, Shengji Yao, Jian Qiang Lin, Yong Zeng, and Armin Eberlein, "Formalisation of product requirements: from natural language descriptions to formal specifications," *International Journal of Manufacturing Research*, vol. 2, no. 3, pp. 362-387, 2007.

[7] W. Lam and V. Shankararaman, "Requirements Change: A Dissection of Management Issues," in *25th Euromicro Conference*, 1999, p. 2244.

[8] M. Lehman, "Software Future: Managing Evolution," *Software, IEEE*, vol. 15, no. 1, pp. 41-44, 1998.

[9] M R Cutkosky and J M Tenenbaum, "Toward a Computational Framework for

Concurrent Engineering," in *IECON Industrial Electronics Society*, 1990.

[10] P John Clarkson, Caroline Simons, and Claudia Eckert, "Predicting Change Propagation in Complex Design," in *20th IEEE International Conference on Software Maintenance*, vol. 126, Sept 2004.

[11] M. Lehman and L Belady, *Program evolution: processes of software change*.: Academic Press, 1985.

[12] G. Kotonya and I. Summerville, "Viewpoints for Requirement Definition," *Software Egnineering ournal*, vol. 7, no. 6, pp. 375-378, 1992.

[13] Hongxing Cheng, Yuanqiang Xia, and Xia Hu, "Requirements Change Management of Information System Based on Keyword Mapping," in *The Sixth Wuhan International Conference on E-Business*, pp. 135-140.

[14] S.J Andriole, *Manage Systems: Requirements, Methods, Tools and Cases*.: McGraw-Hill., 1996.

[15] S., Summers, J., Mocko, G Teegavarapu, "Case Study Methods for Design Research," in *ASME Design Engineering Technical Conference, DTM-4998*, Brooklyn, NY, 2008.

[16] R.K. Yin, *Case study research: Design and methods. Applied social research methods series*, 5th ed. London: : Sage Publications., 2003.

[17] R.E. Stake, "The case study method in social inquiry," *Educational researcher*, vol. 7, no. 2, pp. 5-8, 1978.

[18] Sergey Diev, "Requirements Development As a Modeling Activity," *ACM SIGSOFT Software Engineering Notes*, vol. 32, no. 2, March 2007.

[19] Ralph R Young, *Effective Requirements Practices*. Boston: Addison-Wesley, 2001.

[20] Andreas S Andreou, Andreas C Zographos, and George A Papadopoulos, "A

Three-Dimensional Requirements Elicitation and Management Decision-Making Scheme For The Development Of New Software Components".

[21] Clive L Dym, *Engineering Design a Synthesis of Views*.: Cambridge University Press, 1994.

[22] Tal Cohen and Robert E. Fulton, "A Data Approach to Tracking and Evaluating Engineering Changes," in *Design Engineering Technical Conferences*, Atlanta, 1998.

[23] K.M. Marchek M. Kannapan, "A Schema for Negotiation Between Intelligent Design Agents in Concurrent Engineering," *Intelligent Computer Aided Design*, 1992.

[24] N Nurmuliani, Didar Zowghi, and Susan P. Williams, "Requirements Volatility and Its Impact on Change Effort: Evidence-based Research in Software Development Projects," in *AWRE*, Adelaide, Australia, 2006.

[25] Atsushi Kobayashi and Mamaro Maekawa, "Needs Based Requirements Change Management," in *IEEE International Conference and Workshop on the Engineering of Computer Based Systems ECBS*, 2001.

[26] Saffena Ramzan and Naveed Jkram, "Making Decision in Requirement Change Management," in *First International Conference on Information and Communication Technologies* , 2005, pp. 309-312.

[27] Beshoy Morkos and Joshua D. Summers, "Requirement Change Propagation Prediction Approach: Results From an Industry Case Study," in *Proceedings of the ASME International Design Engineering Technical Conferences*, Montreal, Canada, 2010.

[28] Monica Giffin et al., "Change Propagation Analysis in Complex Technical

Systems," *Journal of Mechanical Design*, vol. 131, Aug 2009.

[29] Eddie Smith, "Re-Engineering a Trash/Recycling Collection Vehicle - Based on Challengine Customer Requirements," Clemson University, Masters Thesis 2010.

[30] Beshoy Morkos and Gary S Palmer, "A Study Of Designer Familiarity With Product And User During Requirement Elicitation," in *Eighth International Symposium on Tools and Methods of Competitive Engineering*, Ancona, Italy, 2010.

[31] Prabhu Shankar, Beshoy Morkos, and Joshua D. Summers, "A Hierarchical Modeling Scheme With Non Functional Requirements ," in *ASME Design Engineering Technical Conference*, Montreal, Canada, 2010.

[32] Suranjan Chakraborty, Saonee Sarker, and Joseph S Valacich, "Understanding Analyst Effectiveness In Requirements Elicitation: A Gestalt Fit Perspective".

[33] Gary S. Palmer, Beshoy Morkos, and Joshua D. Summers, "Investigation of Design Tools as Complexity Management Techniques," in *International Design Engineering Technical Conference*, Montreal, Canada, 2010.

[34] Zhen Yu Chen and Yong Zeng, "Classification of Product Requirements Based on Product Environment," *Concurrent Engineering: Research and Applications*, vol. 14, no. 3, pp. 219-230, September 2006.

[35] Beshoy Morkos, Shraddha Joshi, Joshua D. Summers, and Greg G. Mocko, "Requirements And Data Content Evalution Of Industry In-House Data Management System," in *ASME International Design Engineering Conference*, Montreal, Canada, 2010.

[36] G.A., Stahovich, T.F. Ollinger, "REDESIGNIT - A Constraint-based Tool for Managing Design Changes," in *International Design Engineering Technical*

Conference, Pittsburgh, 2001.

[37] G.Kotonya and I.Summerville, "Viewpoints for Requirement Definition," *Software Engineering Journal*, vol. 7, no. 6, pp. 375-387, 1992.

[38] R., Strens, M. Sugden, "Strategies, tactics and methods for andling change," in *IEEE Symposium and Workshop o n Engineering of Computer-Based Systems ECBS*, 1996.

[39] S., Eason, K., Dobson, J., Harker, "The change and evolution of requirements as a challenge to the practice of software engineering," in *IEEE international symposium on requirements engineering*, San Diego, California, 1993.

[40] C., Clarkson, P., Zanker, W., 2004 Eckert, "Change and Customization in Complex Engineering Domains," in *Research in Engineering Design*., 2004.

[41] I C Wright, "A review of research into engineering change management: implications for product design," *Design Studies*, vol. 18, no. 1, pp. 33-42, January 1997.

[42] P., Malmqvist, J. Pikosz, "A Comparitive Study of Engineering Change Management in Three Swedish Engineering Companies," in *Design Engineering Technical Conference*, 1998, pp. 78-85.

[43] S.R., Tomer, A Schach, "A Maintenance-oriented Approach to Software Construction," *Journal of Software Maintenance Research and Practice*, vol. 12, no. 1, pp. 25-45, 2000.

[44] V. Ravlich, "Modeling Software Evolution by Evolving Interoperation Graphs," *Annals of Software Engineering*, vol. 9, pp. 235-248, 2000.

[45] Prabhu Shankar, Beshoy Morkos, and Joshua D Summer, "Reasons for change propagation: A case study in an automotive OEM," *Research in Engineering*

Design, In Press 2010.

[46] Yuh-Min Chen, Wei-Shin Shir, and Chung-Yen Shen, "Distributed engineering change management for allied concurrent engineering," *International Journal of Computer Integrated Manufacturing*, vol. 15, no. 2, pp. 127-151, 2002.

[47] Elizabeth Hull, Ken Jackson, and Jeremy Dick, "DOORS: A Tool to Manage Requirements," in *Requirements Engineering*.: Springer, 2011, ch. 9, pp. 181-198.

[48] Tal Cohen and Robert E. Fulton, "A Data Approach to Tracking and Evaluating Engineering Changes," in *Design Engineering Technical Conferences*, Atlanta, 1998.

[49] Maria Carrascosa, Steven D. Eppinger, and Daniel E. Whitney, "Using The Design Structure Matrix to Estimate Product Development Time," in *ASME Design Engineering Technical Conference*, Atlanta, GA, 1998, pp. 13–16.

[50] A. A. Yassine, D. E. Whitney, and T. Zambito, "Assessment of rework probabilities for simulating product development processes using the design structure matrix (DSM)," *Engineering Conference*, no. 617, pp. 1-9, 2001.

[51] Josh King, "Design structure matrix-based Engineering Change management for product development," *International Journal of Internet Manufacturing and Services*, vol. 1, no. 3, p. 231, 2008.

[52] P. T. Helo, "Product configuration analysis with design structure matrix," *Industrial Management Data Systems*, vol. 106, no. 7, pp. 997-1011, 2006.

[53] D.V. Steward, "The Design Structure System: A Method for Managing the Design of Complex Systems," in *IEEE Transactions on Engineering Management*, 1981, pp. 71-74.

[54] T.R. Browning, "Applying the Design Structure Matrix to System Decomposition

and Integration Problems: A Review and New Directions," in *IEEE Transactions on Engineering Management*, 2001, pp. 292-306.

[55] John N. Warfield, "Binary Matrices in System Modeling," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 3, no. 5, pp. 441-449, September 1973.

[56] S., Whitney, D., smith, R.P., Gebala, D.A. Eppinger, "A Model Based Method for Organizing Tasks in Product Development," in *Research in Engineering Design*., 1994.

[57] R. Keller, T. Eger, C. M. Eckert, and P. J. Clarkson, "Matrices Or Node-Link Diagrams: Which Visual Representation Is Better For Visualising Connectivity Models," *Information Visualization*, vol. 5, pp. 62-76, 2006.

[58] E., et al. Fricke, "Coping with changes: Causes, findings , and strategies," *Systems Engineering*, vol. 3, no. 4, pp. 169-179, 2000.

[59] C. Lamar and G. M. Mocko, "Linguistic Analysis of Natural Language Engineering Requirements Statements," in *Tools and Methods for Competitive Engineering*, Ancona, Italy, 2010.

[60] Carl Lamar, "Linguistic Analysis of Natural Language Requirements," Clemson University, Clemson, South Carolina, Master's Thesis 2010.

[61] Qi Dong, "Predicting and Managing System Interaction at Early Phase of Product Development Process," Massachusetts Institute of Technology, Cambridge, MA, Ph.D. Dissertation 2002.