ERMM: An Engineering Requirements Management Method

Nanxin Wang
Vehicle Design Research & Advanced Engineering Department,
Ford Research & Advanced Engineering,
2101 Village Road,
P.O. Box 2053,
MD3135, SRL
Dearborn, MI 48121-2053, USA
e-mail: nwang1@ford.com

Automotive product development is a lengthy and complex process. There exists a large body of various requirements, standards, and regulations, which need to be followed by all engineering activities throughout the entire vehicle development process. The underlying relationships between these requirements are very complicated. Although most of engineering requirements can be found in various engineering databases, it is the lack of the underlying relationship between the requirements and their association with the design that makes it extremely difficult for even experienced engineers to follow the requirements in their day-to-day work. This paper introduces an engineering requirements management method that captures these interrelationships and associations using a matrix-based representation. A case study with a real automotive component is also presented.
[DOI: 10.1115/1.2202869]

Introduction

Automotive product development is a lengthy and complex process. After more than 100 years in production and commercial use, there exists a large body of various requirements, standards, and regulations that needs to be followed by all engineering activities throughout the entire vehicle development process. These requirements differ in type, source, and application. They are dynamic in nature, i.e., under constant review and revision.

Each requirement interacts differently with the various customer needs, the product systems, components and component features, and the processes used to design and manufacture the products. The interaction between the customer's needs, engineering requirements, product components/parameters, and processes can be viewed as a web (Fig. 1). This leads to a highly complex engineering environment.

On the other hand, the engineering community that uses and enforces these requirements is quite complicated in nature as well. In large engineering enterprises, thousands of engineers may be employed in the product development process. They can be grouped into three main categories, according to their job functions and responsibilities: (1) product design engineers, (2) attribute design engineers, and (3) process design engineers

The product design engineers are responsible for the overall system initial design and/or redesign and system level integration, e.g., a bumper system or a closure system. The attribute design engineers have responsibilities for system attributes rather than the physical embodiment, e.g., safety, NVH, durability, etc. The process design engineers have the overall responsibilities for developing design or manufacturing processes. Process development is used here to mean either the development of an analytical or physical test method or a manufacturing process such as an assembly or component fabrication process. The process engineers provide input to the product design engineers. There are some overlapping responsibilities between these three groups of engineers. All of them are responsible for developing and supporting product and manufacturing regulations or standards.

While performing their required job assignments, product design engineers will usually encounter the following question: "When making a design change, how do I know what engineering requirements need to be rechecked?" Conversely, attribute and process engineers will ask themselves: "When an attribute or manufacturing process requirement is changed, what components are affected?" In today's practice, these questions are usually answered based on individual engineers' knowledge and experience.

Although most of the engineering requirements can be found in various engineering databases, it is the lack of the underlying relationship between the requirements and their association with the design that makes it extremely difficult for even experienced engineers to follow the requirements in their day-to-day work. What is needed is a systematic way to organize the requirements and track the relationships and dependencies among various requirements and design.

This paper introduces an engineering requirements management method (ERMM) that captures the interrelationship and associations between engineering requirements and design, using a matrix based representation. The next sections start with related research, then a detailed description of ERMM, a case study, and the conclusion.

Related Research

Requirements management methods have been developed and deployed extensively in the software engineering domain and often referred to as software requirement engineering [1]. Software engineering requirements management methods are used to map specifications to functions and constraints on software development. An overview of the field is presented by Nuseibeh [2]. There are many requirement management tools existing in the requirement engineering domain [3–9]. When used in mechanical engineering domain, requirement management often refers to managing requirements for developing a project or defining the functions of a product [10]. In this paper, the authors are focusing on managing the relationships between the existing or known engineering requirements and the products/processes that will follow or obey these requirements.

Matrix-based analyses have been used to represent complex relationships between the components of a system in a compact, visual, and analytically advantageous format. There exist several different matrix-based design methods. Axiomatic design (AD) is defined as the development and selection of design parameters (DP) to satisfy objectives' functional requirements (FR), subject to design constraints [11]. Design problems may be represented as three mappings between four domains: (1) customer to functional requirements, (2) functional requirements to design parameters, and (3) design Parameters to process. matrices are used to capture the mappings between the domains. The functional requirements to design parameters mapping is known as the *axiomatic design matrix* (*ADM*) and has been widely applied.

The design structure matrix (DSM) has been used to represent complex relationships between the components of a system. A good tutorial of the DSM methodology can be found at the MIT website [12]. The DSM matrix captures the interactions, interdependencies, and interfaces between system elements of the same domain. DSM's domains of interest are: (1) design components, (2) design parameters, (3) process, and (4) organization. An optimum system can be achieved by reducing the feedback to a minimum through rearranging the DSM elements. This results in a better system/subsystem grouping in the case of a design component/parameter matrix. In the case of a process DSM matrix, it results in an improved and more realistic execution schedule for the corresponding activities.

Contributed by the Engineering Informatics (EIX) Committee of ASME for publication in the JOURNAL OF COMPUTING AND INFORMATION SCIENCE IN ENGINEERING. Manuscript received January 13, 2005; final manuscript received April 12, 2006. Review conducted by R. Sriram, S. Szykman, D. Durham.

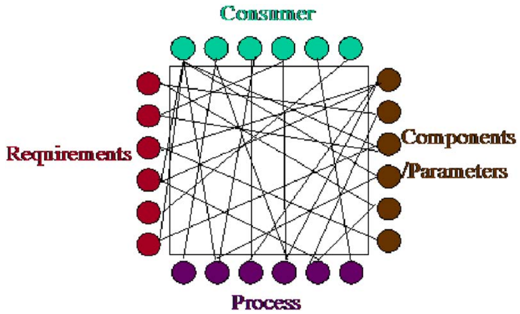

Fig. 1 Engineering requirements management complexity

The AD and DSM methods have been extended and combined in a number of ways as presented by Hintersteiner [13], Dong [14], Melvin [15], Eppinger [16], and Brady [17]. Hintersteiner proposed a standard classification of axiomatic design's functional requirements. Melvin uses the DSM methodology to reorder the leaf-level design parameters in an axiomatic design matrix in order to obtain a noniterative design process. Eppinger describes a method based on three DSM matrices and three mixed-type matrix based mappings. Brady makes use of two extensions to the DSM methodology. He added an interface dependence measure to the DSM component interaction. This adds extra information to the matrix, which contain not only the iteration but also the type of interaction. The other extension is the matrix mapping of the functional phase to the component phase, similar to that of a ADM mapping with more limited scope.

The AD method appears to have greatest utility in supporting new mechanical designs, such as complex automotive designs. DSM is geared toward optimization of systems or processes. The problem ERMM is trying to solve is how to best manage the large body of existing engineering requirements in mechanical engineering domain. The ERMM solution builds upon aspects of both the AD and DSM methods to represent the complex relationships between the engineering requirements and the product design.

Engineering Requirements Management Method (ERMM)

ERMM is represented as a matrix of matrices (Fig. 2). ERMM is designed to capture the existence and nature of the interactions between and among the four domains of interest: (1) customer requirements, (2) engineering requirements, (3) components/parameters, and (4) processes/methods. The matrices on the diagonal capture the relationships within the same domain. The matrices below the diagonal represent the driving or input relationship from one domain to another and the matrices above the diagonal store the feedback information from one domain to another.

The customer requirements are those qualities that buyers of the product wish to purchase. Examples of customer needs for an automobile are the required seating capacity or a CD player. Customer needs reflect consumer's wants and are not the means for achieving them. They can be thought of as high-level requirements that a product should meet to satisfy the customer. The customer needs drive the engineering requirements and subsequently affect the product design, and the processes that produce the product. For example, the customer's need for fuel efficiency drives the demand for a more efficient powertrain and a lighter vehicle structure.

The phrase "engineering requirements" is used generically to include government regulations, corporate standards, and attribute requirements, such as weight requirements as well as manufacturing requirements. These engineering requirements directly affect the product and the methods used for the design and the manufacturing processes. Whereas the AD method focuses on functional requirements of a product, the ERMM engineering requirements domain is much more inclusive. An engineering requirement can be as complicated as a documentation that contains detailed information on how a component should be tested for its function or as simple as a value to specify a limit or a constraint. For example, a minimum flange length is a requirement for welding steel sheet metal components together.

The component/parameters domain refers to the different aspects of the physical embodiment of a product. The terminology "component" used in this paper include systems, subsystems, and components or parts. Design parameters define the shape and characteristics of a product. Design parameters can be used to define the high-level vehicle dimensions such as vehicle length, wheelbase, and windshield angle. They can also be used to describe component-level dimensions such as flange length or radius of a filet. Design parameters also include attributes such as torsion/bending stiffness. The components/parameters have been grouped together in a hierarchical manner. The entire finished product is considered the top-level system. The product is then broken down into systems, subsystems, and components. Design parameters can be associated with the systems, subsystems, or components. A component or parameter may be associated with one or more systems.

The ERMM process/methods domain consists of both the manufacturing processes and the design methods. The manufacturing process information consists of the system and component fabrication processes and is represented as a series of process steps. The design methods include both analytical and physical test methods.

Each of the ERMM domains contains a list of members which are referred to as the domain elements. The matrix contains entries for every domain element grouped according to its domain. Matrices are used to represent the mappings or interactions between and among each of the domain elements. The domains and domain elements located along the vertical axis are identical to those located along the horizontal axis as in the DSM method. Each cell in a matrix represents an interaction between a pair of domain elements, and will be marked if an interaction exists. The element along the vertical axis is the "driver" of an interaction. Its corresponding element along the horizontal axis is the "driven" elements of the interaction. The ERMM interactions therefore represent constraints imposed by the domain elements on the vertical axis upon the elements located along the horizontal axis.

In order to determine the effect of one domain element upon the rest of the design, one would first locate that element among the "drivers" along the vertical axis. All of the interactions in which that design element is the driver will be located in that column. The cascade effect can be found by following the interactions to see if the "driven" elements are in turn the "drivers" of other interactions. This cascade represents the constraint propagation of the driving element.

The ERMM matrix can be decomposed into 16 submatrices. Taken as a whole, the 16 ERMM submatrices can completely and comprehensively capture all of the constraints contained in a mechanical product design. The customer requirements drive the

	Customer Requirement	Engineering Requirement	Components/ Parameters	Process / Methods
Customer Requirement		ADM		
Engineering Requirement	ADM		ADM	
Components/ Parameters		ADM	DSM	
Process / Methods				DSM

Fig. 2 ERMM matrix of matrices

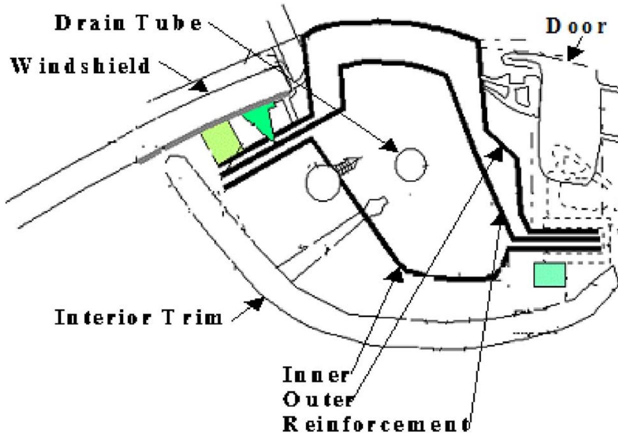

Fig. 3 An A-pillar cross section

need for the engineering requirements, the product systems, and to a lesser extent, the manufacturing processes. The customer requirements are, in turn, constrained by the other domains. In a like manner, the engineering requirements drive the system design and validation methods while the design components may require or drive engineering requirements. The processes and methods drive the requirements and constrain the design systems and components.

Case Study: Engineering Requirements for Body Structure Development

ERMM has been applied to capture and describe the interrelationships of the engineering requirements for an automotive A-pillar design. A simplified version of the ERMM matrix generated from this application is presented here to illustrate the method. A vehicle A-pillar spans the area between the front windshield glass and the side door assembly. A typical cross section of an A-pillar assembly is shown in Fig. 3. The basic functions of an A-pillar are to accommodate the vehicle styling, provide the space for the driver and passengers, and protect the occupants together with the other structural parts.

The customer requirements that affect A-pillar design included in this example, shown in the ERRM matrix in Fig. 4, are better sound quality, visibility, fuel efficiency, and keep rain and water out of the vehicle. These customer needs contained no interaction among themselves. The empty customer needs-to-customer needs mapping shown in the ERMM matrix reflects this lack of interaction

One of the main structural requirements for A-pillar design is NVH (noise, vibration, and harshness) torsion and bending stiffness. Another requirement is the A-pillar binocular obstruction to ensure proper visibility to the driver. The corporate average fuel efficiency (CAFÉ) is a typical government regulation that could affect the weight of an A-pillar. With respect to the A-pillar design, water management requirements pertain to door sealing systems and the drainage tubes located in the A-pillars that direct the runoff of rainwater from the roof. The welding requirements specify weld spacing, flange length, material, and thickness.

The requirements-to-requirements matrix reveals two coupled interactions between (1) NVH torsion/bending and CAFÉ weight class, and (2) NVH torsion/bending and welding. The NVH torsion and bending characteristics are influenced by the vehicle weight and weld pattern, and in turn the desired vehicle weight and weld pattern are influenced by the desired NVH characteristics.

The customer needs drive the engineering requirements. For example, customer needs for better visibility create the demand for less visual obstruction caused by the A-pillar. The engineering requirement to customer requirement matrix stores the feedback

1	1	2	3	4	5	6	7	8	9	10	##	12	13	14	15	16	17	18	19	20	21	22	23	24	25
2	Customer Needs																								
3	Better sound quality							x						x	x	x	x	x							
4	Visibility								x					x		x		x	x						
5	Fuel Efficiency									x				x	x	x	x	x							
6	Keep rain/water out										x									x					x
7	Engineering Req.																								
8	NVH torsion/bending		x							x		x		x	x	x	x	x							
9	A-pillar binocular			x										x		x		x	x						
10	CAFÉ weight class				x			x						x	x	x	x	x							
11	Water Management					x														x					x
12	Welding							x							x	x	x								x
13	Components/Param.																								
14	windshield angle		x	x	x			x	x	x															
15	A-pillar inner		x		x			x		x		x		x		x	x	x	x	x					x
16	A-pillar outer		x	x	x			x	x	x		x		x	x		x		x	x					x
17	A-pillar reinforcement		x		x			x		x		x		x	x	x			x	x					x
18	A-pillar interior trim		x	x	x			x	x	x				x	x				x						x
19	ceramic paint width			x					x						x	x	x	x							
20	Drainage tube diameter		x		x	x					x				x	x	x								
21	Processes/Methods																								
22	A-pillar Binocular								x					x		x		x	x						
23	NVH analysis							x						x	x	x	x	x						x	
24	weight analysis							x		x				x	x	x	x	x					x		
25	Assembly Sequence					x					x	x			x	x	x	x							

Fig. 4 A-pillar ERMM matrices

relationships from engineering requirements to customer needs. These relationships will inform the engineering community about what customer needs would be affected if an engineering requirement were changed. For example, changes on NVH torsion and bending requirement will change the sound quality that the customer would experience in a vehicle.

For this case study, we included four A-pillar components: inner, outer, reinforcement and trim, one vehicle level parameter (the windshield angle), and two A-pillar subassembly level parameters: the paint width and the drain tube diameter. The components-to-components matrix reveals a great amount of interaction. This is to be expected as this matrix reveals the physical constraints between components and systems and the relationships between components and parameters. For example, the windshield angle drives the location of the A-pillar. However, A-pillar trim will not affect the windshield angle.

The elements listed in the ERMM process/methods domain include three design analysis methods: (1) A-pillar binocular obstruction, (2) NVH analysis, and (3) weight analysis, and one manufacturing process: the assembly sequence. This matrix shows a coupled interaction between only two of the analysis processes, the NVH and weight analysis. The NVH torsion bending analysis takes consideration of the weight and weight distribution. In turn the weight may need to be rechecked after changes made to the structure as a result of the NVH analysis.

There is a lot of interaction between the customer needs and the components and very little interaction between the customer needs and the processes, with the exception of the Assembly sequence process. This is due to the fact that the assembly sequence process represents a high investment cost and imposes a constraint upon the customer needs. Also there is little interaction between the customer and the analytical methods used to design the product since the customer is interested only in the end result.

The components-to-requirements matrix tells engineers when a change is made to the components/parameters, and what engineering requirements need to be rechecked. For example, the NVH torsion requirement needs to be verified after changes are made to the windshield angle and/or other A-pillar components. Similarly, the process to requirements matrix informs engineers on what engineering requirements need to be followed when changing the methods or processes. In our case study, a change to the assembly sequence will prompt a recheck or even redefine the welding requirements. The component to processes/methods mapping informs the engineers when making design changes what engineering methods are used to evaluate the performance to the requirements and what manufacturing processes would be affected. In another example, a change to the assembly sequence process due to technology change may require redesign of the body structure construction, in this case, the A-pillar design.

With the information captured and classified in ERMM, engineers can now easily find the answers to their questions. For example, the product design engineers' question: "When making a design change, how do I know what engineering requirements need to be rechecked?" can be answered using the information stored in the requirement to component matrix. The other design elements affected by the design change can be found along the vertical column headed by the design component in question. The cascade effect of the design change can also be determined in a like manner.

Discussion

The ERMM has shown the capacity to represent the complex and inter-related design constraints of a mechanical design system. The matrix of matrices employed by the method is a conceptually easy to understand representation. However, in practice the matrix for even a simple product design becomes very large and hard to navigate. To resolve this scalability issue, a database approach has been taken to facilitate the ERMM and will be the subject of a future paper.

Conclusion

A matrix-based engineering requirements management method has been developed to capture the interactions between the customer needs, engineering requirements, components and design and manufacturing processes/methods. It has been tested using a vehicle structure component design. The result demonstrated that ERMM can effectively capture and establish the relationships and association between the customer's needs, the engineering requirements, the components/parameters, and the process/methods. It has the potential to bridge the gap between the requirements and engineering experience. Further tests are needed using more complicated systems, for example, the entire vehicle body structure. ERMM also lays the foundation for further research on the engineering requirements management process. One of the future works would be how to best manage the information in the ERMM.

References

[1] Thayer, R. H., and Dorfman, M., *Software Requirements Engineering*, 2nd ed., IEE, ISBN 0-8186-7738-4.
[2] Nuseibeh, B., and Easterbrook, S., 2000, "Requirements Engineering: A Roadmap," ICSE, Future of SE Track.
[3] Rational Suite, Analyst Studio – http://www.rational.com/products/astudio/index.jsp
[4] RDT 3.0 (Requirements Design and Traceability) – http://www.igatech.com/rdt
[5] RTM Workshop 5.0 – http://www.chipware.com/sub1.php3?section=rtm&sub=RTM_Overview
[6] Telelogic, DOORS – http://www.telelogic.com/products/doorsers/index.cfm
[7] Omni Vista, On Your Mark Pro – http://www.omni-vista.com/Products/OnYourMarkPro/index.htm
[8] Starbase Caliber – RM – http://www.borland.com
[9] Wiegers, K. E., Automating Requirements Management – http://www.processimpact.co
[10] Seybold, C., Meier, S., and Glinz, M., 2006, "Scenario-Driven Modeling and Validation of Requirements Models," *Fifth ICSE International Workshop on Scenarios and State Machines: Models, Algorithms and Tools*, Shanghai.
[11] Suh, N. P., 1990 *The Principles of Design*, Oxford University Press, NY.
[12] DSM website, http://web.mit.edu/dsm/
[13] Hintersteiner, J. D., 1999, "A Fractal Representation for Systems," *Proceedings of the 1999, International CIRP Design Seminar*, Enschede, The Netherlands.
[14] Dong, Q., and Whitney, D., 2001, "Designing a Requirement Driven Product Development Process," DETC2001/ DTM-21682, Pittsburg.
[15] Melvin, J., and Suh, N., 2002, "Beyond the Hierarchy: System-Wide Rearrangement as a Tool to Eliminate Iteration," ICAD2002-037, Cambridge, MA.
[16] Eppinger, S. D., and Salminem, V., 2001, "Patterns of Product Development Interactions," ICED '01, Glasgow.
[17] Brady, T. K., 2002, "Utilization of Dependency Structure Matrix Analysis to Assess Complex Project Designs," DETC2002/DTM-34021, Montreal.