Management Science and Engineering **ISSN 1913-0341**
Vol. **4**, No. **2**, **2010**, pp. 69-81 **www.cscanada.net**
www.cscanada.org

Augmenting the Participatory Design Concept in Systems Development

Indeje Wanyama[1]
QIN Zheng[2]

Abstract: Participatory Design (PD) is an effective tool for designing organizational systems where views, aspirations and the input of both the system users and developers are sought and reconciled in the development of a system. This paper attempts to highlight and identify the fit between the Soft Systems Methodology (SSM) as applied in systems development and the tools of the Quality Function Deployment (QFD) as applied in manufacturing and how that fit does enhance Participatory Design in systems development. By recognizing the complementarities of the tools of these two approaches (SSM and QFD), we can enhance Participatory Design in systems development. Findings from literature review show that a comprehensive application of this concept is yet to be done in information systems development. The approach builds on the seven phases of Soft Systems Methodology by applying the Quality Function Deployment techniques to elicit information from complex and amorphous real-world situations to augment the Participatory Design process.
Keywords: Participatory Design; Soft Systems Methodology; Quality Function Deployment; House of Quality

INTRODUCTION

The development of organizational systems is characterized by having both “hard” and "soft" aspects. The hard aspects typically deal with hardware and software systems, while the soft aspects deal with organizational, political, and cultural systems - the so called human issues. While much has been written about how to address the hard aspect of designing organizational systems, the soft or human aspect has received comparatively less attention. However, the lack of success of many organizational systems has more to do with the failure to consider non-technical factors in design and development rather than

[1] Shanghai University of Finance and Economics – School of Information Management and Engineering, China. windeje@yahoo.com.
[2] Shanghai University of Finance and Economics – School of Information Management and Engineering China.
* Received on April 15, 2010; accepted on June 8, 2010

technological failings of the system. Several factors complicate design of successful organizational systems. The systems must obviously perform the tasks and functions that the users require. While eliciting requirements from users in traditional systems is difficult enough, organizational systems typically have a broader and more diverse constituency to satisfy, where the entities involved may be from different functions or enterprises. The design must align the viewpoints, needs, and objectives of the organizational groups. Differences of opinions must be resolved in an amicable fashion if the system is to be accepted and used by those involved de Vreede, et al, (1997). The design must not only consider the needs of users, but managers, technologists, and all others potentially affected by the system. Again, this requires the integration of diverse and often conflicting interests. Collaborative or participative design has been proposed as a way to overcome these difficulties. This paper presents an approach for collaborative design that uses Soft Systems Methodology (SSM) as its framework. The approach provides process guidance in the steps to perform for designing an organizational process and incorporates a set of established tools for accomplishing several of the steps.

It's worth noting that this is not essentially a new methodology but instead it is an attempt to integrate aspects of two well-known methodologies, one that has been well applied in systems development and one that has its foundations in the manufacturing field. The following section therefore provides a review of the literature specifically targeting information system development projects that have used the concept of participatory design and to what extent the tools of both SSM and QFD have been applied.

LITERATURE REVIEW

In this section we review documented cases where the concept has been applied as proposed in this paper. More specifically, how QFD has been applied to systems development by looking at specific case studies and different QFD-type methodologies proposed by various authors.

One of the earliest published papers on the subject of applying QFD to software development was by Zultner R. E. (1990). Zultner proposes a framework called Software Quality [Function] Deployment (SQD). This approach follows the idea of the deployment of the 'voice of the user' throughout the entire software development process. The problems associated with the development of software are a consequence of not properly defining customer requirements, as such; SQD is an attempt to address this problem. The approach is not used in isolation but as a complementary framework to conventional software development approaches and project management techniques. The process is split into a number of phases. Each phase involves the production of a number of matrices describing the relationships between the various user and technical requirements, processes and entities. The starting point for SQD is the customer or "the source of the voice – the user".

This approach is very detailed and is very much tailored towards structured systems analysis and design with a high focus on processes and entities. In the initial phases of SQD where the focus is on user requirements and technical requirements, this is pretty much conventional QFD and in particular the development of the house of quality.

According to Betts M. (1990) software has its own jargon, own representations, several schools of thought or even none to some and is still developing as a discipline. However, software is similar to other products in three areas; customer voice, decision support and engineering representation. On the other hand, software is also different in terms of manufacturing, staging and customer segmentation. To use QFD in software development the author advises to start out simple, consider product differences and

adapt the presentation format. Furthermore, this approach uses a set of QFD matrices in association with the product life cycle.

While illustrating how QFD can be used to enhance the system engineering and system development process and provide a visual capture of the decisions and target values made as the program transitions from system development through to product and process design, Melton, (1994a) noted that System Engineering and System Development techniques were standard in the U.S. Department of Defense (DOD) for decades. The QFD initiative in the US for hardware developed items follows a similar structured and disciplined process very analogous to the system engineering process implemented by DOD. He further illustrates how QFD can be used to enhance the system engineering and system development process and provide a visual capture of the decisions and target values made as the program transitions from system development through to product and process design Melton, (1994b). He demonstrates how QFD can be integrated into the system engineering and system development process to provide complementary benefits and aid decision making in defining and specifying a system.

Mcdonald (1995) examines the need for incorporating product development activities related to customer satisfaction into the software development process. He notes that QFD is an established technique for understanding and satisfying customers that is readily applicable to software development. He compliments this by a case study which offers an example of how this advanced quality technique applies to software development.

Presley et. el (1997) also used the QFD techniques in the development of a system to assist in the justification of investments in strategic technologies. The project required the integration of the efforts of three geographically dispersed teams of industry experts. The use of the QFD assisted in the identification, categorization, and prioritization of requirements for both the justification process and tools. The results of meetings held with each of the three teams were integrated using trained facilitators. A decision support system to support the process was then developed. Requirements for the information system were developed concurrently with the requirements for the process itself.

Dearden, A. and Howard, S. (1998) present a methodology for capturing requirements and priorities that can be used in the development of highly innovative interactive systems. Their focus was the development of interactive systems that cannot be treated simply as incremental improvements over existing products, in which case it is not possible to identify user requirements on the basis of empirical techniques, as there are no instances of use of the product from which to collect data. Their premise being that, developers of innovative products must proceed by envisioning the use of the proposed product and examining hypothetical interactions with potential or surrogate users. They propose an approach that provides structure to the process of envisioning and analysing hypothetical use. The methodology they propose combines techniques drawn from scenario-based design, and from Quality Function Deployment (QFD).

Pinto (2006) uses the QFD to uncover and systematize the factors and dimensions involved in the data representation issue and more exactly in the planning and design of the information products in an endeavor to optimize access to the increasing amount of information. The author notes that, by means of these deployments (QFD), the analysis of the factors and dimensions and their corresponding relationships provides an excellent picture for the quality planning and design of information products and representation processes.

Taha (2006), attempts to incorporate the QFD to be integrated strategically in designing and managing e-information provision within a networked library service environment. The project uses a University Library as a case study, where the evidence-based librarianship (EBL) approach is employed in three studies to identify user needs and acceptance of e-services.

Islam et al (2007) evaluates the performance of the website of a popular private television station. By means of an online survey and a few focus group discussions, they identify the website viewer requirements and their corresponding importance level. Subsequently, the technical requirements are identified that can fulfill the viewer requirements. A House of Quality (HOQ) is built to find the relationship between the viewer and technical requirements. The Quality Function Deployment (QFD) is used to provide the prioritized technical requirements, which then guide the website development and maintenance team.

As much as the review we have presented here is not in any way exhaustive, one thing is clear though, that the literature shows that a complementary application of both the SSM and QFD techniques in systems development has not been documented to identify the fit between the two. Whereas most of the recent developments in information systems development methodologies have applied the QFD techniques, few if any, have integrated it with the SSM tools. We therefore propose to identify areas where the two tools can fit and be used in systems development is a complementary manner.

In the following section, we will attempt to define Participatory Design (PD) as applied in information systems development. PD is a tenet of both the SSM and QFD. This will be followed by a brief description of the Soft Systems Methodology concept in the next section. Thereafter we will provide the Quality Function Deployment tools, before we introduce and discuss the fit between the two in the subsequent section followed by concluding remarks and future directions.

PARTICIPATORY DESIGN

Levinger, D. (1998) defines PD as a methodology in which representative end users provide continual feedback to computer systems designers during the development of system prototypes. This collaborative team of people represents the major stakeholders in a product or system design effort. By bringing these “domain experts” together, a vital link is established where users can interact directly with designers in the development process, with their suggestions for product improvements before those suggestions are codified into a program. The intent is to create designs that reflect the way the end-users actually use the product in their work.

According to Magnusson (2001) Design knowledge exists in all those potentially affected by a design, and they can all contribute to design a better product. This is carried out in a social process of communicating, sharing, reconciling, and acting.

It is a process of mutual instruction, where designers and end users learn from each other. The more one shares a social and cultural back-ground [environment], the more one shares a language, the more one participates in the design process. Participatory Design demands not only that end users share in the design process, but also that the designer shares in [work situations].

Participatory Design has its origins in Scandinavian trade unions’ initiatives toward democratization in the workplace. The objective was to include the perspective of the worker, concerning the introduction and development of new technologies. The aim was to strengthen the workers’ position in regards to the introduction and use of computer technology. The original concept was one of “work-oriented” systems design. Ehn, P. (1992), states that, “Democratic participation and skill enhancement – not only productivity and product quality – themselves [were] considered ends for the design”. One concept, the Collective Resource Approach (CRA), was fostered in Norway, Sweden and Denmark. This approach to systems design promoted the notion of collective cooperation between two different areas of expertise

(systems technology and end user experience) in the systems design process. By so doing, the most favorable conditions would be created for understanding the demands and requirements that a particular computer system would need to address.

Within the last two decades, the heavy focus on ”work-orientation” and ”democracy in the workplace” has given way to more socio-technical aspects of user participation; the influence and use of Participatory Design reaches far beyond the computer systems development arena to include such broad and diverse fields of product development as community housing and children’s tutoring aides. Whereas, Participatory Design can and has been applied in other fields, this paper attempts to bring to fore aspects of Participatory Design that are inherent with the “Quality Function Deployment (QFD)” a design tool widely applied in engineering. This shall be done within the SSM framework with a view to identifying the fit between the two.

SOFT SYSTEMS METHODOLOGY

Soft Systems Methodology (SSM) was first introduced by Peter Checkland in 1981 in his book *Systems Thinking, Systems Practice* (Checkland, 1981). SSM has been grouped among the “soft” operations research tools as opposed to the “hard” mathematical and decision models that have traditionally existed in the operations research field. It is a methodology for analyzing and modeling hard to define and complex systems that integrate both a technology (or hard) system and a human (soft) system. The latter system is defined by Checkland as a Human Activity System (HAS) and is posited to be different from natural systems or designed systems due to the introduction of the subjectivity of human desires and objectives into the HAS.

A HAS is defined as a collection of activities in which people are purposefully engaged, and the relationships between these activities. Checkland proposes that the same methods used for engineering technology may not work as well for the more unpredictable and complex human side of the system. SSM addresses fuzzy problems with unclear and multiple objectives and several different perceptions of the problem. SSM recognizes that different individuals will have different perceptions of the situation and different preferable outcomes. It recognizes these differences and explicitly attempts to take these into account from the outset to ensure that the results of the analysis are acceptable to all parties concerned. SSM does not attempt to define a single right method of action but, through an iterative process, defines an acceptable improved path of action. People who are involved in the methodology include not only actors within the designated system, but also clients and owners of the system. Because of these characteristics, it is proposed that SSM can be seen as a valuable framework for Participatory Design activities.

SSM is a seven-stage process in which users, analysts, and designers incrementally define the problem, generate and evaluate alternatives, and choose an acceptable solution. Rather than go into detail about these stages at this point, we will present them as we discuss the proposed approach. This will allow for a clearer identification of the linkage between SSM tools and QFD hence our proposed approach. Specific tools identified to assist in each stage of SSM will also be presented as the discussion develops.

The application of SSM in practical settings has gained popularity with scores of applications over the years Ledington et al (1997). Some settings related to our discussion here include Dang et al. (1995), who applied SSM to end-user business modeling. They enhanced SSM with more precision to the representation of soft-system knowledge and working memory. The authors emphasized SSM use for business modeling since it supports the needs for a set of higher level modeling constructs and a

methodology that guides how to model a business with the available constructs. Hsu and Yeo (1996) describe a case study of SSM to aid in reengineering a public organization involved in innovation and technology management. Instead of the application focusing on the innovation itself, they describe an application to introduce a business process for technology development. Macias-Chapula C. A., (1995) has used SSM to identify the value, impact, and barriers to information access and use, as related to quality of health care by a group of regional directors for a Mexican national health care organization. The innovation focused on new information systems and their requirements for the organization. In this case, there was an administrative innovation for a public service organization.

QUALITY FUNCTION DEPLOYMENT

Many companies use standardized methods for the translation of customer wants and needs into product and process properties (Schütte, 2002). One of the most common methods is QFD. The Quality Function Deployment (QFD) method was developed at the Kobe Shipyard of Mitsubishi Heavy Industries, Ltd., and has evolved considerably since then. In 1978 the first book on the subject was published in Japanese and then later translated into English in 1994 (Mizuno and Akao, 1994). QFD facilitates translation of a prioritized set of subjective customer requirements into a set of system-level requirements during system conceptual design. In Akao's words, QFD "is a method for developing a design quality aimed at satisfying the consumer and then translating the consumer's demand into design targets and major quality assurance points to be used throughout the production phase. ... [QFD] is a way to assure the design quality while the product is still in the design stage." As a very important side benefit he points out that, when appropriately applied, QFD has demonstrated the reduction of development time by one-half to one-third (Akao, 1990). A similar approach may be used to subsequently translate system-level requirements into a more detailed set of requirements at each stage of the design and development process. The sequence of activities which constitute the QFD method are shown in Figure 1.

In the context of system engineering, QFD facilitates a strong correlation between customer requirements and design requirements, and the inclusion of supportability requirements within the spectrum of design requirements. As such, the method goes a long way in making the customer an integral part of early design synthesis, analysis, and evaluation activities.

In essence therefore, QFD is a systematic planning process designed to explicitly incorporate the 'voice of the customer' into product design. Used in the manufacturing industries (Akao, 1990, and Cohen, 1995) to prioritize customer needs and map these needs onto technical constraints, applications of QFD have been done in the software domain including (Shindo, 1991 and Zultner, 1992). QFD's primary components include a structured planning process which draws input from cross-functional teams working within a concurrent engineering paradigm. Ongoing decision making and communication is supported through the use of a graphical notation termed the 'House of Quality'. Although aspects of the planning process are hardware specific (Cohen, 1995), the House of Quality notation appears to work well for software development (Shindo, 1991 and Zultner, 1992, Ala-Siuru, 1993 and Lundell and Williams, 1993).

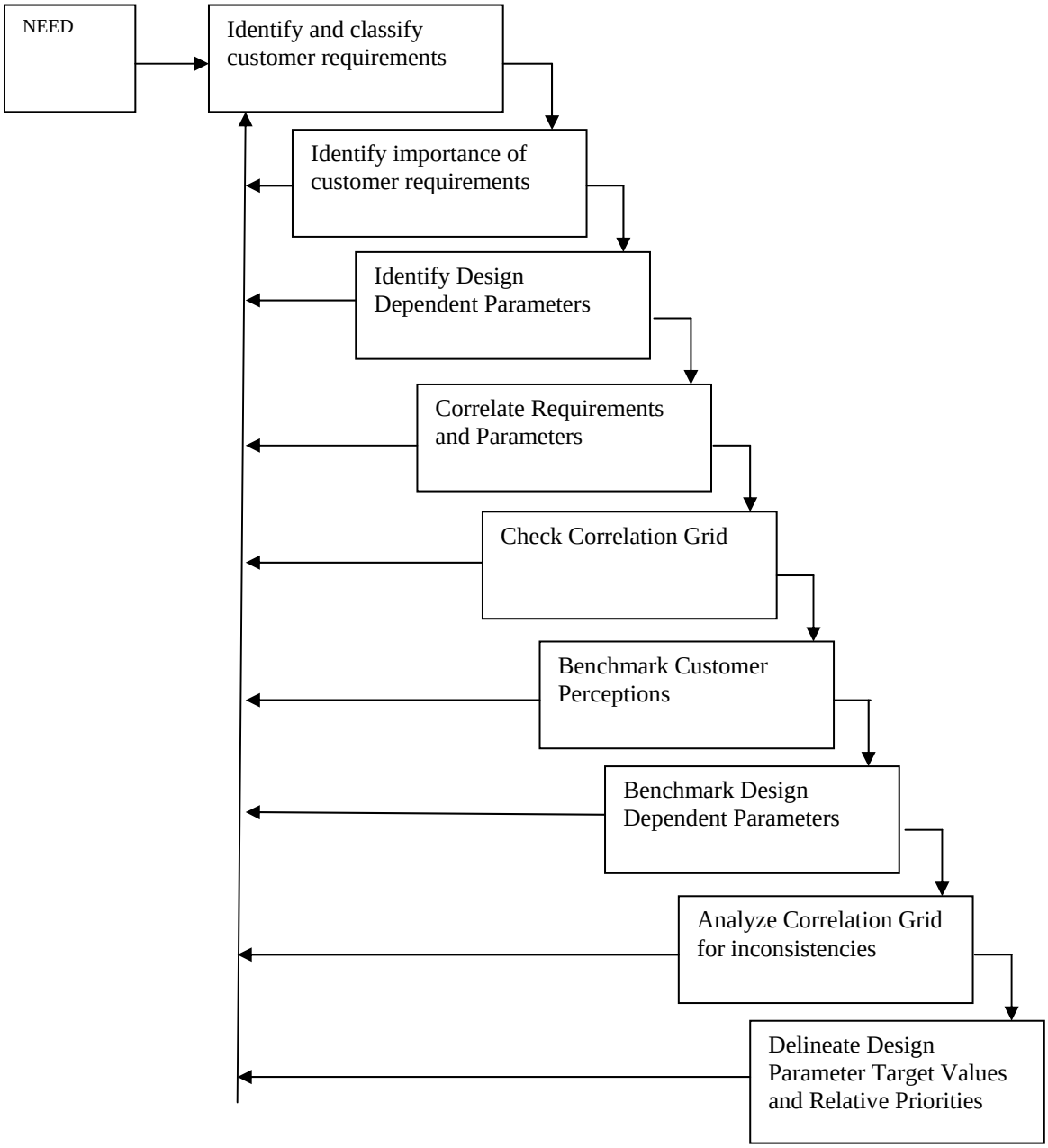

Figure 1: The Quality Function Deployment (QFD) process.[3]

In the proposed approach, (QFD) is used to support several stages of the soft-systems methodology. We provide this brief introduction to facilitate the description of the development process. The basic premise of QFD is that only the customer can define the quality of a product or service. It is a method employed to convert customer requirements into directions and actions that can be deployed through

[3]Adopted from Verma et al Quality Function Deployment (QFD): Integration of Logistics Requirements into Mainstream System Design.

planning, engineering, and productivity disciplines. Fig. 2 illustrates the central element of a QFD analysis—the relationship matrix often referred to as the "house of quality" because of its general shape. The House of Quality starts with a "What-How" Matrix that identifies the wants, desires, and needs of the customer (Akao, 2004). These customer requirements are shown on the left part of the House of Quality. The matrix lists the customer requirements (or "whats") along the rows. Each of these requirements has an importance value elicited from the customer. Design requirements for meeting the customer's requirements are listed along the top of the matrix. The "roof" represents the relationships among the various design requirements. The roof assists in identifying the synergies and tradeoffs that may exist in meeting the design requirements.

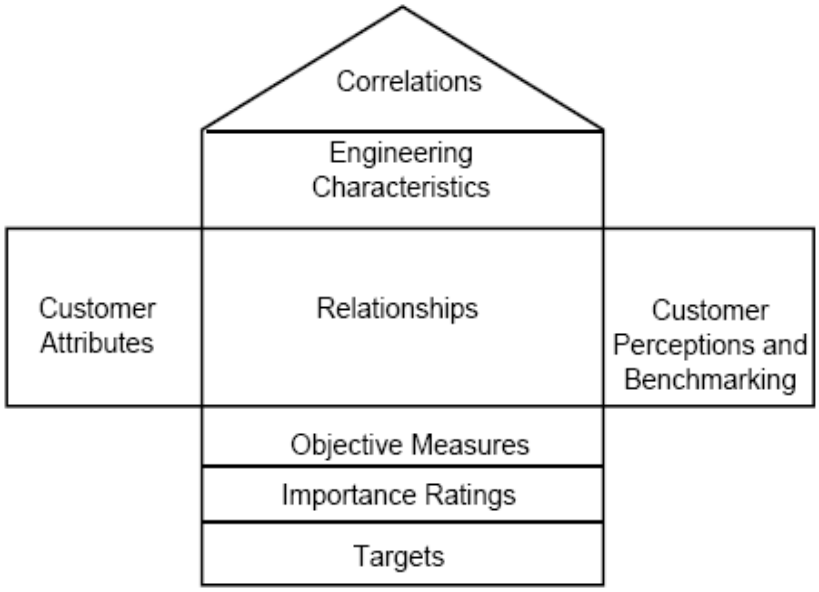

Figure 2: "Rooms" in the House of Quality (HOQ)[4]

The right-hand side of the house shows the comparative evaluation of competing alternatives. The central portion of the house is made up of a number of cells that relate the design requirements to the customer requirements. The relationships are typically specified as "strongly related," "moderately related," "weakly related," or "not related," and the matrix is developed using a symbol for each. The bottom of the matrix incorporates importance measurements, and target values of each of the design requirements. The importance measurements are usually computed based on the number and strengths of relationship of each design requirement to the customer requirements.

QFD APPLICATION WITHIN THE SSM CONTEXT

The foregoing sections briefly outline aspects of both the Soft Systems Methodology and the Quality Function Deployment tools indicating their relevance to Participatory Design is systems development. In

[4] Adopted from Lou Cohen - see also Hauser and Clausing.

this section we propose an approach that can enhance the powerful tools of the Soft Systems Methodology by incorporating Quality Function Deployment procedures. We shall therefore discuss the proposed approach in relation to the steps of SSM, and will discuss pertinent tools and techniques that we think are valuable in accomplishing the steps. The Soft Systems Methodology consists of a seven stage process: 1) problem situation unstructured; 2) problem situation expressed; 3) root definitions of relevant systems; 4) conceptual models; 5) comparison of conceptual models with the real world; 6) feasible, desirable changes; and 7) action to improve.

Stages 1 and 2 are most often combined in descriptions of the SSM. They represent the identification and representation of the problem situation in terms of a "rich picture." A rich picture is a representation of the problem situation, typically presented in the form of an abstract drawing, which describes aspects of the system that are relevant to the problem definition. The problem situation is defined by these two stages. The identification of the problem situation may come from several sources including managers, employees, or users.

These stages are executed through the use of teams of potential users, owners, and client. Typically, two teams are required, a "customer" team and a "development" team. The customer team consists of those individuals who will be users of the system, or other individuals most affected by the system. Selection of the customer team members is a critical aspect of the approach, as they will define the problem domain and the requirements of the system to address the problem. The development team may consist of domain experts from a broad background, as well as facilitators adept at extracting information from potential users.

In software, 50-60% of the software defects originate in the requirements phase. Quality Function Deployment is a proven technique that can reduce the number of defects, subsequently resulting in gains for product development and customer satisfaction. The goal of this SSM phase is to elicit the requirements of a customer/user. As highlighted above the Quality Function Deployment (QFD) is an especially valuable tool in eliciting requirements from a diverse group as is the case in Stages 1 and 2 of the SSM. QFD is a means of translating the "voice of the customer" into product, process, and production requirements using a series of matrices (Houses of Quality). It is a means of translating and prioritizing customer requirements into the appropriate technical requirements for each stage of product, process, and service development and implementation. It has been successfully used in software and hardware development (Haag, et al., 1996) and can be easily adapted to other projects.

The customer requirements are input into the "product planning" HOQ, which helps to determine the design requirements that are the inputs into the "part deployment" HOQ. The HOQ development is accomplished through a set of meetings with the customer teams. Customer requirements are elicited from the customer team and categorized. Consensus relative importance ratings are then developed for the requirements. There are several group decision-making techniques available to facilitate the identification of the importance ratings. One such technique that has been successfully applied is the Analytical Hierarchy Process (AHP), Saaty, (1996). Design requirements are then developed. These design requirements represent the characteristics of an innovation required to meet the customer requirements. Target values (customer satisfying levels of performance) for each of the design requirements are developed by the development team and experts. The team then identifies the relationships between the customer and design requirements, along with levels of relationships. This portion of the approach requires iteration to ensure that the requirements are grouped and categorized, as well as complete.

SSM advocates a pictorial representation of the "rich picture." While not the usual pictorial representation, the QFD houses can be considered a representation of the problem situation. Further, this approach does not rule out the development of the usual rich pictures used in other SSM application.

In Stage 3, a "root definition" is developed. The mnemonic CATWOE is used to guide the development of a root definition: **C**ustomer: people affected by the system; **A**ctor: people performing activities in the system; **T**ransformation: the transformation carried out by the system; **W**eltanschauung: the "**W**orld-view" or viewpoints held of the system; **O**wners: the person(s) with the authority to decide how (and if) the system will be carried; and **E**nvironment: the larger system within which the system under consideration exists and operates. The elements of CATWOE emphasize the need to examine the problem from a number of viewpoints. The Root Definition and CATWOE provide the analyst with a framework for ensuring that all points of view and interest are considered in the requirements elicitation.

Stage 4 includes the construction of a conceptual model identifying what the system needs to accomplish including its activities and their interactions. These activities describe what has to happen for the system to meet the goals and aims defined in the root definition. Checkland (1981) states that the conceptual model should focus on what is done, not how it is done. The process model that is developed CATWOE can then be directly mapped to design and customer requirements from QFD. This mapping requires another level of iteration that can link the process or product back to requirements and specifications that have been determined through QFD.

In Stage 5, the conceptual model is compared with the real world system to highlight possible areas where changes are necessary. This conceptual model will identify where problems or deficiencies exist between what is happening (the 'rich' picture) and what is desirable (the 'root definition') as defined by the models. In Stage 6, changes to address the 'disconnects' or gaps between the conceptual model and the real world identified in Stage 5 are introduced and evaluated for feasibility. These alterations may include changing the way certain activities are completed, or could result in the identification of activities not currently achieved in the real world.

SSM recommends the development of what are essentially "as-is" and "to-be" models of the situation under consideration. This model development is certainly possible with the approach being described here. However, rather than attempting to model the as-is situation as recommended by SSM, the QFD requirements analysis is conducted in Stage 6 to define the desired future system, without the need for current methods, technologies, practices, etc., to bound the design.

An addition to this step is the conversion of the model into the implementation mechanism. This implementation can be in the form of new procedures or processes, software, or any number of different deployment systems. The actual system is dependent on the innovation being pursued, and any deployment requirements defined during the QFD analysis.

Finally, in Stage 7, recommendations for change are implemented. These changes then result in a modification of the problem situation. This new situation may then lead to a new cycle of the approach as described above.

CONCLUSION AND FUTURE DIRECTIONS

In this paper, we have proposed that incorporating Quality Function Deployment in Soft Systems Methodology (SSM) is an effective approach for enhancing participation in the design and development

of systems. Without negating the importance of each of these approaches, we have highlighted the complementarities inherent in each, which if well utilized can be more targeting and rewarding. Diverse groups of participants come together to identify requirements for organizational systems. By incorporating customers (not only developers) into the development process, the approach described has potential for the development of other organizational systems. This review provides support for the value of SSM for Participatory Design. SSM is a general methodology to which we have attached the QFD tool. Generally, when the House of Quality is applied to software design, the resulting software requirements are diverse in their scope and coverage. The result is that product acceptance extends beyond basic functionality to serve as an indicator of reliability, usability, and other customer preferences and design considerations. Nonetheless, we believe that even without this extra tool, SSM offers a valuable framework for Participatory Design and that other tools are available to guide and facilitate the process. We also believe that the approach and tools we have proposed would benefit greatly from the use of other tools including Analytic Hierarchy Process (AHP), to facilitate the collaborative design process.

REFERENCES

Akao, Y. (1990). *Quality Function Deployment*. Cambridge, MA: Productivity Press.

Akao,Y. (2004). *Quality Function Deployment: Integrating Customer Requirements into Product Design*. Cambridge, MA: Productivity Press.

Al-Siuru, P. (1993). Using QFD Techniques in User Interface Specification: first experiences. In *Proceedings of the Fifth International Conference on Human Computer Interaction*. Pub: Elsevier Science.

Betts M. (1990). QFD Integrated with Software Engineering, *2nd Symposium on Quality Function Deployment.*

Checkland, P. (1981). *Systems Thinking, Systems Practice*. London: John Wiley & Sons.

Cohen, L. (1995). *Quality Function Deployment: How to make QFD work for you*. Addison Wesley ISBN 0-201-63330-2.

Dang Q. C., Baylis T., and Patel D., (1995). *Modeling business through soft systems methodology in end-user development: A claim and an approach*, presented at the Proc. 7th Int. Conf. Adv. Inform. Syst. Eng. 1995 (CAiSE 95), Jyvaskyla, Finland.

De Vreede, G-J., Dean, D., Lee, J., Vogel, D., and Orwig, R. (1997). Collaborative Design of Organizational Systems in *Proceedings of the Association for Information Systems*. Indianapolis, IN.

Dearden, A. and Howard, S. (1998). Capturing user requirements and priorities for innovativeinteractive systems. *Computer Human Interaction Conference, 1998. Proceedings. 1998 Australasian Volume , Issue 30, Nov-4*, pp. 160 – 167.

Ehn, P. (1992). “Scandinavian Design - on skill and participation" in Adler, P. And Winograd, T. (eds.): *Usability - Turning technologies into tools*, Oxford University Press.

Haag, S., Raja, M. and L. Schkade. (1996). Quality Function Deployment Usage in Software Development. *Communications of the ACM 39(1) ,* pp. 41-49.

Hauser and Clausing, (1988). The House of Quality. *Harvard Business Review 3, May-June*, 63-73.

Hsu J. P. and Yeo K. T., (1996). "A systematic approach to re-engineer a public research institute (PRI) for commercialization," *International Journal of Project Management*., *Vol. 14, No. 6*, pp. 387–393.

Islam, Rafikul, Ahmed, Mohiuddin and Alias, Masliza (2007): Application of quality function deployment in redesigning website: A case study on TV3. Published in: *International Journal of business Information Systems 2 (2),* pp. 195-216.

Ledington P. and Donaldson J., (1997). Soft OR and management practice: A study of the adoption and use of soft systems methodology, *J. Oper. Res. Soc., Vol. 48, No. 3*, pp. 229–240.

Levinger, D. (1998). Participatory Design History. PDC 98 *Proceedings of the Participatory Design Conference*: *Broadening Participation*. Seattle, Washington.

Lundell, J. and Williams, D. (1993). Integrating QFD into Software Development. In *Proceedings of the Fifth International Conference on Human Computer Interaction*. Pub: Elsevier Science.

Macias-Chapula C. A. (1995). Development of a soft systems model to identify information values, impact and barriers in a health care information system. *Journal of Information Science, Vol. 21, No. 4*, pp. 283–288.

Magnusson, C. (2001). *Introduktion till designmetodik, rehabiliteringsteknik och design* (in Swedish), Certec, Lund Technical University, Sweden.

Mcdonald M. (1995). Quality Function Deployment - Integrating Product Development into the Systems Development Process. *The 7th Symposium on QFD* (ISBN1-889477-07-9).

Melton D. L. (1994a). Quality Function Deployment in Concurrent Engineering and System Development Process. *The 6th Symposium on QFD* (ISBN1-889477-06-0).

Melton D. L. (1994b). Integrating Quality Function Deployment (QFD) in to the System Engineering and System Development Process. *The 6th Symposium on QFD* (ISBN1-889477-06-0).

Mizuno, S. and Y. Akao, ed. (1994). QFD: *The Customer-Driven Approach to Quality Planning and Development*. Asian Productivity Organization, Tokyo, Japan, available from Quality Resources, One Water Street, White Plains NY.

Pinto M. (2006). Data representation factors and dimensions from the quality function deployment (QFD) perspective. *Journal of Information Science, Vol. 32, No. 2*, 116-130.

Presley, A., Sarkis, J. and Liles, D. (1997). A Multi-Criteria Justification Methodology. *Journal of Engineering Valuation and Cost Analysis, 1(2),* pp. 111-123.

Saaty, T. L. (1996). *Multicriteria Decision Making: The Analytic Hierarchy Process*, 2nd edition, Pittsburgh.

Schütte, S. (2002). *Designing Feelings into Products- Integrating Kansei Engineering Methodology in Product Development*. MSc thesis in Quality and Human-Systems Engineering, Sweden.

Shindo, H. (1991). *Quality Function Deployment Applied to Software at Digital Equipment Corporation*. Thayer School of Engineering, Dartmouth College.

Taha, A. (2006). Streamline e-information service for virtual users: A quality function deployment (QFD) approach. In C. Khoo, D. Singh & A.S. Chaudhry (Eds.), *Proceedings of the Asia-Pacific Conference on Library & Information Education & Practice 2006 (A-LIEP 2006), Singapore, 3-6 April 2006* (pp. 141-147). Singapore: School of Communication & Information, Nanyang Technological University.

Verma, D., A (1994). Fuzzy Set Paradigm for Conceptual System Design Evaluation, Dissertation Manuscript, Virginia Tech, Blacksburg, Virginia.

Zultner R. (1990). Software Quality Deployment, Adopting QFD to Software, *2nd Symposium on Quality Function Deployment*.

Zultner, R. (1992). Quality Function Deployment for Software. *American Programmer Magazine.*