SAE World Congress, Detroit 2007

Knowledge Management and Axiomatic Design

Christopher A. Brown, Mechanical Engineering Department, Worcester Polytechnic Institute, Worcester, Massachusetts, USA brown@wpi.edu

Abstract

The objective of this paper is to explain how axiomatic design can be used both to design a knowledge management system and to formulate the knowledge in the system. Knowledge is an important part of the capital of any company. The capacity to extract value from knowledge depends on the abilities to locate the essential knowledge, have it formulated in an appropriate manner, and apply it at the right time to find the best solution to a problem. Much of what has been written about knowledge management principally stresses the importance of management and the value in utilizing the knowledge effectively. However, in the existing publications, it is frequently not clear exactly how the knowledge system should be designed or how to formulate that knowledge within the system in order to extract optimal value. In this work, it is postulated that the value- adding intellectual activity in an enterprise can be formulated as an engineering design problem, using axiomatic design. Axiomatic design formulates a decomposition structure that includes four domains: customer, functional, physical, and process. Knowledge exists within the entities in domains and in the relation between the entities in adjacent domains. Once an entity has been identified in one domain, a properly designed knowledge management system can suggest all the known solutions in the adjacent domains, and the interactions between the domains, along with details about how well these solutions work.

1. Introduction

The objective of this work is to develop elements of the optimal design of the high levels of a knowledge management system using axiomatic design (Suh 1990, 2001). This is important because a good knowledge management system can add value to a company. The design of the knowledge management system is critical for realizing the maximum value of the knowledge system. The design of the system largely determines the ability of designers and planners to extract value from knowledge, which, in turn, depends on certain functions, including the ability of the system to allow users to create, upload, store, and then locate and download the essential knowledge in a timely manner. Axiomatic design is an engineering design method that can be used to develop a system that optimally fulfills these functions.

Much of what has been written about knowledge management emphasizes the importance of capturing knowledge and the value in using it effectively. It is not clear in the literature how the knowledge system should be designed or how best to formulate the knowledge in the system in order to extract value from it (Morgan and Liker 2006, Nonaka 1998, Leonard and Straus 1998, Kleiner and Roth 1998, Brown 1998, Kennedy 2005). It is this certainty of the desirability of a knowledge management system, yet uncertainty in the design of the knowledge management system that drives the need for the current work. The customer need (CN, an element of the customer domain) for knowledge management is well established in discussions in the literature. The axiomatic design process begins with this CN. Axiomatic design has been applied to a wide range of engineering designs, reducing design times, improving communications, and fostering innovation (Suh 1990, Suh 2001). It is based on the proposition that all good designs comply with two axioms: maximum independence and minimum information. Together these two axioms maximize the probability of the design to fulfill its purpose, and thereby achieve the optimal design for a set of functional requirements. The application of these axioms drives design descriptions to have a certain structure and the design process to follow a certain sequence as well (Brown 2005).

The approach in this work is to use axiomatic design to design a knowledge management system. The strategy is based on the postulation that the value-adding intellectual activity in an enterprise can be formulated as the process of finding the solution to an engineering design problem. Axiomatic design is a process that finds optimal solutions to design problems. Based on this postulate, the design structure used in the axiomatic design process is also used as the knowledge structure for the information stored in the knowledge management system.

2. Method

Axiomatic design consists of three elements: the axioms, the structure, and the process (Brown 2005). There are two axioms, or laws, that have been said to govern all good designs (Suh 1990). The first, the independence axiom, states that the independence of the functions should be maximized, and the second, the information axiom, states that, among those designs with maximized independence, the one with the least information will be the best.

A certain structure is required in order to apply the axioms. This structure involves decompositions in two directions, laterally into four domains: customer, functional, physical and process, and vertically in each domain into hierarchies from general to specific. In designing the knowledge management system, this work primarily considers the application of the independence axiom to the high levels of the functional and physical domains. The customer need for managing knowledge has been established and discussed in the literature.

The structure of the knowledge within a knowledge management system should include all four domains. The complete axiomatic design process includes a top-down decomposition process followed by a bottom-up integration process. This work only includes the decomposition. The physical integration is thought to be company- or situation-specific enough so that it could be the topic for a separate paper.

The decomposition of a design in the functional and physical domains is such that each element in the functional domain maps onto an element of the physical domain. The elements of functional domain are called functional requirements (FRs), and they describe the design intent. The elements of the physical domain are called design parameters (DPs), and they describes how to accomplish the intent, i.e., how to fulfill the corresponding FRs.

Activities within the company are required to add value to products before the products cross the company boundary in exchange for money. These activities can be defined as the utilization of resources -- energy, materials, and knowledge -- to add value to the product; essentially, this is production, i.e., design and manufacture. The two objectives in production are to maximize value added and to minimize cost.

There are also activities, internal to the company, which enhance the ability to add value and reduce time and cost during design and production, without directly adding value to a specific product. These activities add value to the company and include organizational design and knowledge management.

Not all knowledge has the potential to add value to a company or its products. This work focuses on value-adding knowledge. Moreover, not all knowledge is worth capturing and managing. If there is no potential for knowledge to add value, then there is no value in capturing and managing this knowledge. Given that current perceptions of what knowledge actually has the potential to add value could be wrong, then there may be some value in erring on the side of storing knowledge with no immediate perceived potential value. The fact that there is little cost to storing and searching information strengthens this argument. The cost would be in structuring the information so that it is useful and immediately applicable.

Given this context of value-adding activities and a specific design method, the procedure is to develop upper-level FRs based on perceived customer needs. Assessing the customer needs and formulating the FRs is a limiting step in design, and the final design resulting from the axiomatic design process cannot be better than the FRs. The customer need that this work addresses is to develop a system for managing knowledge in order to optimize the utilization of knowledge in adding value to the company and its products. The formulation of the FRs and DPs and their interactions are considered results of the design process and appear in the next section.

3. Results

The proposed decomposition of the upper level, level one, of the knowledge management system is shown in Table 1, which includes the high-level FRs and DPs. The intent is that these four FRs are the best decomposition of “manage knowledge.” The FRs are imperatives describing the functions. The decomposition of the FRs should be collectively exhaustive, i.e., the four functions together compose the required functions of a knowledge-management system at level one. They should also be mutually exclusive, i.e., these functions do not overlap – they are distinct. In addition, these FRs should be the minimum number that describes the functions required for managing knowledge. In this list, assessment and continuous improvement of the overall system, which are important elements of any management system, for now, can be considered as a separate design problem. The assessment of the knowledge itself is handled by DP2. The DPs are straightforward transformations of the FRs. The formulation of the DPs may seem trivial, as no content of the systems is apparent; however, this serves to define that at level one there are four separate systems of hardware, software, and people in organizational structures that need to be developed to satisfy the corresponding functions. Each of these level-one systems has its own set of FRs and DPs, some of which are explored below.

Table 1. Upper level of the knowledge management system

FR1: acquire knowledge	DP1: system for acquiring knowledge
FR2: assess knowledge	DP2: system for assessing knowledge

FR3: transfer knowledge	DP3: system for transferring knowledge
FR4: apply knowledge	DP4: system for applying knowledge

The assessment of coupling for compliance with axiom one at level one is shown in Table 2. The 0s indicate lack of interactions. The Xs indicate natural coupling for the indicated DP-FR interactions, and the Ss indicate a special kind of coupling called sequential (Brown 2006). All of the natural coupling is intended, forming a diagonal matrix, with each DP influencing only its corresponding FR. The sequential coupling is indicative of the sequence of events in operating the system, which in this case is the same as the flow of information in the main system. Sequential coupling exists in that the knowledge cannot be assessed, transferred, or applied until it has been acquired. The design of each of the level-one systems can be independent from the others, with the constraint that everything is stored and accessed digitally.

The system for assessment of the knowledge is tied to the same sequence as the systems for acquiring and transferring knowledge. This indicates the designer's intent that the knowledge should be assessed before it is transferred or applied. The knowledge could physically be transferred and applied without being assessed; but this system might not be as valuable, since the assessment is intended to add value to the system by evaluating, sorting, and formulating the knowledge for optimum utilization.

Table 2. The design matrix for level one

	DP1: system for acquiring knowledge	DP2: system for assessing knowledge	DP3: system for transferring knowledge	DP4: system for applying knowledge
FR1: acquire knowledge	x	0	0	0
FR2: assess knowledge	s	x	0	0
FR3: transfer knowledge	s	0	x	0
FR4: apply knowledge	s	0	s	x

Acquiring knowledge can be decomposed at the next level down, level two, as shown in Table 3. This decomposition acknowledges that there are two sources of knowledge, outside and inside the organization. There are two separate systems for acquiring knowledge, one for outside and one for inside. Having these two items separate provides the opportunity to have the internal system formulate the created knowledge in a way to assist assessment and utilization, while maintaining the ability to accept any type of external knowledge. The internal knowledge creation system could be almost anything.

However, based on the postulation that all value-adding problems can be formulated as design problems, it would necessarily follow that the system would be based on developing FRs to suit CNs and then finding the suitable DPs and process variables (PVs, entities in the process domain). These two functions can be completely decoupled.

Table 3. Decomposition of acquiring knowledge, level two, branch one

FR1.1: acquire knowledge from outside the organization	DP1.1: external knowledge acquisition system
FR1.2: create knowledge within the organization	DP1.2: internal knowledge creation system

Assessing knowledge can be decomposed at level two to include formatting, evaluating, and sorting, as shown in Table 4. In a design problem, as this is postulated to be, the value is captured in matching elements between domains, i.e., CN-FR, FR-DP and DP-PV pairs, and in what is known about their performance. Note that one-to-one pairing is expected between entities all of the domains, except for the customer domain. CNs are usually stated in more abstract terms and frequently do not pair one-to-one with FRs. The formatting for applicability should therefore follow a system that utilizes the pairing of entities between the domains, and the kinds of decompositions and integrations that they comprised. Because the knowledge is only formatted for applicability and not formatted for searching puts the responsibility for complex searching on the transfer branch. Powerful search engines are now common, so this responsibility transfer should not be onerous. The evaluation for potential value would be based on how well this pair has performed, where there is performance data. Where there is not, then the potential value is based on compliance with the design axioms: the independence and the information content. The sorting based on value (2.3) would logically follow the system for evaluation. If this were to be decomposed, then one could imagine a system based on the certainty that a DP or PV would fulfill or manufacture its corresponding FR or DP. The design matrix for this branch at this level, assessing knowledge, is triangular. This follows from the nature of the formatting, evaluation, and sorting, all being based on applicability to produce value. As discussed above, the system for formatting drives the rest of the branch, and it is critical for a satisfactory result. The systems in this branch are linked by the domain pair formatting and evaluation. The sorting adds value to the data base, in that information can be more readily evaluated by the user.

Table 4. Decomposition of assessing knowledge, level two, branch two

FR2.1: format for applicability	DP2.1: system for formatting for applicability
FR2.2: evaluate for potential value	DP2.2: system for evaluation for value
FR2.3: sort based on value	DP2.3: system for sorting based on value

The decomposition of the transfer knowledge branch is shown in Table 5 as comprised of storage, uploading, searching, and downloading knowledge. These are all dependent on the storage system, which could be a system of computers connected to the internet with some kind of security system. Current web technology would facilitate access, uploading, searching and downloading.

Table 5. Decomposition of transferring knowledge, level two, branch three

FR3.1: store knowledge	DP3.1: storage system
FR3.2: upload knowledge	DP3.2: upload system
FR3.3: search knowledge base	DP3.3: search system
FR3.4: download knowledge	DP3.4: download system

The design matrix for this level and branch is partly triangular, as shown in Table 6. The diagonal Xs combined with the first column of Xs indicate that everything is dependent on the storage system, as stated above. Therefore, the storage system needs to be designed first and then the other components can be designed in any order.

Table 6. The design matrix for transferring knowledge, level two, branch three

	DP3.1: storage system	DP3.2: upload system	DP3.3: search system	DP3.4: download system
FR3.1: store knowledge	x	0	0	0
FR3.2: upload knowledge	x	x	0	0
FR3.3: search knowledge base	x	0	x	0
FR3.4: download knowledge	x	0	0	x

The decomposition of the application branch, where axiomatic design is used explicitly for structuring, fitting, and for evaluating appropriateness, is shown in Table 7. As implied above, axiomatic design would be used in further decompositions of the other branches. In 4.1, the problem would be structured consistent with axiomatic design, following the proposition on formulating problems as engineering design problems. The system for applying axiomatic design would include training in axiomatic design and software to assist in the execution. Fitting the knowledge to the problem would be done through the system of matching entities across the domains, such that, once a CN has been identified, then FRs, DPs and PVs that have been used previously to satisfy that kind of CN would be searched for and downloaded and then evaluated for appropriateness, i.e., potential to add value in solving the design problem of current interest. The appropriateness in axiomatic design is determined by the application of the design axioms (DP4.3).

Table 7. Decomposition of applying knowledge, level two, branch three

FR4.1: structure the problem	DP4.1: system for applying axiomatic design
FR4.2: fit the knowledge to the problem	DP4.2: system for matching entities across domains
FR4.3: evaluate the appropriateness	DP4.3: system for applying design axioms

4. Discussion

The notion of the value-adding process in this work comes from the satisfaction of customer needs or desires. The ultimate customer for a company is someone from outside the company who pays money to the company, in return for having the need or desire satisfied by an ultimate product or service that the company provides. The knowledge management system facilitates the addition of value to products and services within the boundaries of the company.

It has been proposed that axiomatic design is an approach to engineering design that maximizes the probability of success in fulfilling the selected design functions (Suh 1990). Therefore, it can be postulated that, based on the assumption of being able to formulate value-adding knowledge as the solution to design problems, the solution proposed here should have the best probability of success in fulfilling the functions of knowledge management systems, as stated in this work. The solutions proposed here are dependent on the functions proposed here. Others may interpret the customer needs differently and develop other functions for a knowledge management system and arrive at other solutions that may have different probabilities of success.

The knowledge within the system is structured primary for utilization rather than searching. This might not have been the case a decade ago. Current search engines are powerful and fast enough so that search ability and usability can now be decoupled. While the emphasis in this design is on formatting the knowledge for value, it is not required. Knowledge that has not been evaluated or formatted could be placed in a special section of the system and still be searched.

5. Conclusions

Axiomatic design can be used to develop the design of the high, abstract levels of a knowledge management system. The designed system focuses on applying knowledge, created within the company or acquired from outside it, to the addition of value in the solution of design problems.

This work also postulates that all intellectual value-added exercises can be formulated as engineering design problems. Based on this, the optimum formulation of knowledge is based on the best format for applicability to design problems. Using the axiomatic design process, this optimum formulation of knowledge is captured in the interactions of entities, i.e., couplings across domains between the entities and their probabilities of success in satisfying the CNs and FRs.

6. Acknowledgements

The author would like to thank ADSI (www.axiomaticdesign.com) for the use of Acclaro software, Greg Adams of Nypro and a graduate student at WPI for fruitful discussions, and Nam Suh for his encouragement and friendship.

7. References

C.A.Brown (2005) Teaching Axiomatic Design to Engineers – Theory, Applications, and Software, SME Journal of Manufacturing Systems, v24/3 to be published.

C.A.Brown, “Kinds of coupling and approaches to deal with them,” Proceedings of ICAD 2006, The Fourth International Conference on Axiomatic Design, Firenze – June 13-16 2006.

J.S.Brown (1998) “Research that reinvents the corporation,” The Harvard Business Review on Knowledge Management, Harvard Business School Publishing, Boston, p153-180.
M.N.Kennedy (2005) “Implementing the principles of the Toyota development system – myths and realities,” Target v21, n1, p5-10.

A. Kleiner and G. Roth (1998) “How to make experience your company’s best teacher,” The Harvard Business Review on Knowledge Management, Harvard Business School Publishing, Boston, p137-151.

D. Leonard and S. Straus (1998) “Putting your company’s whole brain to work,” The Harvard Business Review on Knowledge Management, Harvard Business School Publishing, Boston, p109-136.

J.M. Morgan and J.K. Liker (2006) The Toyota Product Development System, Productivity Press, New York.

I. Nonaka (1998) “The Knowledge Creating Company,” The Harvard Business Review on Knowledge Management, Harvard Business School Publishing, Boston, p21-46.

N.P. Suh (1990) The Principles of Design, New York: Oxford University Press.

N. P. Suh (2001) Axiomatic Design - Advances and Applications, New York: Oxford University Press.