Jouko Koski

Quality Function Deployment in Requirements Engineering: A Review and Case Studies

Thesis submitted in partial fulfillment of the requirements of the MBA Program

Espoo, December 8, 2003

SUPERVISOR

Petri Parvinen, Associate Dean
Executive School of Business
Helsinki University of Technology

INSTRUCTOR

Jyrki Kontio, Professor
Software Business and Engineering Institute
Helsinki University of Technology

Helsinki University of Technology
Executive School of Business

Abstract of the MBA Thesis

Quality Function Deployment in Requirements Engineering: A Review and Case Studies

Jouko Koski, M.Sc. (Tech.)

Espoo, December 8, 2003

Number of pages
$vi + 102$

Supervisor
Petri Parvinen, Associate Dean, HUT Executive School of Business

Instructor
Jyrki Kontio, Professor of Software Product Business, HUT

Capturing customer needs is crucial to building products of high quality. Quality function deployment (QFD) is a method for getting in touch with the customer and for using this knowledge to develop products that satisfy the customer.

This thesis describes the QFD method, surveys the literature for QFD experiences, and analyzes four real-life application cases of the method. The thesis synthetizes a set of guidelines for applying QFD in requirements engineering.

Preface

This thesis research was carried out in the Competitive Advantage through Stakeholder-Driven Requirements Engineering (CORE) research project in the Software Business and Engineering Institute (SoberIT) at Helsinki University of Technology. The research project is funded by the National Technology Agency of Finland (TEKES) together with about ten industrial partners.

Several people have contributed to my work. First, I would like to thank my instructor, Dr. Jyrki Kontio for the invaluable advice and great care that he gave me whenever I faced problemss with my research. I would like to thank my colleagues in the CORE project—Marjo Kauppinen, Tero Kojo, Sari Kujala, and Laura Lehtola—for empathy, ideas, and inspiring discussions. I would also like to thank the interviewees and other representatives of Vaisala Oyj for their time and co-operation.

I would like to thank my supervisor, Dr. Petri Parvinen and the staff of the MBA program for their support. Particular thanks go to Carola Juselius, who managed to guide to the right track a student who appeared to get lost continuously. I would like to thank Ruth Vilmi for checking the language of my thesis.

Finally, I would like to thank my wife Tuula and my daughters Helinä and Terhi for all their love and care, and tolerance of Daddy's occasional grumpiness during this project.

Espoo, December 8, 2003 Jouko Koski

Contents

Chapter 1

Introduction

1.1 Background

High product quality provides a competitive advantage for a company. The quality of a product is highly dependent on how well the product satisfies customer needs. Therefore, capturing customer needs is crucial to building products of high quality. Quality function deployment (QFD) is a method for getting in touch with the customer and for using this knowledge to develop products which satisfy the customer. In QFD, the knowledge of customer preferences is called "the voice of the customer".

In software engineering, the term "customer" is often used rather loosely and with a broad meaning. A better alternative is to use the term "stakeholder", which refers to all customers, users, maintainers, etc. involved with the software system, and use the term "customer" precisely for the customers of the system. Stakeholder needs, wishes, and expectations concerning the system are restated as requirements. Requirements engineering covers all of the activities involved in discovering, documenting, and maintaining a set of requirements for a system. The QFD method can be used as a tool for carrying out the activities in requirements engineering. Stakeholder requirements represent the voice of the customer in the software development process.

Analyzing heterogeneous stakeholder needs and transforming them into competitive features of a product is not a trivial process. It is one of the focus areas of the Competitive Advantage through Stakeholder-Driven Requirements Engineering (CORE) research project. The CORE project is being carried out in the Software Business and Engineering Institute (SoberIT) at Helsinki University of Technology together with approximately ten industrial partners.

The CORE project team observed experiments of use or suggestions of use of three methods in partner companies: conjoint analysis (CA), analytic hierarchy process (AHP), and quality function deployment. CA and AHP can be seen as methods for supporting decision-making. QFD is also a decision-support tool, but it is also a method for organizing activities in a development process. Therefore it was suggested that applying QFD would be a promising approach, but that more knowledge about the method was necessary. Interest in the project was shared by both the researchers and participating practitioners.

This study will investigate the QFD method itself and the attempts to apply the method. The study will strive to make four main contributions: First, it will give an overview of the QFD method. Then, it will give a state-of-the-art review of reported experiences of QFD in software engineering as well as review and analyze QFD application cases. Finally, it will suggest application guidelines for practitioners and experimenters with QFD in the software engineering area.

1.2 Research problem

The goal of the study is to survey QFD experiences in the literature and in the real-life application cases in order to synthetize a set of guidelines for applying QFD in requirements engineering.

The QFD method has its origins in manufacturing industry, where it has been applied successfully. The success is probably due to some characteristics of QFD that make it effective. Understanding these characteristics could help in making QFD successful also in software engineering.

1.3 Objectives of the study

The study has four main objectives. The first objective is to create understanding of the QFD method: how it works, the principal ideas on which it is built, and how it might best be applied. The method will be reviewed in the context of other tools used for similar purposes. An understanding of the application areas will be developed, since the QFD method is not widely used in the development of software products.

The second objective is to summarize the pre-conditions of successful application of the QFD method: what the success factors in QFD projects are, what the limitations of the method are, and whether there are any particular pitfalls in applying the method. A specifically interesting issue will be the applicability of the method to requirements engineering.

The third objective is to review the experiences of practitioners in applying QFD in real-life projects: how the method was adopted, how it was applied, and whether the project was successful. It was anticipated that practitioners might have interesting ideas as to what might be done differently in the future.

The fourth objective is to synthetize a list of guidelines for applying QFD in software development projects: how to prepare to using QFD in the first time and what are the key issues in applying the method. The viewpoint for the guidelines will be small and medium size organizations who are trying to adopt QFD in discreet steps.

1.4 Scope of the study

This study will concentrate on the quality function deployment method. QFD can be seen as a tool for decision-making. While there are other methods and tools, they will be covered only to a minimal extent in order to put QFD in context. Likewise, there is whole branch of research covering the problems and mechanisms of decision-making, however, in this thesis, the decision theory will be considered only in so far as it helps in developing an understanding of the effects of QFD characteristics.

The study will not cover all aspects of QFD in manufacturing industry.

The bias is towards software engineering and software related products. Other application areas will be covered only to give an overview of the applicability of the method.

The work will be based on completed research and projects. There are no plans for doing experimentation or giving guidance on ongoing projects.

1.5 Research approaches and research methods

The approach of this study will be mainly nomothetic in the sense that the research aims to discover empirically valid guidelines for applying the QFD method. There will be two research methods for meeting the target: literature review and interviewing practitioners.

A literature review will be carried out to describe the QFD method and to position it in the context of other decision support methods in product development. The review will be expanded to material describing the application of the method and to the documented experiences of its usage.

Practitioners in a company participating in the research project will be interviewed. Interviews will be carried out in a semi-structured manner. A list of prepared questions will be sent to the interviewees before a meeting, but the interview will follow the list loosely. All interviews will be recorded for later analysis. The number of interviews will be small; hence the study will not be concerned with statistical validity.

1.6 Structure of this work

The material in this work is divided as follows:

In Chapter 2, the concepts and procedures of the quality function deployment method are presented. The purpose is to give an understanding of the QFD techniques and their application to different areas. QFD is also compared to some other decision-support methods.

The utilization of QFD in software engineering is described in Chapter 3. The main use of QFD is in the field of requirements engineering. The software engineering domain has certain characteristics that make slight adjustments to the QFD method inevitable. These adjustments come in the form of several different approaches, none of which has become predominant. However, the approaches do share some common parts and features.

The purpose of Chapter 4 is to give an analysis of the QFD characteristics and present the prerequisites for the successful application of QFD in development projects. The observations and conclusions are derived from the literature.

Interviews and case studies are presented in Chapter 5. Practitioners' experiences are gathered in four separate projects in a Finland based company, Vaisala Group. Each application case is briefly presented and the chapter is summarized by setting out the findings of the case studies.

Chapter 6 synthesizes a list of recommendations for applying QFD in requirements engineering. The bias is in utilizing the method for the first time in a relatively small organization or organization unit.

The final chapter draws some conclusions from this study.

Chapter 2

Quality Function Deployment

This chapter describes the quality function deployment method and how it is applied in industry. The description covers the basic elements of QFD and the phases of the application process. Literature sources are used to give examples of QFD variants and approaches on different application areas.

2.1 Overview

2.1.1 The concept

New businesses typically emerge because unsatisfied customer needs are detected. New offerings to satisfy such needs may be innovative and unique services or products. No matter how effectively a company meets the initial needs of its customers, it must remain constantly alert and responsive to its customers' continuing wants and needs (Day 1993).

Customer satisfaction can be monitored using questionnaires and registering and receiving customer complaints. Often monitoring fails to reveal the real wants and needs of customers; this is because it is based on sales and marketing inputs, rather than on structured and consistent questioning. Understanding customers' wants and needs properly requires a major effort, which itself requires careful planning.

Quality function deployment is a process—a method—for planning products and services. It starts with customer needs and wants—the voice of the customer—which become the driver for the development of the requirements for the product or service. The QFD process is best conducted through teamwork, because there are many inputs and decisions involved. This kind of approach tends to remove organizational barriers and effectively links the company to its customers. The organization works more cooperatively, and the new product or service has an increased potential for satisfying its ultimate customers (Day 1993).

The concept of quality deployment was first proposed by Dr. Yoji Akao in 1966 (ReVelle, Moran & Cox 1998). The method has its origins in Japanese heavy industries where it was developed and adapted during the 1970s. The first book on QFD was published in 1978 by Dr. Shigeru Mizuno in conjunction with Akao. Toyota Auto Body adapted QFD in the late 1970s and made refinements to quality tables. The refined structure of tables became later known as ”the house of quality” matrix. QFD was formally introduced to the USA in 1983. The first case study outside Japan was reported in 1987. Since then, QFD has been used in ever widening circles. The application of QFD to software development began in Japan in 1982, in North America in 1988, and in Europe in 1990 (Zultner 2000).

QFD encourages a proactive development approach. More time is spent in the initial planning phase, but fewer changes are made to the product in the later phases compared to a more reactive approach. In early stages, the changes are frequently made to plans and concepts rather than to materials, parts, or implementation. These “paper changes” are faster and less expensive to make. The reactive approach requires more people and therefore results in additional cost.

QFD uses matrix techniques for advanced cause-and-effect analysis as well as a form of quality assurance. The techniques are not difficult to use. Day (1993) summarizes some of the key issues of the QFD concept:

- QFD is a planning process as opposed to a tool for problem solving or analysis.
- The customers’ wants and needs—their requirements—are the inputs to the matrix. The process cannot begin without these inputs. QFD

essentially forces an organization to get in touch with the people who use its products.

- It uses a matrix to display information vital to the project in brief outline format.

- This collection of information in the matrix format facilitates examination, cross-checking, and analysis. It helps an organization set competitive targets and determine the priority action issues.

- The output resulting from analysis of the QFD matrix is twofold: First, competitive targets are established for key action items related to the customer's voice, and second, certain priority issues are selected for special emphasis. An effective response to the targets and to the selected priority issues will result in increased customer satisfaction.

2.1.2 The process

QFD is a technique for requirements engineering borne from the quality movement. It did not originate as a requirements engineering technique, but rather as a systematic method for translating customer requirements in to specific product design targets. It can be seen as one of the applicable tools in the total quality management (TQM) concept (Day 1993).

A matrix format is used in QFD to capture a number of issues important to the planning process. However, the purpose is not to build matrices, but rather to get in touch with the customer and to use this knowledge to develop products, which satisfy the customer. The matrix helps in collecting information from various sources and permits the organization to examine the information in a multidimensional manner. The process follows the phases of filling-in the information in the matrix.

The QFD matrix has two primary parts (Figure 2.1 on page 9). The first primary part is the customer portion as the QFD process starts with the customer. Customers' wants and needs are expressed in their own language or jargon. The voice of the customer is complemented with a measure of the relative importance that customers assign to each of the

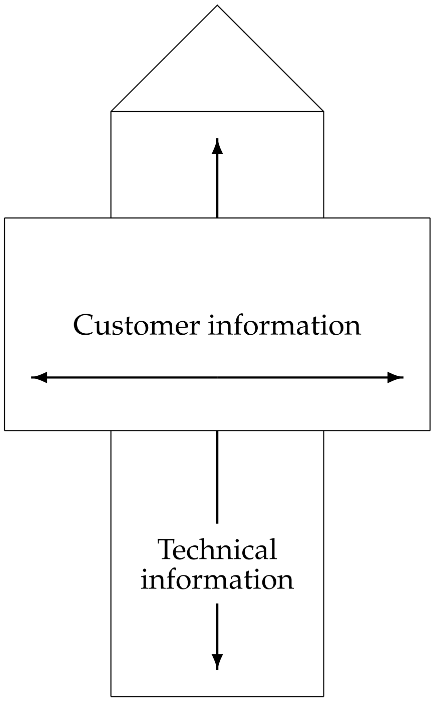

Figure 2.1: Two primary parts of the QFD matrix

voices. Likewise, the number of complaints can be used to indicate the importance of a specific voice. The customers' competitive evaluation of the product or service makes it possible to rate the offering against that of competitors.

The second primary part of the QFD matrix is the technical information portion. The process continues with determining how the company will respond to each voice. The technical or design requirements the company will use to describe and measure each customer's voice are placed across the top of the matrix. The technical portion represents the "hows" to the "whats" given in the customer portion.

The customer and technical portions intersect in the center of the matrix. This area gives an opportunity to record the presence and strength of relationships between inputs and action items.

Each technical requirement can be analyzed in order to evaluate the performance of the company against that of its competitors. Results are presented in the part of the matrix below the intersection or relationship area.

The information in the matrix can be examined and weighted by the QFD team. The team can set goals or target values for each technical requirement. These goals—the "how muchs"—are stored in the bottom part of the QFD matrix.

The described items "what", "how", "relationships", and "how much" are the four parts of the basic QFD matrix. QFD applications in various areas generally employ the four-part basic matrix.

Tradeoffs can be examined in a triangular matrix on top of the basic QFD matrix. This is done by comparing each technical requirement to other technical requirements one by one. The purpose is to determine the net effect that changing one requirement has on the others. The shape of the triangular matrix resembles a rooftop; that is why the QFD matrix is sometimes referred to as "the house of quality".

The QFD process does not necessarily stop at the completion of the basic QFD matrix. The team can use the outputs from the basic matrix as inputs to the subsequent levels. For instance, the technical requirements and their measurable target values—the substitute quality characterstics—resulting from initial product planning can be used in determining parts deployment. In parts deployment, the substitute quality characteristics are the inputs according to which the necessary part characteristics are analyzed. The part characteristics are the inputs to process planning where the manufacturing operations are characterized. Further on, the manufacturing operations are the driver for the production requirements in production planning.

The QFD method is inherently flexible. The matrix representation is applicable in many kinds of cause–effect analysis and the team can insert additional items in to matrices. QFD is not a scientific method. Rather, it is a process that helps to analyze incomplete information.

2.1.3 The voice of the customer

Customer wants and needs or customer requirements are referred to as "the voice of the customer". The QFD process starts with capturing this.

Determining the voice is a complex process involving multiple steps.

The first step is to identify the target markets. The issue of target markets is vital, since it dictates who should be surveyed. Day (1993) suggests that the following list should be considered in determining which people to survey:

- Determine the target market.
- Determine the demographics.
- Determine the geographical distribution.
- Use a nonaffiliated survey organization.
- Survey people external to the organization.
- Survey with or without samples of the current product.

The goal is to capture customer needs and wants without hiding anything and without any particular bias.

Day (1993) presents approaches for obtaining the voice of the customer: focus groups, interviews, mail questionnaires, product clinics, and observations. Data can be gathered from multiple sources, for instance, from direct interviews and customer complaints received by product-support department. The questioning should be continued until the root want is discovered; it will be too late once the respondents have gone and the company personnel are back-home trying to figure out what the customer meant.

The process of questioning will not reveal everything involved in understanding the customer needs and wants. The Kano model (Figure 2.2 on page 12) helps to understand the types of customer expectations and satisfaction. During the interviews, customers typically refer to issues concerning the expected quality. For instance, a car should have good acceleration, low consumption etc. Satisfaction increases when expectations are met, while dissatisfaction arises when they are not. It is very important to know the real customer needs.

The customers seldom mention the basic quality issues, which are the "givens" for the product or service. For instance, there is an implicit

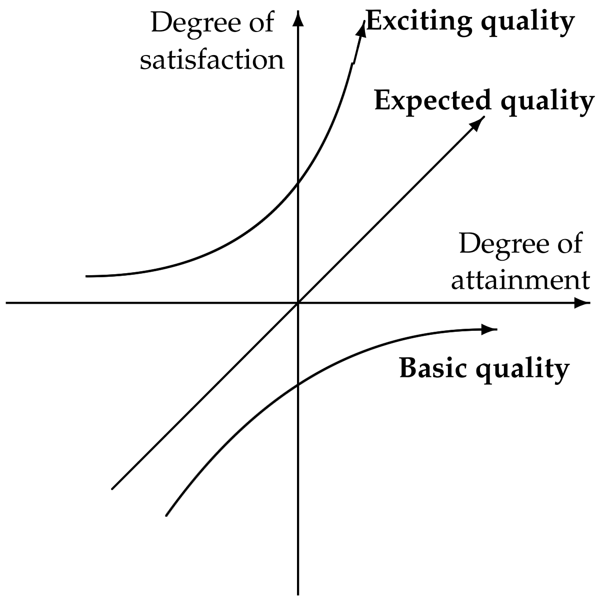

Figure 2.2: The Kano model

requirement that the engine of a car should start without trouble. However, if the basic functional requirements are not fulfilled, the customer is very dissatisfied.

Things that go beyond customer expectations represent "exciting quality". Typically customers give only indirect information about these issues in surveys. For instance, extra room for storing accessories, drinks, or small items in a car is probably not required, but nevertheless has a very positive effect on customer satisfaction. The lack of exciting quality does not increase dissatisfaction. The surveys should be carefully searched for clues for providing exciting quality. Generating product excitement often involves creative ideas, new approaches, or shifts in technology.

Through time, the expected quality features are likely to turn into basic quality features and the exciting quality features are likely to become expected features. Product or service development must continuously endeavor to find new ways to satisfy the customer.

Finding the root-wants of customers is essential to the survey process. Customers tend to mix needs, solutions, and problem concerns; therefore asking “why” is important. The actual verbatim comments should be documented. In processing customer voices, some voices need to be summarized to catch the essential point, while multiple voices representing the same issues are best consolidated verbatim into a single voice.

Organizing customer voices into natural groups can be a difficult task. The voices develop in a random manner. For using the information, the voices should be grouped according to relevant topics or subjects. The voices should represent approximately the same abstraction level. The QFD method does not specify any particular method of organizing the voices. Day (1993) mentions the affinity diagram process, where the voices written on small cards are sorted and grouped in a simple team process and where the grouping into higher-level groups continues until a satisfactory level is reached.

When the customer voices have been gathered and organized, undertaking the customer level of importance rating and the competitive evaluation should begin. Often these are carried out in a separate survey. Each respondent is not required to answer all the questions, which can make contributing to the survey a little easier. The collection of questionnaires covers all the voices.

The importance rating describes how essential fulfilling a requirement is to customer satisfaction. Rating is done on a numeric scale. Typically scales using odd numbers on scales 1 to 3, 1 to 5, or 1 to 9 are used, but also the use of some other prioritization method—like the analytic hierarchy process—is possible (Zultner 1998). Not all customers are likely to share the same opinions. Averages of respondents’ values may be used to reach a consensus, if there is team of respondents, group discussions can be used for this purpose.

Competitive comparisons are typically made with respect to existing products or services in order to indicate how well they meet customer expectations. The comparison should cover at least the principal competitor offering but a whole group of competitors is preferred. Scales for rating the products or services vary, but a numeric scale of from 1 to 5 often seems to be prereferred. Competitive evaluation may be problematic when developing a totally new product or service. Solution alternatives in these cases might be to compare the closest applications in the same genre or to make an in-house review of the best current offerings.

Surveying the customers' voice is a continuous process. It is necessary for feedback on accomplishments and for determining trends and changing voices.

2.2 QFD matrix

2.2.1 Customer information

The QFD product planning matrix has two major components: a horizontal customer information table and a vertical technical information table. The customer information table is developed from the captured voice of the customer. The team must do some grouping and processing, since the customer voices follow no order and the verbatim comments are not delivered in an organized manner.

The customer information table contains the developed customer requirements, accompanied typically by importance ratings and competitive evaluations. Competitive evaluations can be illustrated with a graphical diagram representation. The table can contain additional items like the number of complaints received in each requirement category (Figure 2.3 on page 15).

The initial version of the table can be considered as a pre-planning chart that needs some processing. Examining voices using the chart will help limit the number of voices by rejecting those that involve attributes, styling, or items that are on a lower level and should be handled in a subsequent matrix. Day (1993) suggests that matrices should have between 25 to 50 entries in the customer requirement area. While there have been matrices with hundreds of entries, 30 seems to be the practical limit.

2.2.2 Technical information

After developing the customer information table, the cross-functional team starts working on the technical portion. Work on this portion can also begin after the voices have been collected, if the project is time-critical. There is

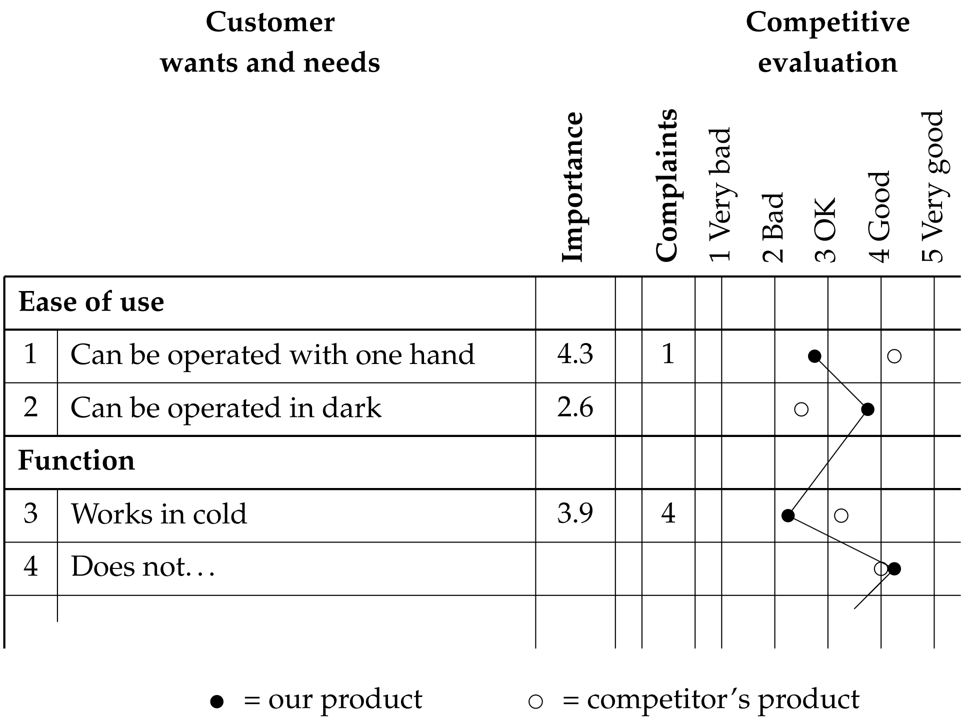

Figure 2.3: Customer portion of the matrix

no specified order in which information is entered into the matrix, but the sequence described is the one most typically followed.

First, the team should start translating the customer voices into technical requirements. The technical requirements are represented in a language that the company uses to describe its products for design, processing, and manufacture. The technical requirements must not represent solutions. Each technical requirement should be worked on to satisfy a voice, be measurable and global in nature, and should not imply any specific design intent (Day 1993). As a rule of thumb, the team should try to keep the ratio of technical requirements to customer requirements somewhere between 1 and 1.5.

Second, the team should examine the relationship between the technical requirements and customer requirements. Although numeric values can be used, the relationships are typically indicated using symbols. Usually a double circle or a filled circle is used for a strong relationship, a single circle for a moderate relationship, and a triangle for a weak relationship (Figure 2.4 on page 16). The team should work in columns, looking at each technical requirement and asking: “Would working on this technical requirement help to satisfy this customer requirement?” The relationship symbol is written in the intersection cell. After finishing with the requirements, the team should review the relationship portion of the matrix. There should be no rows or columns with no relationship symbols or with only weak symbols.

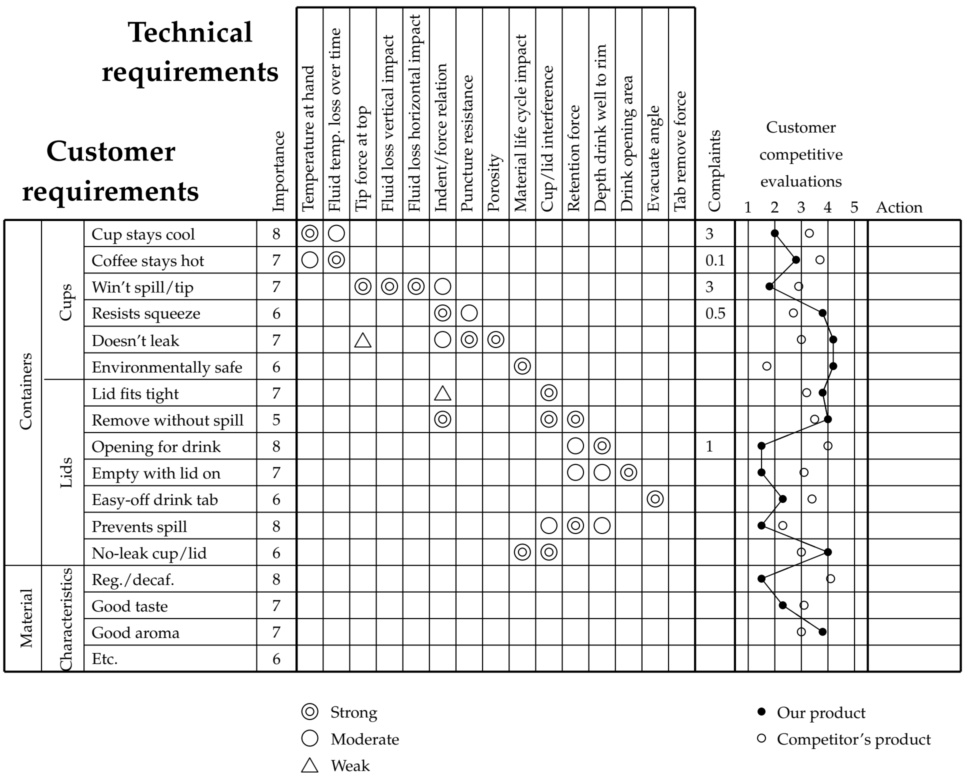

Figure 2.4: QFD matrix with technical requirements and relationships, after Day (1993)

In the third phase, the team should set a measurable target value for each technical requirement. This may also involve testing exiting or competitors’ products. The testing results of the products can be illustrated with plotted diagrams, just like the competitive evaluations in the customer information table. Associated with each target value there can be an indication of the direction in which the value migh be improved. The target values set concrete goals for design of the product or service. They can be also used later to benchmark how succesful the design was.

The final step in entering technical information is to analyze the co-relationships between technical requirements. Working to improve one may help a related requirement. On the other hand, working to improve one requirement may negatively affect a related requirement. Unlike technical requirements that should be generic and non-design-specific, the determination of co-relationships requires that a specific design is considered. The tradeoffs are entered into a triangular matrix on top of the technical information table. Each requirement is compared to others one by one. The co-relationships are often indicated using double-plus, plus, minus, and double-minus symbols.

The technical portion of the matrix can contain additional items also. When considered useful, the team can attach various characteristics to the technical requirements. For instance, field experience, organizational concerns, difficulty, or required effort related to each issue can be estimated. There is no single recipe for developing the QFD matrix.

2.2.3 Priorities

Development of the QFD matrix involves a major commitment by an organization. They are not willing to make such a commitment unless there is obvious value received. The principal purpose of developing a QFD matrix is to put the organization in touch with its customer needs and wants and to help determine the priority items for improved customer satisfaction (Day 1993).

Prioritizing is necessary to balance resources against requirements. Priorities can be determined by analyzing the QFD matrix. Each customer' requirement should be reviewed. By considering the competitive evaluations and the number of complaints, for instance, the team should determine actions for each requirement. The action can be a more thorough examination of concepts or analysis of a competitor's offering, for instance. If there is room for improvement in satisfying the customer's requirement, a target satisfaction level is set. A new "improvement ratio" column can be calculated by dividing the target value by the current value (Figure 2.5 on page 19).

The team can also consider other viewpoints in analyzing the customer requirements. For instance, marketing personnel can fill-in specific sales point factor values in a dedicated column. Higher values of this factor indicate the specific attractiveness of the related requirement in marketing.

The analysis of customer requirements is concluded with calculating weights for each requirement. This weight is given by multiplying the customer importance column by the improvement ratio and sales point factor. Typically the results are also normalized so that the sum of weights equals 1 or 100.

The team can develop action plans after the analysis of the customer requirements, and can also continue with calculating priorities for technical requirements. The priorities are calculated by summing up the relationships in each technical requirement column multiplied by the normalized weight of each customer requirement row. The relationship values can be 1 for a weak relationship, 3 for a strong relationship, and 9 for a very strong relationship. Also, here the results are typically normalized so that the sum of technical requirement priorities equals 1 or 100.

The team should keep in mind that the calculated priorities are just numbers for helping to make decisions. They are not a substitute for common sense. There are also other values that may be essential in making judgments. The approach ensures that all of the customer requirements developed during the QFD study make their impact on the technical specifications that the organization uses as a basis for its products (Day 1993).

2.2.4 Subsequent levels

Building the QFD matrix relating customer requirements to technical requirements constitutes the initial phase of the process—the product planning. Developing this house of quality matrix makes a major contribution

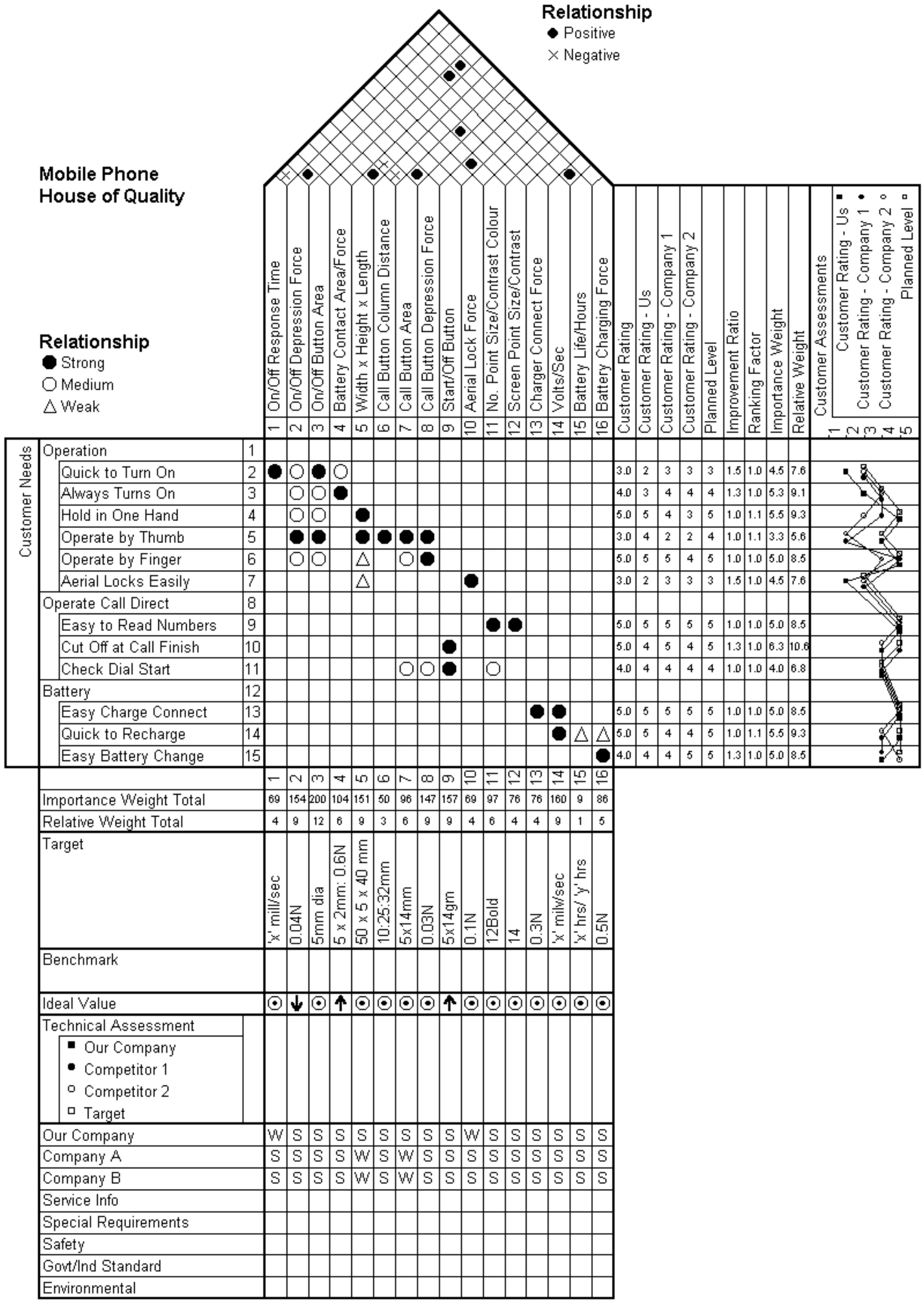

Figure 2.5: A QFD matrix produced with QFD2000 software (Total Quality Software 2003)

to product planning by bringing the company into contact with its customers. Day (1993) claims that if the QFD concept is used only for product planning, it will measurably improve a company's opportunities for long-range survival and success. However, the QFD process can also be a valuable tool for planning at subsequent levels of the product deployment, i.e. at the part, process, and manufacturing levels. Because of the number of phases, the overall process is often referred to as the four-phase QFD process.

One of the outputs of the product planning phase is the technical requirements. These substitute quality surrogates must not represent solutions. The solutions are considered in the next phase—the part deployment. The part deployment begins typically with concept selection.

Again, QFD uses matrices for illustration and analysis. One approach to concept selection is to use the Pugh concept selection matrix. The priority technical requirements from the first phase are used as input rows in the Pugh matrix. If there is a current product, its measured properties are used as reference values. Alternative solution concepts—the columns of the matrix—are judged either better than the current (+), worse than the current (−), or the same as the current (S). The analysis after the ratings consider the number of pluses and minuses, and the team determines whether the pluses overcome the minuses. No explicit weight factors are used. An alternative approach for concept selection is to use must–want analysis. In this approach, the requirement rows are divided into musts—the requirements the solution must satisfy—and wants—the requirements whose fulfillment level can be adjusted. The must requirement fulfillment is rated simply using "yes" and "no" values. The want requirements are rated using a scale from 1 to 10 based on the measured properties. The selection is based on overall fulfillment of the must requirements and weighted sum of the want requirements' ratings. Concept selection is usually accompanied by a more elaborate analysis of the strongest solution candidates and by a fault analysis.

The part-deployment phase concludes with the development of a part-planning matrix. In this matrix, the technical requirements are used as input rows. Columns represent critical part requirements. Each part requirement should have a relationship with a technical requirement. The relationships are rated using the same technique as in the product planning phase. The "how much" portion of the matrix represents the exact part specifications.

The third major phase is the process deployment. The goal of this phase is to determine the critical requirements of the processes that are necessary in producing parts specified in the previous phase. The structure of the process deployment is very similar to the part-deployment phase. Actually these two phases are often at least partially parallel, since the analysis required in the part-deployment phase needs consideration of the process issues as well. In the product-planning matrix, the design requirements responded to the question "What measurable items would we work on to satisfy this customer voice?" in the part-planning matrix, the requirements responded to the question "What are the elements we must control in the part to assure that the design requirements are met?" The process requirements respond to a similar question: "What are the elements we must control in manufacturing to assure that the part will meet its requirement?" (Day 1993)

Some process requirements involve conformity with a procedure. This is true particularly in service oriented processes, in which the number of procedural requirements is typically higher. Procedural requirements may be difficult to specify or measure and often do not have a relationship indication in the matrix; they are just part of the process.

The final phase is the manufacturing deployment. In the earlier phases, the "hows" from one matrix are transferred to the subsequent matrix and become the "whats". When the manufacturing phase is reached, the situation is slightly different, but the matrix-like table works well for initial manufacturing planning.

For instance, the general sequence of events is as follows (Day 1993):

- Critical part requirements are identified.
- The process steps that will affect variation of the critical part requirements are identified.

- The process variables that will influence part variation, such as time, speed, amount, and temperature, are determined.
- The operating windows for these process variables are then established. These are the windows within which the process must be operated to ensure that variation is under control.
- The last step is one of developing the manufacturing plans that define and describe the implementation of the necessary process controls to ensure operation within these windows.

If the organization does not identify the critical requirements during the part and process development, the manufacturing may end up treating all controls as equally important. Concentrating on typically large numbers of equally important factors is usually an indicator that effort is being wasted.

The outcomes of the manufacturing deployment phase are a manufacturing planning document, a quality assurance planning table, maintenance instructions, and operator instructions. The controls and timing of the actions taken in manufacturing reflect the importance of the requirements.

2.2.5 Alternative QFD approaches

As QFD users became more adept at applying the house of quality, they realized that there were matrices and tables that could assist them in organizing, controlling, and carrying out more of the design process with the same valuable results they had come to expect from using the house of quality (ReVelle et al. 1998). A couple of approaches have been popularized. One of them is the four-phase approach of the American Supplier Institute, which has been outlined in the previous section.

The GOAL/QPC (a not-for-profit educational institution) has been popularizing the matrix of matrices approach. The matrix of matrices approach presents a set of roughly 30 matrices arranged according to their intended content (Figure 2.6 on page 23). An individual matrix can be addressed using the letter of the column and the number of the row of the matrix in the set of matrices. For instance, the most common matrix—the house of quality—is often referred to as the A1 matrix.

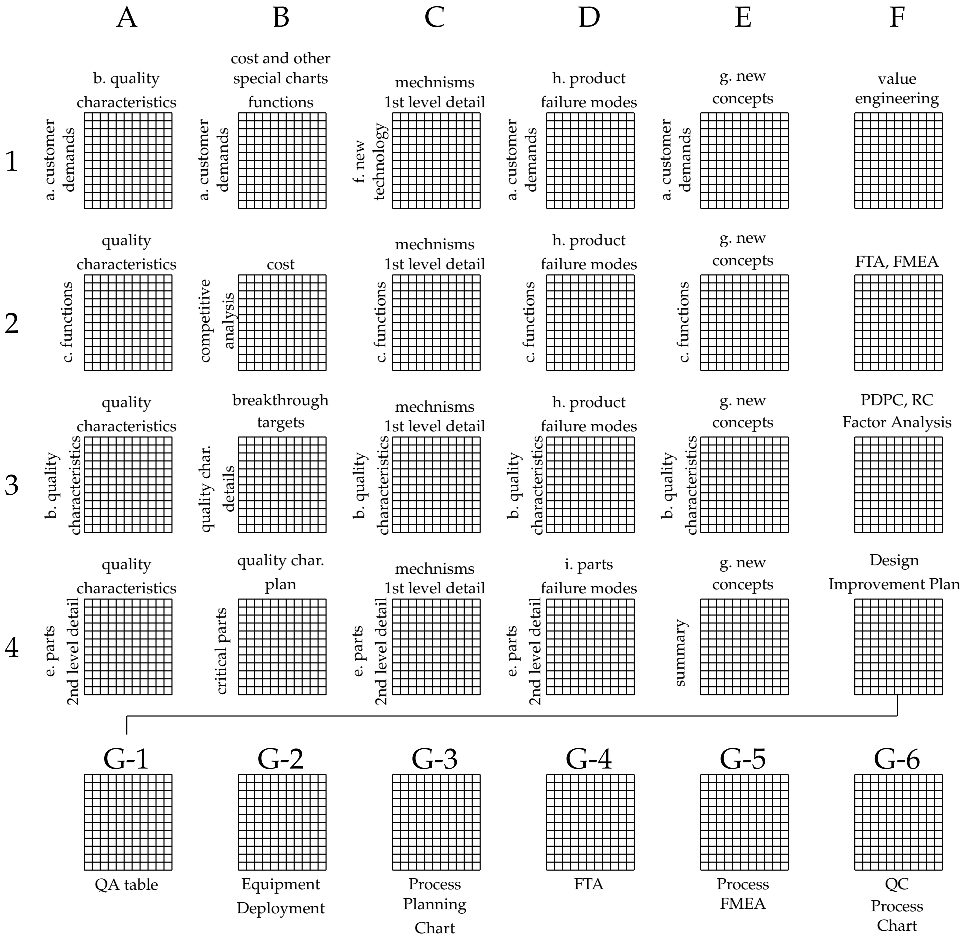

Figure 2.6: Matrix of matrices

The QFD concept is very flexible and dynamic and can be applied in the design of a multitude of complex products, services, processes, and systems. An ongoing problem when using the established approaches is that there is so much complexity that the design team may become confused as to which matrices need to be worked through, and in which sequence, to reach the goals. Not all possible matrices are relevant to all QFD projects.

The matrix of matrices approach is probably closer to the origins of the

QFD concept. QFD can be utilized at many levels of the process. There are some guidelines and even software tools (ReVelle et al. 1998) for selecting the relevant matrices from the set. The four-phase approach can be seen as a special case or an instance of a specific path in the matrix of matrices. Furthermore, the four-phase approach does not exclude the possibility of using auxiliary matrices when considered necessary.

Sometimes it is simply not possible to do a comprehensive QFD project involving any combination of the four-matrix or matrix of matrices approach. For these cases, Richard Zultner has developed a streamlined approach to QFD, called Blitz QFD (ReVelle et al. 1998).

Beyond the requisite planning of the QFD process, there are seven steps in applying Blitz QFD:

1. Gather the voices of the customers. The idea is to go to the "gemba"—the real place—to identify customers; focus only on the most important customers and their needs.

2. Sort the verbatim reports received from the customers. A customer voice table is used. In the customer voice table, those concerns that are most important to customers are the entered into columns; these provide the dimensions critical to the success of the project.

3. Structure the customer needs. Affinity diagram techniques can be used in arranging customer needs into their natural groups.

4. Analyze customer needs. This is done using a hierarchy structure. The emphasis is on understanding customer thinking, finding unstated requirements, and quantifying the needs.

5. Prioritize customer needs. The analytic hierarchy process allows the prioritization of the needs using ratio-scale numbers.

6. Deploy the prioritized customer needs. A maximum-value table is used for this analysis. Items that contribute the most to satisfying the most important customer needs are the maximum-value items.

7. Analyze only the important relationships in detail and only to the extent that is warranted.

Instead of using matrices as in the comprehensive QFD, the Blitz QFD uses various tables and structure representations as illustrated in Figure 2.7 on page 25. Not using matrices does not mean that QFD is not applied. The usefulness of the derived information is still there and can serve as the foundation of a comprehensive QFD project.

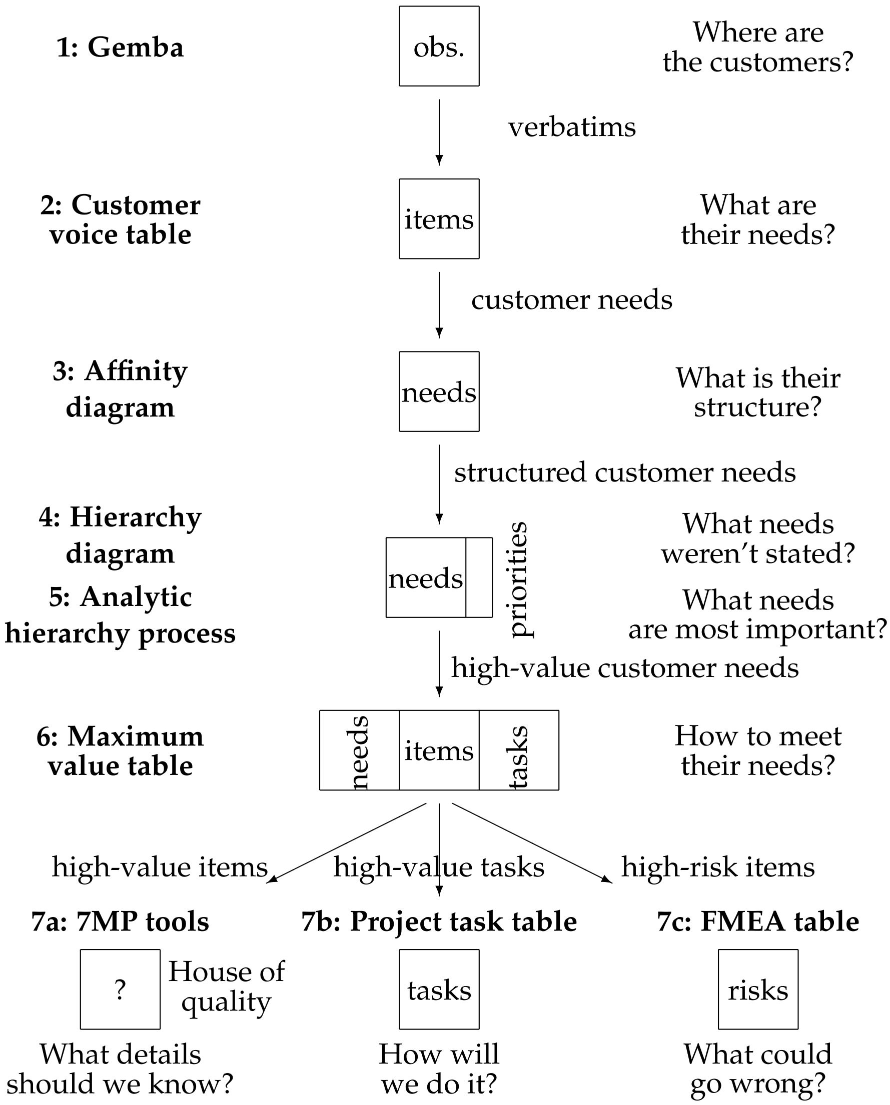

Figure 2.7: Series of Blitz QFD tables and diagrams

2.3 QFD application areas

2.3.1 Manufacturing

The origins of QFD lie in Japanese shipyards and car manufacturing industry. Many of the QFDs features and phases contribute in a quite natural way to the development process of manufactured products. The idea is to build quality into products before the production begins. Many other approaches focus on repairing defects during the production process or afterwards. The final product design has an excellent chance, once it is released, of going into fully ramped production with few, if any, engineering change orders. By simultaneously addressing internal and external needs, the result is a product that is easier to make and less costly to put into production (ReVelle et al. 1998).

2.3.2 Services

Service industries have "discovered" QFD and its ability to help in designing services. Some early applications (in 1981) of applying QFD to service organizations in Japan were for a shopping mall, a sports complex, and a variety retail store (Mazur 1998).

Mazur (1998) describes the following phases in applying QFD to services:

- Organization deployment is used to map the QFD steps to the different organizational functions. It shows who is responsible for what activities and when it occurs during the service planning and development process. It is recommended that organization deployment be done before QFD is applied to a specific service. The tools used are flow charts and matrices.

- Customer deployment is the deployment of organizational goals into core competencies, customer attributes, or target customer segments. Services often focus on niche markets and this helps to tailor the offerings accordingly. The tools used are analytic hierarchy process (AHP), matrices, and matrix data analysis charts.

- Voice of the customer deployment. Tables are used to record raw customer data, use characteristics, and separate the different types of service attributes.
- Quality deployment is used to translate customer-demanded quality and priorities into measurable service quality attributes. The tools are affinity diagrams, hierarchy trees, prioritization matrices, tables and AHP.
- Function deployment is used to identify functional areas of the organization that are critical to performing tasks that must achieve the quality-attribute targets. The tools used are affinity diagrams, hierarchy diagrams, and relationships matrices.
- Process deployment is used to diagram the current and re-engineered processes. The tool is a variant of the diagrams used in time–motion studies called blueprinting.
- New concept deployment is used in conjunction with a structured problem-solving approach to select a new process that will best satisfy customer needs. The tools are blueprinting and tables.
- Reliability deployment is used to identify and prevent failures of critical customer requirements. The tools are fault trees, process decision program diagrams, and relationships matrices.

Depending on the particularcase, some phases are more relevant than others. It is not necessary to work through all the phases.

Software development can be seen as a special case of services. It shares more common characteristics with services than with manufacturing. Applying QFD to software engineering is described in Chapter 3.

2.3.3 Business planning

In addition to developing products and services, QFD is applied also to process development and business planning. The QFD process encourages listening to customers—both external and internal—, provides a format for comparing inputs against outputs, and supports the tracking of flows.

Day (1993) outlines a business process containing successive levels (Figure 2.8 on page 28). Different organizations may use different terminology, but the highest level is the vision statement of a company. The vision resembles the customer voice in product planning in that it is broad and is not in specific operational language. The vision is normally translated into a set of objectives, which are more specific actionable issues. At subsequent levels, the organization can develop strategies and action plans. Measures can be established to evaluate performance. A review mechanism can be established to provide for review of progress.

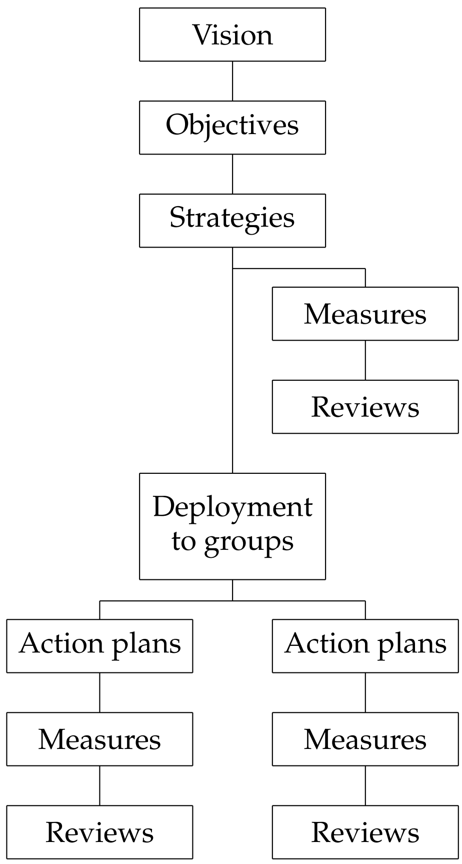

Figure 2.8: Conceptual diagram of a typical business planning process (Day 1993)

As the plan moves through a series of matrices, each subsequent matrix will add an additional level of detail to the overall plan. Most companies are reluctant to indicate levels of priority for issues as broad as the vision statements. Likewise, objectives may be so broad that there is no suitable measurement. Working on the objectives–strategies matrix, an organization may feel comfortable with assigning importance values and measurements. At the next level, action plans identify how the strategies will be accomplished. It is advisable also to analyze the interrelationships of inputs or outputs using the co-relationship matrices. Action plan responsibilities can be described using the matrix format. This matrix is useful in describing the review frequencies, responsible and supporting groups or persons, and special actions planned.

The matrix approach helps the organization effectively examine, relate, and track items. The matrix itself is not the objective. The approach supports the organization in its efforts to organize its thought processes in the area of business planning and business process re-engineering.

2.3.4 Other applications

QFD practitioners come in a variety of guises: marketing, engineering, manufacturing, assembly, quality and material, finance facilities, and human resources. Because of the diversity of backgrounds of QFD practitioners, some rather unique applications of QFD have been recorded. These include (ReVelle et al. 1998, Day 1993):

- Creation of an entire curriculum of education and training. Even the contents of a specific course have been identified using QFD to prioritize specific topics from a long list of potential topics.
- Modification of an existing software package and development of new software concepts.
- Creation of an individual job description.
- Zero-basing a collection of an excessive quantity of policies and procedures created by organization's corporate headquarters. Proceeding through the usual QFD steps, a prioritized listing of those

policies and procedures that should remain in the books and those that should be canceled was determined.

- Situational analysis and problem solving. Typically brainstorming and discussion is used for selecting the most likely problem causes. QFD can help in cause–effect analysis and in weighting the issues.
- Improvement planning. Analysis of customer care-abouts and employee care-abouts for development of core business processes and procedures.
- Creation of checklists. This involves analysis of customer concerns and the offerings to the customers to resolve their concerns.

2.4 Other tools for prioritization

There are a variety of decision-support methods that could be used in a design process. In the following, two of them are introduced briefly: analytic hierarchy process and conjoint analysis, because they are often considered as alternatives or complementors to QFD, particularly in prioritization of requirements. The approaches of these methods are discussed in the context of QFD, because it could alleviate the method selection and because the CORE research group has observed some interest towards these methods among the partner companies.

2.4.1 Analytic hierarchy process

Analytic hierarchy process (AHP) is a decision-support method for a quantitative ranking of a set of alternatives. AHP has been used for prioritizing stakeholder requirements directly—that is, without analyzing the factors constituting the priority—(Lehtola 2003), for cost–value analysis in requirement prioritization (Karlsson & Ryan 1997), and for determining the importance of customer requirements when building the basic QFD matrix.

When AHP is applied, the problem is structured as a means–ends objectives network. The overall goal is on the highest level, divided into contributing sub-goals, and the alternative choices are on the lowest level.

The process involves pairwise quantitative comparisons of nodes on the same level in the network. Comparisons are typically made using weights from 1 to 9 with odd numbers, but corresponding verbal statements can be used instead (Table 2.1 on page 31). The selection of the scale is justified by the fact that humans are not very good at comparing things that differ by more than one order of magnitude.

Table 2.1: AHP pairwise comparison scale

Numerical Value	**Verbal statement**	**Explanation**
1	Equal importance of both elements	Two elements contribute equally
3	Moderate importance of one element over another	Experience and judgment favor one element over another
5	Strong importance of one element over another	An element is strongly favored
7	Very strong importance of one element over another	An element is very strongly dominant
9	Extreme importance of one element over another	An element is favored by at least an order of magnitude

Results from the pairwise comparisons are stored in a matrix. When, for instance, item 1 is considered "strongly more important" than item 2, weight factor 5 is stored in matrix cell (1,2) and its reciprocal value $\frac{1}{5}$ is stored in cell (2,1). At the lowest level in the hierarchical network, the alternative choices are compared to other alternatives with respect to every subobjective at the higher level. If there are no intermediate levels—sub-objectives—and the number of alternatives is n, then the total number of $n(n-1)/2$ comparisons are required.

The value of each item can be determined using linear algebra and calculating the eigenvalues of the matrix. The value of each item is represented on a ratio scale. A value can be interpreted in the form "the alternative 2 contributes 37 percent of the total value in respect of the objective", for instance. Because of the redundancy in the AHP matrix—the reciprocal values in "mirror" positions—a consistency ratio of the resulting values can be calculated.

AHP was first introduced by Saaty (1980). Some characteristics of the method have been criticized, for instance by Salo & Hämäläinen (1997). A particularly controversial issue seems to be rank reversal. Rank reversal happens when adding a new—and maybe clearly poorer—alternative to an already ranked set changes the order of already ordered alternatives. The new alternative may emphasize the insignificant differences between existing choices and cause the reversal. Although this may be mathematically justified, decision makers may find the behavior very counterintuitive.

Lehtola (2003) has used AHP for requirement prioritization. One observation was that users found the use of reciprocal values in the matrix slightly confusing. Perhaps corresponding verbal statements or even a coloring scheme used in the experiments would be easier to comprehend. An observation was also that the odd-numbered weight scale from 1 to 9 tends to have too many values. Users used only a subset of the scale. The last observation in considering AHP was that the comparison matrix becomes impractically large when there are more than 20 requirements being prioritized. Users found it difficult to concentrate on such a large number of comparisons and the sheer size of the evaluation form caused problems.

2.4.2 Conjoint analysis

Conjoint analysis—or originally, conjoint measurement—is concerned with the joint effect of two or more independent variables on the ordering of a dependent variable (Green & Rao 1971). Its procedures require only rank-ordered input, yet yield interval-scaled output. The use of conjoint analysis originates in mathematical psychology in 1964. Since 1971 it has been widely adapted in marketing research. Conjoint analysis has been used also as a decision-support tool in the design process (Gustafsson, Ekdahl & Bergman 1999).

Conjoint analysis (CA) is any decompositional method that estimates the structure of a consumer's preferences (i.e., estimates preference parameters such as part-worths, importance weights, ideal points), given his or her overall evaluations of a set of alternatives that are pre-specified in terms of levels of different attributes (Green & Srinivasan 1990). Price is typically included as an attribute. Depending on the type of the analysis, different estimation methods—like monotone analysis of variance (MONANOVA), ordinary least squares (OLS) regression techniques, or a kind of multi-way frequency analysis (MFA) such as Probit—can be used.

In a typical case, a set of critical attributes of a product is chosen for analysis. For each of the attributes, a couple of viable levels or alternative values are chosen by product specialists or by carrying out pilot research. Customers are then asked to rank the combinations of all attribute values. Conjoint analysis attempts to find out how much each attribute contributes to customer preferences.

A problem is that the number of attribute value combinations can easily grow beyond practical limits. For instance, if a product has six distinct characteristic attributes and each attribute has four alternative values, the number of combinations—or profiles—is $4^6 = 4096$. Often this kind of full-factorial analysis can be avoided by estimating the orthogonal sets where the number of profiles is $6 \times (4 - 1) = 18$. Alternative approaches to presenting the profiles to customers are full profile, two-attributes-at-a-time, and the self-explicated approach. In the full profile approach the customer must rank combinations of multiple attributes, which may be difficult. The two-attributes-at-a-time approach is easier, but the customer may consider it boring because of the large number of comparisons. The self-explication approach lets the customers choose the attributes, which makes the comparison of answers harder.

CA and QFD provide similar recommendations on most dimensions (Pullman, Moore & Wardell 2002). Differences probably reflect a difference between what customers say they want and what managers think will best satisfy customer needs. The use of CA seems to be more market research oriented. The direct link between design features and choice as well as the ability to carry out individual-level analysis suggests that CA may be better able to predict the impact of design changes on sales, profit, and cannibalization. QFD appears to have more of a product-development or engineering flavor.

2.4.3 Alternatives and QFD

AHP and CA are linear impact analysis tools, in contrast to the QFD approach, which is an ordinal impact analysis tool (Hurri 2000). Linear impact analysis involves developing linear weighted additive utility models and assessing or estimating weights and criteria values. Ordinal impact analysis assesses the order of importance of alternatives via ordering criteria and consequences.

Each of the methods has a slightly different standpoint. The use of the methods is not by any means mutually exclusive. Instead, they can be used together as complementing each other. For instance, Zultner (2000) suggest the use of AHP for rating the importance of customer needs in Blitz QFD. CA could be used first to determine the most important features for a subsequent QFD study, or CA could be used to refine feature levels and improve predictions after QFD has screened the problem down to a smaller number of features (Pullman et al. 2002).

AHP and CA are more mathematically oriented than QFD, which involves only fairly simple and intuitive mathematics. QFD also has properties which support communication and the overall development process. Because the mathematics and method behind the scenes in AHP and CA are somewhat complicated, people may be reluctant to utilize the methods. If parts of the process work as a "black box" that they do not fully understand or control, they become suspicious and less confident of the value of the method. The selection and utilization of a tool is not conducted on mathematical grounds alone. Human and social factors are also significant.

In practical cases, the use of any of these methods should be supported by appropriate tools. In the simplest cases, the use of QFD techniques can be supported with a plain spreadsheet application. In more complex cases, or when utilizing AHP, CA, or other (mathematical) decision-support tools, the use of some dedicated software product is often inevitable.

Chapter 3

QFD in software engineering

This chapter describes the application of QFD in software engineering in detail. There are a couple of QFD approaches devoted to software products found in the literature. The presentation emphasizes differences between software QFD and traditional QFD as well as between different approaches in the software area.

3.1 Domain characteristics

Developing and producing software products have some significant differences from making hardware or manufactured products. For instance, software products are abstract, complex, and unique systems. Software is developed, not manufactured. Once the development of a product is finished, the product can be duplicated with no, or insignificant, additional costs since software products carry virtually no material or material-based costs. The factors of time and data become much more important in the overall process. Software development involves communication, since it is a human and creative phenomenon. The need for communication means that most software engineering tasks are not partionable. It is difficult to control all relevant parameters of the process. Because all systems are unique and technology evolves constantly, it is not straightforward to replicate success in other projects.

The QFD method has its most successful applications in the manufacturing industry. Some adaptations to QFD are necessary in order to make it usable in software engineering. The core of the QFD process—capturing the voice of the customer—fits naturally into the requirements engineering activity of the software engineering process. The requirements engineering covers activities for determining and maintaining the stakeholder requirements imposed onto the software system. However, since there are no material-related activities, the subsequent levels and phases of QFD must be altered and some structural changes made.

As simple as it might appear, applying QFD to software engineering has not been a straightforward task. Practices vary, and there seems not to be any single well-established and widespread approach for applying QFD to software. In the following sections, some of the most common basic models are presented.

3.2 Software Quality Deployment

According to Zultner (1998), quality deployment can be applied at various levels of sophistication, ranging from using just four basic matrices to 30 matrices, or even to 150 matrices. Basic to his approach—the Software Quality Deployment (SQD)—is the use of a variety of matrices to examine in detail the interaction of various dimensions. These dimensions include cost, customer demands, and facility structure, among others. The model follows the basic four-phase approach with some additional levels (Figure 3.1 on page 37).

QFD starts with "customer demands" in the house of quality matrix. In software projects, customer demands are the requirements of the users and other stakeholders. Usually software has to serve several classes of stakeholders. These stakeholder classes must be first identified, understood, and prioritized before beginning work on the house of quality matrix. SQD introduces a new matrix for classifying users—or actually stakeholders—according to their characteristics (Figure 3.2 on page 38).

After the users are characterized, the raw expressions of stakeholder

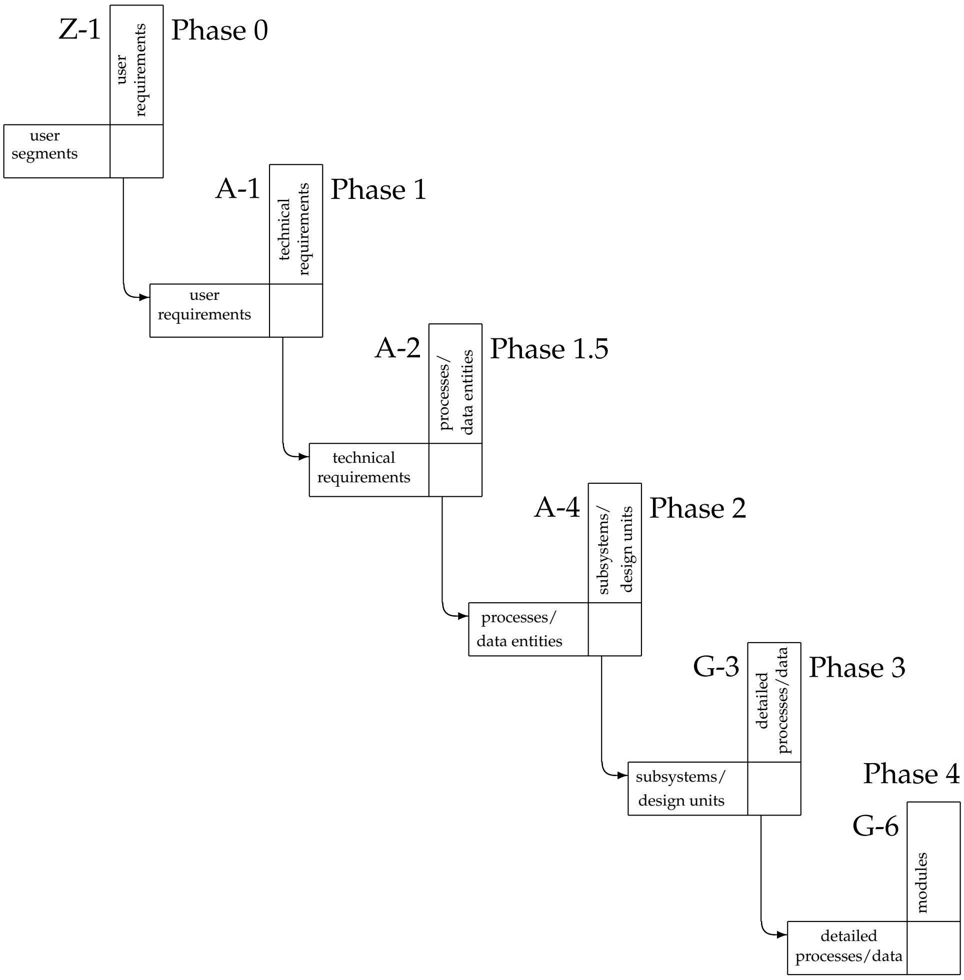

Figure 3.1: Phases and sequence of matrices in SQD

needs, wants, and concerns are gathered. This is accomplished by interviews, surveys, team analysis sessions, focus groups, trouble reports, problem logs, and compliments for any existing systems. The expressions must be refined into clear, consistent statements of user expectations. The statements are organized into a hierarchy of requirements and the relationship between the requirements and user segments are analyzed in a dedicated matrix.

The stakeholder requirements are refined further and their raw priorities are calculated. The analytic hierarchy process (AHP) can be utilized for this purpose. The raw priorities can be adjusted by an adjustment

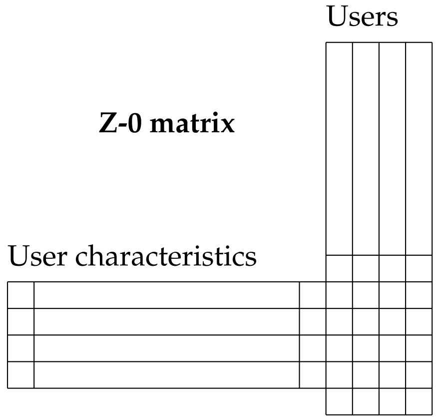

Figure 3.2: Z0 matrix for classifying users

factor such as the number of users in each category. The adjusted priorities are then deployed in the basic house of quality matrix. The raw priorities reflect what the users want most. The adjusted priorities reflect those users we want to satisfy most.

The user requirements are translated into technical requirements, using the house of quality matrix. The process is quite similar to the case of hardware or manufactured products. However, Zultner (1998) suggests that the matrix should not represent the “whats” and “hows” as in the usual case, but hint as to the “whys” and “whats”, as in the case of software. Anyway, the requirement weights are calculated in the usual way, using possible adjusting factors such as sales points and competitive comparison data. The technical-requirement weights are also calculated in the conventional way.

Instead of continuing the QFD process at the next part-deployment level, SQD proceeds with mapping the technical requirements to data models and process models. Zultner (1998) refers to usage of entity relationship diagrams (ERD) and data flow diagrams (DFD) to structure the data and process models. Obviously the instructions predate the advent of more object-oriented approaches and the unified modeling language (UML) notation, which are the state-of-art tools currently. The relationships between the requirements, entities, and processes are analyzed in a singe T-style matrix. The entities in the matrix are refined into precise statements of what data are required but not how to implement them. The processes are refined into describing what processing is required but not how to do it.

SQD provides an additional matrix type for mapping the relationships between entities and processes. This can ensure that the diagrams are consistent with each other. Other matrices of similar format may be useful as well. For instance, an object–entity process would support an object-oriented analysis approach. An event–entity process matrix would support an event-oriented analysis approach. However, the current tools—such as UML diagram editors—may accomplish the same task as the matrices in a much more effortless way.

The SQD analysis can be carried out to the subsequent levels. There are some guidelines in the adaptation of the QFD into software: data replaces material, time replaces cost, and process replaces function. SQD—or QFD—has several matrix types that may prove useful in structuring and communicating various issues. For instance, the suitability of new concepts can be analyzed against technical requirements or risks involved with the concepts that can be addressed. The nature of software projects may vary and the SQD approach can be adjusted accordingly.

3.3 PriFo Software QFD

According to Herzwurm, Schockert & Pietsch (2003), the preference-setting and focusing aspects of QFD are more important when determined by means of the house of quality than the deployment by a matrix sequence. That is why they call their approach PriFo—prioritizing and focused—Software QFD. The entire PriFo QFD process is carried by a QFD team, which is put together from departments such as development, quality management, marketing, etc., and customer representatives.

The initial task of the project concerns setting the project's goals, discussing the schedule, cost planning, and putting together the QFD team. The planning phase also includes defining the project's content, identification of the customer groups, and selecting customer representatives.

Zultner's customer deployment can serve as guidelines in identifying and weighting the customer groups.

One of the first QFD team meetings tries to ascertain customer needs as a substitute for supplementing a customer survey. The needs are classified, structured, and weighted in the voice of the customer table using affinity and tree diagrams. Herzwurm et al. (2003) also mention AHP as a viable weighting method. The weighting should be done by as many members of the customer groups as possible and under the overall control of the customer representatives.

In case of a further development of an existing product, the customer representatives evaluate the requirements according to the level of satisfaction fulfillment that the requirements have reached already. The suggested scale ranges from 1, indicating total dissatisfaction, to 5, indicating perfect satisfaction. Competitive evaluation is considered costly, since customers are not also likely to be able to evaluate competitor's products at the requirements level. Therefore, additional customer representatives would normally have to be consulted. The process may end up involving a wide-ranging customer survey.

A software product is not actually identified by its physical characteristics but by its behavior. In software QFD, one has to distinguish between functional characteristics—product functions— and non-functional characteristics—quality elements—of the product. The effort of capturing the voice of the customer results in two quality tables: a software house of quality matrix representing function deployment and a classic house of quality matrix as in the four-phase model. The division of customer requirements into the two tables is done by the QFD team and the developers in particular. The internal team determines the product functions and measurable quality elements in the matrices. Identifying the relationships between product characteristics and customer requirements is ideally done together with the customer representatives. Further elaboration by the internal team leads to a table of the most important product functions and a table of the most important quality elements.

The product characteristics are examined for potential synergy effects and conflicts as in the correlation portion on top of the basic matrix. The most important product functions and the most important quality elements are consolidated into design points and form the basis for setting up a requirements specification as a result of the requirements engineering process.

The PriFo QFD method (Herzwurm et al. 2003) focuses more on recognizing and satisfying customer needs than formalizing and specifying customer requirements.

3.4 Other approaches

The QFD method has many applications in the software industry. Although it does not seem to be very widely used, various sources report that the method has been successfully applied, particularly in the requirement specification phase of software projects. It is probably due to the inherent flexibility of the method that QFD is adapted in a wide spectrum of ways, sharing some common features—including the name software QFD or SQFD— but focusing on slightly different aspects.

3.4.1 Method/1

Accenture applies QFD in their Method/1 software process (Krogstie 1999). The majority of activities associated with software QFD occur during planning and analysis where the project's scope and value are determined. The software QFD process is divided into six main areas:

1. Determine stakeholder types and characteristics. This step resembles Zultner's approach and produces the stakeholder types and characteristics table, and the stakeholder types and project success factors matrix. This phase is necessary for the project team to identify who is important to the project and why.

2. Evaluate stakeholder inputs. The stakeholder input is collected and organized during requirements gathering for the stakeholder input table. The table provides initial input to data, process, or object models, and to functional and non-functional requirements.

3. Define business needs that represent problems and opportunities that the solution could address. The business needs affinity diagram provides an initial view of the business needs hierarchy.

4. Assign business needs to stakeholder types. The project team has to evaluate how important the fulfillment of business needs is to the stakeholders. The level of stakeholder satisfaction provides a means of prioritizing the business needs.

5. Align requirements to needs. The business needs and functional requirements matrix prioritizes the system's functional requirements based on their contribution to the business needs. The same procedure can be used to align non-functional requirements and project requirements. The output of this phase is three matrices for relations between business needs and functional requirements, non-functional requirements, and project requirements. Compared to Zultner's SQF approach, the Method/1 output has one matrix more, since there is no direct correspondent to the matrix containing the project requirements.

6. Managing value. Prioritizing requirements serves as a guide to the downstream activities in the system development process. The priorities and value define where the project team should devote their scarce resources.

In the Method/1, the SQFD process is accompanied by the use of groupware system tools.

3.4.2 Software QFD

Liu (2000) describes a variation of the basic four-phase QFD process applied to software engineering. The SQFD process builds around the house of quality matrix, which is constructed in the requirement analysis phase. Customer requirements represent the voice of the customer and the system technical specification represents the voice of the company—the "whats" and the "hows".

The part-deployment phase of the four-phase QFD process is replaced with a design phase. In the design phase, engineers develop software architecture, module structure, data structures, and user interface according to functional specification and non-functional constraints resulting from the requirement analysis phase. These are analyzed in a matrix representing relations between technical specifications and design characteristics. In Zultner's SQF approach, analyzing this data is actually divided into three separate phases, so compared to that, this phase has fairly low granularity.

The third phase—the process deployment—of the QFD process appears as the implementation phase in SQFD. The design characteristics are mapped to implementation strategy using a matrix. Programming languages and tools are chosen and programs are developed according to the design specification. Zultner's approach does not have a direct correspondent to the matrix used in this phase.

The last QFD phase—the manufacturing deployment—is replaced with the testing phase in SQFD. A matrix representation is used to map implementation strategy to testing strategy. In the testing phase, test plans are developed and testing is conducted to remove defects in the programs.

The presented SQFD approach (Liu 2000) appears to give a fairly ideal and high-level view of the software engineering process. In fact, a practitioner may begin to suspect the utility of the model. There seems to be a slight mismatch between the order of phases in the model and real-life software projects. For instance, the test planning begins typically much earlier in the process than suggested in the model. The model gives instructions to mapping issues on various granularity levels, but does not further elaborate how it should be done.

3.4.3 Software development with Blitz QFD

Zultner (2000) suggests adapting the Blitz QFD approach also to very rapid software development. He characterizes software development as facing the need for speed and quality simultaneously. Early market entry has great significance and the product must satisfy customers. Customer satisfaction comes from the value that the product delivers. Out of the large number of work objects produced during software development, only a few directly or strongly relate to any high-priority customer need. While all the objects are there for a reason, some are more important than others. The process should concentrate on finding essential items, which have a high value in terms of satisfying customers and of essential tasks whose time-management is counterproductive or involves high risk.

The application of the Blitz QFD process begins with a preparation phase. In preparation, the process is tailored with considering questions like

- What does "success" mean in this project?
- Which customer segments are critical to our success?
- If we understand what success is, and who our customers are, where will our software add value to our customers?
- What are the constraints on our project?

From questions like these the generic Blitz QFD (outlined in section 2.2.5 earlier) can be tailored to fit the project precisely.

According to Zultner (2000), the development process must be continuously improved. Blitz QFD offers a framework for becoming more sophisticated at delivering value to customers better and faster. The Blitz QFD can be extended to a more comprehensive QFD process.

Blitz QFD also supports rapid application development (RAD) techniques by focusing on customer satisfaction. While fast feedback, rapid prototyping, iterative development, and other techniques are excellent for mid-course corrections, they work even better when Blitz QFD helps to reduce the number of unnecessary iterations. QFD mechanisms can also help in project management and scheduling.

3.4.4 Distributed QFD

Obtaining customer needs and wants and translating them into optimal product functionality is always a challenge. The task becomes even more challenging when the customers are geographically, culturally, and functionally diverse. Digital's Corporate Telecommunications Software Engineering used groupware techniques supported by the distributed use of QFD to identify product features that meet customer needs (Hrones, Jedrey & Zaaf 1993). The approach was developed to avoid the costs of bringing the widely scattered cross-functional team together in a single location. The practice of running QFD with involvement from multiple physical locations is called distributed QFD (DQFD).

In essence, DQFD was used for defining global requirements. The output of the process is a completed and analyzed house of quality matrix. Differences between this and the conventional QFD approach come from the organization and scheduling of work to multiple sites. Every site has a facilitator. At each site, the DQFD participants are organized into teams connected by means of teleconferencing or videoconferencing equipment. The teams work together under the control of the designed "primary facilitator". The work is scheduled to phases, which may even exploit the time zone differences around the globe.

In the preparation phase, the project is planned and the logistics that will work best are determined. The most appropriate participants are chosen and the team is trained. Various methods can be used to collect customer data. An overview meeting takes place at the end of preparation in order to help participants get to know each other and to share thei common visions and knowledge of the project.

The actual QFD matrix processing is carried out in three days, for instance with team participants in Europe and the USA. In the first session, brainstorming is used to gather customer needs and product features. Because of the time zone difference, the American team continues with affinitizing the needs and sends the results to Europe. In the morning the European team reviews the affinitized needs. Both teams continue together with adjusting the affinitized needs, attaching customer values to needs, and completing the planning matrix. Again, the American team continues with affinitizing product features and sends the results to Europe. The European team reviews the affinitized product features. Teams continue together adjusting affinitized features, completing the correlation matrix, and summarizing the results. By taking the future steps, the business-requirement document is produced.

The work can be arranged differently and the arrangement is likely be different, depending on the exact location of the participants. Hrones et al. (1993) reports cases where DQFD has been used in practice.

3.5 Commonalities in approaches

Although there seems to be no dominant way to adapt QFD to software engineering, the approaches share some common features. The focus is in the early activities of the software development process: identifying customers, i.e., the stakeholders, and requirement definition.

A matrix or tabular approach is used to identify users and customers and to classify them into various stakeholder groups. Groups are analyzed according to their needs and importance. The overall goal is to satisfy customers; this analysis aims at understanding who the customers are.

The requirements are gathered from the stakeholder groups. Requirements are formulated using the basic house of quality matrix approach. The analysis may categorize the requirements into functional—or binary—and non-functional—or adjustable—requirements and even into project requirements.

Lehtola (2003) observed that prioritization of requirements is considered problematic in practice in the software development. Prioritization may mean different issues in different projects and organizations. The term "priority" is rather loosely used in the requirements engineering context, without being very specific as to the various factors of priority. Projects and organizations should specify these factors. The QFD matrix approach offers a way of addressing explicitly the factors like importance ratings, number of complaints, sales points, difficulty of implementation, required effort, and estimates of cost and return.

Although there are suggestions that QFD might also be applied in the successive phases of the software development process, QFD seems to have less value there. This is probably because software is an abstract entity, which is developed, not manufactured. The Pugh concept selection matrices might have some use in architecture design, while relationship matrices could be useful in test development. However, there is a wide variety of software development techniques for successive design and analysis levels. For instance, the use of unified modeling language (UML) techniques is quite common in the software industry. Software developers may consider UML diagrams to have more expressive power than relational QFD matrices in describing the associations between system entities, objects, and behavior.

Chapter 4

Prerequisites of successful application

This chapter discusses the meaning of success and gives an overall analysis of QFD from literature sources. The purpose is to gain an understanding of what QFD actually promises and what is necessary in order to keep those promises.

4.1 Assessing success

4.1.1 Success aspects

In order to assess the success in applying QFD, we should first consider how the success can be observed. There are several reported success stories in applying QFD in real-life projects. For instance, ReVelle et al. (1998) have collected quite a few from various sources. The cases include:

- Customers of Motorola America's Parts Division were 60% more satisfied with their product and pricing information after the system was improved using QFD principles.
- Toyota Auto Body Company in Japan reported a cumulative 61% reduction in startup costs related to the introduction of four different van models over a seven-year period. During the same time period the product development cycle was reduced by one-third.

- The Wiremold Company reduced new product development times by 75% in three years. It was able to introduce over five times more products per year with no increase in salaried personnel. Simultaneously, higher quality products and dramatic sales growth were achieved.

Obviously these cases are all extremes, but they show that there may be high potential payoff in utilizing QFD.

There are many aspects to success in QFD projects. Herzwurm, Ahlemeier, Schockert & Mellis (1998) define the meaning of success simply as reaching the given goals. Goal setting is case-specific and the goals are not always complementary to each other or neutral. Although some aspects of success can be assessed numerically, they might still be problematic to evaluate quantitatively. ReVelle et al. (1998) summarizes some of the benefits of the QFD method. They will be presented here and discussed shortly.

- *QFD will give you a better product, service, or process than you would have achieved otherwise.* This kind of comparison is possible only if the company develops nearly same kind of products over long period of time or in parallel teams, and measures the "goodness" of the products. The claim is hard to verify quantitatively when unique products, services, or processes are developed, because development takes place only once with the same knowledge and conditions.

- *QFD will give you this better outcome faster than will other methods.* Also, this measurement is possible only when the process can be repeated in successive or parallel instances.

- *QFD will typically require fewer resources.* Comparison of resource requirement is possible only if there are comparable projects. Actually, the first project introducing QFD is not a typical case, since introducing, learning, and trying QFD for the first time is likely to require more time and resources than in successive cases.

- *QFD will give definition to the design process, helping the design team to stay focused and effective, giving team members greater ability to see and* *understand how they contribute to the design process as well as how to work with customers and other team members.* These benefits can only be measured by assessing perceived value. However, this kind of internal communication is seen as important to the effectiveness of team members and to the atmosphere of the workplace.

- *QFD will allow for easy management and peer review of design activities as they progress, with graphical representation of the different sets of information driving the design as well as the linkages between information sets.* Although this benefit is hard to measure quantitatively, it seems to be fairly easy to accept. The QFD process builds up to completing the matrices and the progress can be estimated by looking at the completion rate of the matrices. However, the readings may be fairly inaccurate.

- *QFD will leave you very well positioned should you need to improve upon your results for the next-generation product, service, or process.* This kind of effect is hard to measure. An obvious reason for this claim is that when the customer needs and the surrounding constraints and forces are understood, it is easier to increase and use this information in the successive generations.

As we can see, direct quantification of the aspects of success is practically impossible. Success can be defined as the level of accomplishment in the preceding benefits, for instance. The accomplishment level can be measured only indirectly by asking those involved for their perceived level. The success is often a very subjective issue.

4.1.2 Success in software engineering

The QFD method was first developed with the product development process in mind. During this time, the method was extended and applied to other areas. QFD is not seen as a scientific process, but as a quite flexible and easy to adapt tool for analysis and development. It is likely that utilizing QFD in tasks not involving analysis and development will not be very successful.

There are success stories relating to the application of QFD in product, service, and process development. Software engineering differs from traditional product development as characterized in section 3.1. Furthermore, software development also shares some characteristics with service and process development. Software engineering has many existing methods, approaches, tools, and practices for developing the software products. Some of the approaches are carefully tailored for taking the special characteristics of software development into account. QFD is not likely to replace those existing practices, but it may be seen as an accompanying tool for them.

The main application of QFD in software engineering is in the requirements engineering activity. By default, this fits the analysis and development power of QFD quite well. According to Standish Group (1995), user involvement is the most significant success factor in software projects, while a clear statement of requirements is the third significant. Furthermore, lack of user input and incomplete requirements and specifications are two of the most significant factors challenging and impairing software projects. Since all of these factors are related to things to which QFD is supposed to address, success can be assessed by estimating QFD's positive effect on user involvement and on producing clear, complete, and unchanging requirements and specifications. Naturally, the other aspects of success, such as schedule, budget, and quality, still remain significant.

4.2 SWOC analysis

One approach to the possibilities of QFD is to analyze it for its strengths, weaknesses, opportunities, and constraints (SWOC). For instance, ReVelle et al. (1998) present such an analysis. Here the analysis is combined with observations of other sources and some conclusions. In this kind of analysis, classification of issues between strengths and opportunities, and correspondingly between weaknesses and constraints, is not always clear. Sometimes an issue can even be considered as both a strength and a constraint simultaneously.

4.2.1 Strengths

The following issues are addressed as strengths of QFD (ReVelle et al. 1998):

1. Structured (more effective, less waste of resources)
2. Planned up front (less risks, greater knowledge gathered and greater likelihood of accomplishment)
3. Overall cost is less per development program due to
   (a) Fewer changes to be made later because of the extra work during the conceptual stage
   (b) A greater proportion of the changes made occurring in earlier stages when they are easier and cheaper to make
   (c) Reduced development time—achieved as a result of increased efficiency, greater information flow, and more sharing of information in a structured format.
4. Return is greater because
   (a) Market entry is earlier
   (b) Market share is earned more easily.
5. Life cycle costs are lowered because
   (a) The product's entire cycle is dealt with as an integrated whole—from concept to production to customer use to disposal or recycling, thus resulting in a better design
   (b) Better designed product or service resulting in
      i. Greater customer satisfaction
      ii. Fewer returns (on product), fewer complaints (on service and product)
      iii. Fewer warranty claims
      iv. Lower service parts inventory needed
      v. Less maintenance and service needed.

Haag, Raja & Schakade (1996) have collected some benefits of QFD in the software engineering context. These are the strengths:

6. Fosters better attention to customers' perspectives
7. Creates better communication among departments
8. Provides decision justification
9. Quantifies qualitative customer requirements
10. Represents data to facilitate the use of metrics
11. Facilitates cross-checking
12. Avoids the loss of information
13. Reaches consensus of features quicker
14. Can be adapted to various system development life cycle (SDLC) methodologies.

QFD is a structured, planned, and relatively simple approach, which helps the management of the development process and supports team involvement. Simplicity means that the method does not contain complicated mathematics, for instance.

QFD can be seen as an implementation vehicle of total quality management (TQM) (Haag et al. 1996). Because QFD aims at preventing defects early in the development process, it is a proactive and preventive quality tool.

During its progress, the process produces documentation in a structured way. The involvement of a cross-functional team increases the flow of relevant information, while the team, typically residing in the same physical location, makes communications more effective, so the process throughput time is shorter.

4.2.2 Weaknesses

ReVelle et al. (1998) identifies the following issues as weaknesses of QFD:

1. Needs long-term (3–6 months plus) commitment from many different (conflicting?) segments of the organization at both a high and a low level
2. Is a new approach that
    (a) May not have a champion
    (b) Requires additional up-front work (compared to other design approaches)
    (c) Requires the organization to address many QFD tasks that it has never done before and to integrate the results
    (d) Requires teamwork from a group of (often) near strangers who may not (often do not) have team-building skills or experience to be effective
    (e) Is difficult to institutionalize because it is a one-time event in the case of a QFD product or service development project (although some persons on the first project team may serve on later project teams, in which case their QFD knowledge might be deployed).

Cristiano, Liker & White (2001) mention also the following aspect which can be understood as a weakness:

3. QFD may be seen as one of the many "fads" that companies have pursued as quick fixes to complex problems.

Haag et al. (1996) support the view that in software engineering, the QFD model is still in its infancy state. There seem to be no clearly established or standardized methods of applying QFD to software. Companies have to develop their own way of applying QFD; apparently they are not particularly eager to reveal their policies and methods to others. This makes it harder to adapt QFD.

There also seems to be a general perception that the QFD process is costly, difficult, and hard to learn. There are some confirming observations of QFD taking more time and effort when applied for the first time in companies (Karlsson 1997). The difficulty of the method is a very subjective attribute. The method involves manipulating data in a relatively simple matrix format and does not involve complex mathematics, for instance, so the perception of difficulty is very challenging. One can only ponder the possibility that the root cause of this weakness is human prejudices and the relative newness of the method. Because of the relative simplicity of the QFD matrix approach, learning the method should not be too hard. However, QFD material can be found from many different sources and the views are not always quite coherent. Learning can be rather hard if it does not take place in a conducted manner; effort has to be spent on evaluating and finding the most appropriate books and sources—as can be confirmed when collecting material for this work.

In spite of the structured approach of QFD, the way to gather and analyze the raw data—customer needs and wants—seems to remain rather case-specific. An inherent weakness arising from the QFD matrix approach is that it may be very time consuming to analyze and work through a large number of requirements. The number of items to handle or decisions to make grows quadratically.

Compared to other methods in software engineering, QFD may seem to lack expressive power. For instance, the use case diagrams of the unified modeling language (UML) may be considered as being visually more attractive than matrix presentation and thus more expressive. However, the techniques are not mutually exclusive; they can be used for slightly different purposes and hence complement each other.

4.2.3 Opportunities

QFD offers some opportunities. Success of application could also be defined as how well the opportunities can be met and exploited. The following are issues considered as the opportunities of QFD (ReVelle et al. 1998):

1. To reduce development costs in the near term
2. To reduce manufacturing and distribution costs in the midterm
3. To reduce life cycle costs in the long term

4. To be more responsive to the market with a more flexible product or service

5. To deliver higher quality (addressing customers' needs with fewer gaps and missed opportunities for greater competitive impact) in less time (from concept to fully ramped production) at lower cost (initial and lifetime).

Haag et al. (1996) add the following benefit to their list:

6. Reduces product definition interval.

This point closely matches the first point of the list, reducing development costs.

By summing the opinions gathered from various sources, the QFD opportunities are realized by putting quality into the design process. The number of requirement and design changes is reduced, costs are lowered, and productivity increased. The quality and reliability of the resulting product is improved, and, because the work is done in less time, market entry is earlier and the market share increases. As such, the list seems to be very attractive, but probably not unique when considering the marketing material of the various methods, tools, and techniques in general.

4.2.4 Constraints

QFD does not come without constraining issues. ReVelle et al. (1998) list the following:

1. Since a QFD team runs across several "jurisdictions", it can easily be made ineffective by
   (a) Political moves
   (b) Withholding or reducing resources
   (c) Management not making adequate provision for the QFD team members to be relieved of their day-to-day ongoing responsibilities (a de facto reduction in team resources).

2. Because QFD is a sequential process, a blocked step cannot easily be worked around.

When put in the software engineering context, the QFD approach seems to favor the waterfall model of development. Software is seldom developed by applying the pure waterfall approach where each process phase is followed by the next, sequentially. A more common approach follows the iterative model, where the overall development process is divided into sub-processes that themselves are small waterfall processes. This approach provides better feedback, makes progress more visible, and offers better chances for corrective actions. While QFD could also be used in the miniature waterfall processes, there seems to be very little by way of reported experiences of doing so. The problem might be how to miniaturize each QFD step to fit in with the iterative model.

One of the strengths of QFD arises from requiring all team members to meet in one physical location at same time. In the modern development environment, this can be also a constraining issue. Software development sometimes takes place in multiple continents in parallel. With careful arrangements, the requirement to meet in one physical location can be circumvented (Hrones et al. 1993).

The QFD matrix approach does not inherently build hierarchies between requirements. Humans cannot concentrate on a great number of issues at the same time. This results into a finite number of requirements that are possible to handle at one time.

4.3 How to overcome weaknesses and constraints

The QFD method is a human and organizational process with lots of case-specific dependencies. The method has weaknesses and constraints whose effects can be diminished by concentrating on some prerequisite and enabling factors. Various such factors can be found in the literature. This work tries to collect the main prerequisites and enablers and categorize them. The following broad categories are identified: management commitment, organizational recognition, goal definition, organizational learning avidness, training and support, teams and champions, and customer focus.

Because of the human and organizational aspect of applying QFD, an indirect observation is that cultural issues tend to have an effect on the evaluation of QFD application success. For instance, some practitioners and researches seem to feel more comfortable with clear organizational structures and thus emphasize that issue. Others may be more sensitive towards consensual decision-making, individualistic goal attainment, or organizational accomplishment. Apparently this reflects practitioners' and researchers' national and cultural differences (Hofstede 1991).

The prerequisites and enablers are discussed more in depth in the following.

4.3.1 Management commitment

According to ReVelle et al. (1998), management wants and needs the benefits of QFD. Management must understand the QFD process and its strategic position. QFD efforts require cooperation between various functions. Visible leadership at the senior management level supports these processes. Management should also participate in QFD team communication and act upon it, especially on those item that are an impediment to the team's progress.

Beskow, Johansson & Norell (1998) report that some of their interviewees asked for an active management involvement in the actual QFD analysis, to make management's support clear and to make their understanding stronger. Although QFD implementations were initiated by management, people felt that the management lost interest later and that the results became less successful.

According to Cristiano et al. (2001), management support in terms of both empowerment of people and availability of resources has been identified as a key factor in the success of QFD studies. Their own observation is that management support was the largest predictor of the general QFD impacts on general product-process improvements.

4.3.2 Organizational recognition

Once the QFD team is assigned, withdrawing or switching team members during the project causes corresponding reductions in the effectiveness of the team. It also lowers the chances of success of the QFD process. Department managers must understand and formally address this issue (ReVelle et al. 1998).

Herzwurm et al. (1998) emphasize that QFD, being an innovative product development strategy that commits all employees to quality responsibility, has positive effects. QFD should bring structured project organization, which helps in the realization of design objectives.

Cristiano et al. (2001) suggest that QFD seems to be more ingrained in corporate culture in Japan than in the USA. QFD is more positioned in the context of overall total quality control (TQC) effort in Japanese companies, with a stronger emphasis on the overall QFD process. Giving QFD a recognized position in the structure and operation of the organization, and applying QFD in a variety of project phases is indicated as having a positive effect on product innovation.

Beskow et al. (1998) point out that it is equally important to anchor the use of a new method like QFD in the upper levels of the organization as within the product development teams. Management should give appropriate priority to QFD and assign the required resources.

4.3.3 Goal definition

As Herzwurm et al. (1998) define success as reaching goals, the goal setting is essential. The project goals should be written down and they should be passed on to all employees of the project.

Beskow et al. (1998) claim that the planning phase was often overlooked in the companies participating in their research. However, a project should have a clearly defined need and well-defined goals. Representatives from all potential users should have the possibility of discussing and influencing the goal before it is finally formulated. The goal should be concrete and every effort should be put into finding ways of measuring the results against the goals in order to enable an evaluation of the implementation.

Human behavior also has a tendency to resist any change. Goal definition can and should satisfy the information need among the people involved and thus fight against their resistance.

Karlsson (1997) relates his experiences of QFD in sofware requirements analysis, stating that the development project must have a distinct and agreed-upon purpose. The project team was often caught up in discussions about issues such as the relevance for the project of activities performed at each moment. Efforts become more coherent after a team member wrote down the purpose of the project and the statement was refined until it was considered to be both clear and unambiguous.

4.3.4 Organizational learning avidness

Organizational learning avidness is emphasized in two ways. First, the QFD team works for learning and understanding customer needs and wants, which is not always obvious in engineering-oriented development group. Second, the QFD method is still often a new approach in organizations. Learning the method and establishing appropriate practices require collective effort.

In their review Cristiano et al. (2001) mention viewing QFD as an investment in the product and the team as one key success factor. They also observed that in Japan QFD is used and viewed by management as a tool for organizational learning. Applying QFD can transfer product knowledge quickly to novice engineers, making them more effective in less time.

Completing a QFD process is likely to take more time than initially planned (Karlsson 1997). After obtaining basic QFD knowledge from the literature it may not be just that easy and straightforward to complete the house of quality matrix. The organization must be willing to tolerate the learning curve and allow time for a first application.

4.3.5 Training and support

Training and support address the issues related to learning of individuals. The QFD method is often also anew approach to people working in organizations.

Beskow et al. (1998) state that education and training play a big role in anchoring the QFD method in an organization. The team members can gain sufficient knowledge and skills in using the method by right timing and the breadth of their educational program. Education and training help overcome any resistance that rises from a lack of ability.

On his list of prerequisites for applying QFD, Karlsson (1997) suggests that the team should be cross-functional and adequately trained in QFD. Despite the knowledge gained in a two-day QFD course, a cross-functional team is likely to have different views and opinions on the QFD concepts. The views can become more unified by having discussions and using a very small example to try to learn more about the concept of QFD before applying it to large-scale software development projects.

Herzwurm et al. (1998) recommend that in the design of QFD projects a bottom-up introduction strategy through pilot projects and an at least two-day training opportunity should be emphasized. The number of participants undergoing the training should be less than ten persons. Also a project-specific adjustment of QFD should take place using supplementing methods, taken from, for example, the marketing department.

4.3.6 Teams and champions

In their recommendations, Herzwurm et al. (1998) emphasize that the QFD team should be an interdisciplinary team drawn from the spread of departments and containing more than ten persons. Team members should include customer representatives, too. On the other hand, the size of the teams vary but two thirds of the teams have reported to consis of ten people or less (Cristiano et al. 2001). Additionally, a review of case studies indicates a need for strong cross-functional involvement in the QFD project team (Cristiano et al. 2001). Expertise from different functional areas contributes to the decision making process, forcing consideration of downstream issues earlier in the process. Cross-functional teams are considered important also in the implementation phase of the QFD process, where the suggested size of the team is from six to eight persons (Beskow et al. 1998).

Herzwurm et al. (1998) suggest that the team should be led by a project manager who doesn't act at the same time as moderator of group meetings. Several interviewees have pointed out the importance of having a well-trained person—a facilitator—available for support when performing the QFD activities (Beskow et al. 1998). According to a review by Cristiano et al. (2001), most companies report the existence of an internal champion who has been instrumental in the acceptance of QFD. Furthermore, if the usage of QFD was dictated by management, the result of the QFD application seems to be unsuccessful.

Beskow et al. (1998) observe the need for evaluation of the QFD project. Evaluation is easily overlooked and therefore should be planned for at the beginning of the project. An evaluation should give everybody involved a chance to summarize and discuss how bes to reach the goals, the main problems and advantages, the value produced, and further implementation.

4.3.7 Customer focus

The collection of new customer data to determine needs at the start of the new design cycle is identified as a factor having a positive relationship with QFD product and process improvements (Cristiano et al. 2001). The opposite approach is to make use of existing data on past products, such as warranty claims.

Using real or potential customers in the needs and wants connection phase is generally considered important. Herzwurm et al. (1998) emphasize comprehensive surveys of customer requirements and a determination of solutions that contain both functional and non-functional characteristics. They recommend an intensive, direct and indirect customer questioning concerning customer requirements and customer satisfaction, absolute and in comparison to the competition. As the QFD method stipulates, a detailed correlation analysis of the connection between customer requirements and (technical) solutions is also emphasized.

As a prerequisite for applying QFD, Karlsson (1997) states that customers and users should be visible. He derives this notion from experiences in a project in which not explicitly having customers and users turned out to be problematic. He concludes that explicitly having customers and users is mandatory to make full use of the complete QFD process and to gain accurate results.

4.4 Researchers' observations

There seems to be a general consensus about the benefits of applying the QFD method. QFD application techniques vary considerably, while typically, design projects are very case-specific. However, the literature seems to have a fairly unanimous view about the virtues of QFD and the ways to practice it. But is the QFD really a more successful approach than any other, and if so, what are the key elements of the success?

Measuring success is difficult, as discussed in section 4.1 earlier. Assessing quantitative values of QFD issues is problematic. Even if there are actual measurements, organizations may be hesitant to share their knowledge (Haag et al. 1996). During the preparation of this work, quantitative analysis and results were not so easy to find. Luckily, two unique studies were found addressing the interesting issues: comparison between QFD and traditional approaches, and verifying hypotheses on QFD success factors. The results of these studies are discussed below.

4.4.1 QFD and traditional approaches

Haag et al. (1996) have studied QFD usage in software development. They used a combination of open-ended and closed questions for data collection. A total of 37 major software vendor companies were chosen to do the research. Of the participating firms, 16% identified themselves as users of software QFD (SQFD).

Among other interesting issues, Haag et al. (1996) carried out a comparison of results achieved between traditional approaches and SQFD. They composed a set of 12 goals and the companies rated the results achieved on a five-point Likert scale: 1—result not being achieved, 5—result being achieved very well (Table 4.1 on page 64). SQFD achieves significantly higher results than traditional approaches in most of the areas. The only areas where results are lower are in systems developed within budget and systems developed on time, but the difference made to the results of traditional approaches is not very significant.

Table 4.1: Comparison of results achieved between traditional approaches and SQFD (Haag et al. 1996)

Results achieved	Mean traditional rating	Mean SQFD rating
Communication satisfactory with technical personnel	3.7	4.09
Communication satisfactory with users	3.6	4.06
User requirements met	3.6	4.00
Communication satisfactory with management	3.4	3.88
Systems developed within budget	3.4	3.26
Systems easy to maintain	3.4	3.42
Systems developed on time	3.3	3.18
Systems relatively error-free	3.3	3.95
Systems easy to modify	3.3	3.58
Programming time reduced	3.2	3.70
Testing time reduced	3.0	3.29
Documentation consistent and complete	2.7	3.87

Haag et al. (1996) conclude the comparison by stating that the utilization of SQFD improves results achieved in most areas associated with the system development process.

4.4.2 QFD success factors

Cristiano et al. (2001) have tried to determine and analyze the key factors in the successful application of QFD. Their study covered more than 400 companies in the United States and Japan using QFD. Based on a literature review, the researchers formulated ten hypotheses about QFD usage characteristics (Table 4.2 on page 66). Each hypothesis captures a suggested factor, i.e., an independent variable, whose effect on product-process improvements, product innovation, and reduced lead-time were assessed. The regression analysis contained some other control variables also.

The data collection was conducted using self-administered mail surveys. The respondents were asked to assess the QFD activities on specific scales, indicating an objective, quantifiable measure of the process or outcomes. When this was not possible, a generic five-point scale was used.

The hypotheses were classified into three high-level categories: organizational characteristics, data sources, and QFD tools and techniques. The results of the analysis provide some interesting insights into the factors that lead to successful application of QFD. They indicate that management support has a significant, positive relationship with QFD product and process improvements. Likewise, the collection of new customer data to determine needs at the start of the new design cycle has a relationship with product innovation and a weaker, less significant, relationship with reduced lead-time. Also, internal motivation has a less significant, positive, relationship with reduced lead-time. All other analyzed factors have insignificant or only marginally significant relationships with outcomes.

Management support and the use of new customer data were associated with larger positive impacts of the QFD study. The specific QFD tools used did not matter. The researchers reemphasized that the process of designing and manufacturing a product reflects the organizational structure and culture. Specific tools and techniques like QFD are just tools. Management fundamentals like management support, sufficient resources, and customer orientation in design seem to matter most. When QFD helps to accomplish this, it can be a powerful tool.

4.5 Practitioners' experiences in software engineering

Qualitative research and experience-based reports on applying the method in various areas seem to be more common in the QFD literature than

Table 4.2: Proposed factors in successful application of QFD (Cristiano et al. 2001); H2 and H5 are significant

Hypothesis	
Organizational characteristics	
H1	Internal motivation for QFD use has a positive association with the success of QFD projects.
H2	Management support for QFD has a positive association with the success of the QFD study.
H3	The extent of cross-functional involvement in QFD studies has a positive association with the success of QFD projects.
Data sources	
H4	The use of existing customer information sources for the QFD study has a positive association with the success of the QFD study.
H5	The use of customer information collected specifically for the QFD study has a positive association with the success of the QFD study.
QFD tools and techniques	
H6	The use of structure techniques for analyzing customer information has a positive association with the success of the QFD study.
H7	The use of the product planning (phase I) matrix of the QFD methodology has a positive association with the acceptance of the product.
H8	The use of advanced phases of quality deployment of the QFD methodology has a positive association with a reduction of lead-time and manufacturability of the product.
H9	The use of the relationships consistent with the comprehensive model of the QFD methodology has a positive association with the success of the QFD study.
H10	The integration of analytical tools in QFD has a positive association with the success of the QFD study.

quantitative research. The value of such research should not be belittled by any means: it can serve as a precious source of information for those new to QFD and those applying QFD in their business. One can always try to learn from the experiences of others, while a vision of the capabilities of the method helps in setting the expectations to the appropriate level.

A particular area of interest in this work is that of applying QFD in software engineering. In the following, experiences and conclusions from studies of Karlsson (1997) and Brandt (1996) are discussed.

4.5.1 Positive experiences

Applying QFD can give a better focus on customers and users (Karlsson 1997). There is often a degree of technician dominance in developing high-technology products. To a certain extent, the driver is the creativity and knowledge of the developers. Applying QFD in their project, the cross-functional team kept asking "are we now actively contributing to customer and user satisfaction or not?" This gave direction to efforts and the approach sped-up the work, since the discussions tended to be much more effective.

Karlsson (1997) states that an activity emphasized in QFD more than in many other development methods is that requirements must be accurately prioritized. The study indicates that the importance of requirements tends to vary by orders of magnitude, so setting the priorities is important. In traditional QFD the customers and users prioritize the requirements on an ordinal scale typically ranging from 1 to 5. This tends to be problematic since there is no clear distinction between the numbers. For instance, separating priority 4 from priority 5 is not always obvious or repeatable. Brandt (1996) even used an abridged scale from 3 to 5 in his work. The QFD's approach towards prioritizing customer and user requirements is beneficial, but Karlsson (1997) suggests that QFD could be complemented by the use of the analytic hierarchy process (AHP) which is more accurate and efficient for prioritizing than the use of an ordinal scale.

The QFD view of requirements has only a slight relationship to that of traditional software development methods (Karlsson 1997). QFD makes no distinction between functional and non-functional requirements, but rather forces the cross-functional team to state all requirements in a measurable and verifiable manner. The fulfillment of non-functional requirements, or adjustable requirements as Hurri (2000) terms them, is relative, while often the non-functional requirements are put aside, since their representation is not always straightforward. The QFD process forces the cross functional team to develop exact targets for factors such as usability or reliability that must be met by the software system.

Brandt (1996) found the use of the A3 matrix, i.e. the matrix for finding correlations between technical requirements, very useful in finding more accurate definitions of the technical requirements. The A3 matrix is an alternative representation of the "roof" of the house of quality—or the A1 matrix. Developing technical requirements for an immaterial product such as software may be rather difficult. It might be beneficial to complete the A3 matrix before completing the A1 matrix, because it may help to develop the requirements in a more iterative manner (Brandt 1996).

According to Brandt (1996), determining the correlations between technical requirements and customer needs is smooth and actually the most fruitful section in the whole process. The cross-functional team end up in discussions and new ideas are introduced. Some results of this phase cannot be seen in the matrices. The outcome is a higher quality product achieved by the improved cross-functional cooperation required in the QFD process.

Brandt (1996) gives credit to the better documentation and traceability, and to the additional systemacy which QFD seems to offer to the software development.

4.5.2 Issues not fully supported

If the level of abstraction differs among the requirements, there can be problems selecting the correct set of requirements for implementation, and for cost-estimating the system (Karlsson 1997). Developers need guidelines to help them specify the requirements at the right level of detail, and a metric to ensure that the requirements are stated at the same level of detail. Karlsson (1997) considers it a weakness that QFD offers neither such guidelines nor a method for verification. The same problem reoccurs also in the context of control characteristics.

Brandt (1996) arranged customer needs in a hierarchical structure that used higher levels. This produced some difficulties, since some misunderstandings happened occasionally. The situation can be improved by keeping the hierarchy tree diagram constantly available in the meetings.

Sometimes the software requirements have temporal relationships to other requirements. In his study, Karlsson (1997) observed that the cross-functional team did not find a straightforward way of expressing such temporal relationships in using QFD. Because there is no obvious way to express temporal relationships in the house of quality matrix, complementary techniques must be used.

The QFD concept involves different people evaluating existing systems in order to optimize the one under development. When developing new products this may be problematic. If there are no current systems in existence, competitive assessments cannot be made, and thus important inputs to the QFD process disappear (Karlsson 1997).

Karlsson (1997) found it tricky to manage functional requirements using QFD. It is not obvious how to find adequate measurements for a functional requirement such as “the system must be able to print out invoices.” A functional requirement—or a binary requirement, as termed by Hurri (2000)—is either fulfilled or not. The key role of control characteristics disappears in the context of functional requirements.

4.5.3 Other issues

Brandt (1996) observed that the collection of customer needs by interviewing the current and potential system users took a relatively long time and required lots of work. The laborious nature of this is emphasized in a pilot project. The commitment of the group and the atmosphere for customer-oriented development are very important.

Karlsson (1997) believes that software developers tend to work in terms of what they are familiar with and of what they have a deep understanding.

The requirements that the team had a thorough knowledge of were stated in great detail, and the requirements the team were less knowledgeable about were stated at a much higher level of abstraction. The level of abstraction can therefore serve as a warning flag of inadequate knowledge to software managers and developers.

It is often recommended that the house of quality matrix should not be larger than 30 by 30 relationships. Karlsson (1997) observed that the cross-functional team adapted the level of abstraction of the requirements so that they could fit into a 30 by 30 matrix. This caused some re-work, because while the number of requirements increased, they had to be reformulated and the abstraction level had to be altered. One cannot determine the size of the relationship-matrix beforehand and then adapt the level of abstraction of the requirements accordingly.

Lastly, Karlsson (1997) reports that using English as the language of documentation is problematic when the cross-functional team members are not native English speakers. The role of the secretary became more important than expected, because the secretary had occasionally to memorize and translate the issues and intentions written down earlier in the project.

Chapter 5

QFD in practice

This chapter presents direct observations from QFD application cases. Practitioners were interviewed for their experiences and findings from the cases are summarized.

5.1 Background of the research

In this work, the purpose of the case-based analysis is to find out how the QFD techniques are applied in practice. This is also important when keeping the partner companies of CORE research project in mind. The analysis complement the studies found in the literature, because most of the reported cases have taken place in application areas, organizational structures, and cultures deviating from those that are present in the CORE partner companies.

Because the role of the case-based analysis was considered to be accompanying the literature sources, it was felt that the scope of the study need not necessarily be very broad. The sample was chosen from one CORE research project partner company, which was known to have experiences in utilizing QFD. Some of the QFD practitioners had been found in the context of other research in the CORE project. Others were found by asking these persons if they knew any other practitioners. The process was initiated within the CORE project; formal enquiries, requests, or channels were not used. All candidates were not willing to participate in the research. In the background, there were reasons like temporally distant or only superficial experiences on the subject.

For a small number of cases—four, in total—interviewing was considered the most practical approach to data collection. A small set of interviews is not supposed to produce statistically significant results. On the other hand, a small set of personal interviews can reveal some of the effects of random events and personal characteristics on the outcomes. This does have some value, since although books and literature sources describe the QFD techniques, examples and the experiences of others help practitioners to utilize those techniques.

The QFD method was studied at the same time as the interviews took place. During the first interview in May 2003, preparation of this work was in the early phases and the interviewer had only vague knowledge about QFD. During the last three interviews in October 2003, most parts of the literature review were already completed, which had certain effects on the question-setting and focusing in the later interviews.

The interviewee candidates were approached with a brief description of the study and its background, a request for an interview, and a skeletal list of questions. The question list presented to the candidates was kept short in order to keep the threshold for participation low. The same list of questions was used in all interviews with accompanying questions and some variation in focus:

- *In which kind of a project has QFD been applied?* This question category aimed at figuring out the practitioners' role in the project, position in the organization, the goal of the project, and the application area.

- *Which QFD approach has been applied?* In addition to the application approach, this question was used to find out the respondent's sources of QFD knowledge, awareness, and experience.

- *How was QFD applied?* The purpose was to find out the motivation behind the application, application details, the steps taken, and the steps left untaken. How thorough was the application and which phases were covered?

- *What kind of experiences resulted?* Was the application considered successful or not, and in which dimensions? Were there any ill-effects or contentious issues? Is the respondent satisfied with the method?
- *Are there any ideas for applying QFD in the future?* This question category aimed at finding out whether the respondent was willing to utilize QFD in future projects also, whether there were plans for a wider application of the QFD techniques in the organization, whether there was anything that the respondent would do in some other way in other projects, and whether there were any guidelines she or he would give to others?
- *Are there any ideas for applying other methods?* Are any alternative approaches considered or already used?

The interviews were carried out in a semi-structured manner. It was pointed out to the respondents that the interview was confidential. Each interviewing session took about 1 to 1.5 hours. Notes were taken in the sessions and the sessions were recorded for later analysis with the interviewees' permission. Because the interviews were successive, the interviewer's knowledge was cumulative, and some variation of focus was allowed; the case-based analysis has an iterative nature.

5.2 Cases

The case studies were carried out in a Finnish company, the Vaisala Group. The following excerpt characterizes the company (Vaisala Group 2003):

> Vaisala develops, manufactures and markets products and services for environmental and industrial measurement. Vaisala's markets are global. The goal is to provide a basis for a better quality of life, environmental protection, safety, efficiency and cost savings.

> The major customer groups are national meteorological services, aviation authorities, defense forces, road and rail authorities, land and water resource management agencies, research institutes, electric power utilities and industry worldwide.
>
> Vaisala's competitiveness is based on product leadership in environmental measurement and related industrial applications. The Vaisala Group is a global market leader in sounding systems, aviation weather, road weather, wind profilers, thunderstorm systems and information, as well as in professional equipment for measuring relative humidity and barometric pressure.
>
> The parent company Vaisala Oyj, domiciled in Vantaa, is listed on the Helsinki Exchanges in Finland. The Vaisala Group has offices and operations in Finland, Northern America, France, United Kingdom, Germany, China, Sweden, Malaysia, Japan and Australia.

Year 2002 key figures	
Net sales, EUR million	196.2
New orders, EUR million	213.3
Profit before extraordinary items, EUR million	21.5
Solvency ratio, %	83.9
Return on investment, %	15.4
Personnel 31 December	1213

Although all of the cases involved the development of software or selection of software tools in some way, none of the projects were actually software development projects. However, this has more implications on drawing conclusions from the QFD implementation than all of the surrounding issues.

To preserve the anonymity of the interviewees and other persons inside and outside the company, they are all identified with unique capital letters and addressed with the pronoun "he" in the following case descriptions. Products and systems are described only to the extent necessary to expose the problem area and project dynamics; otherwise they are left unidentified. The cases are presented in the temporal order of the interviews.

5.2.1 Gas detector

An organizational unit in Vaisala is making sensors for analyzing air and gases. The unit is successful in mastering the technology and it is also continuously searching for new applications of the technology because of new business opportunities. One realized opportunity was to apply the existing know-how and develop a detector for gas *Z*, for which Vaisala did not have existing products. The idea was not initiated by any particular customer request. Instead, it resulted more from a technology-push type of approach.

A is working as a product line manager in the organizational unit. He represents the marketing point of view although he has training in chemical engineering. A product line manager looks after the products and product lines and there are separate application managers dedicated to on applying the products in various areas and customer segments.

The gas *Z* is used in a variety of application areas. The organizational unit had carried out marketing research as a thesis project. Based on this earlier research, a promising industry segment was chosen. The segmentation was necessary, because the needs in various application areas differ so much that it is extremely difficult to satisfy them with a single type of a product. From then on, the work concentrated on finding out the customers' needs in the chosen segment.

A did not have earlier experience using the QFD techniques. In the company there was a person *E* who is known for his familiarity with different methods and interest in developing practices. *A* got in touch with *E*, who provided material and knowledge about QFD. *A* became convinced that QFD could be a suitable method for determining the customer needs for the new product. *A* studied QFD from the book written by Day (1993). *A* considers himself a practitioner; it is unlikely that he would have obtained QFD just by reading a book; a case and some support were necessary. *E* acted as a facilitator, provided help and guidance, and supplied the project with a commercial software tool for QFD analysis. *A* received training in QFD in an external training company only after the project was actually completed.

A and the developers had an initial vision for the new product. The earlier marketing research was one cornerstone of that vision. The intended customer segment was chosen as having the most commercial potential. The goal was to settle the vision of the new product and find out the customer needs in the chosen segment. There were detectors for gas *Z* on the market, although Vaisala did not have any. The detectors were implemented with various technologies, each of which having advantages and drawbacks. None of the technologies was superior or in a dominant position. Vaisala's approach was totally new; the product would be the first commercial product utilizing the particular technology. There was not a comparable product in the same category.

A prepared a questionnaire for gathering the customer needs and their importance ratings. The purpose was to figure out what things the customers appreciate, what is available on the markets, and how satisfied they are with the current offerings. The questionnaire left some details open on purpose. The idea was to find out the needs and their evaluation, not an evaluation of the solutions. Constructing the questionnaire was considered relatively easy, but the lack of existing products caused some difficulties. There were two categories of questions: general issues and measuring-technology related. The questionnaire contained approximately 20 questions. Some of the questions were in open format for gathering some background information for possible future development.

A emphasizes the need for testing the questionnaire, because it may contain language and issues which are hard to understand even by the people in the neighboring unit, not to mention the customers. This questionnaire was tested internally and with one customer. The observation was that the question formulation might be such that it was better not to mail the questionnaire to the customers. Instead, guided interviews with the questionnaire were carried out.

A had already completed a more general interviews about the application area. The coverage had been over 50 people. The interviews with the QFD style questionnaire were targeted at 20–30 people who were not necessarily the same as in the general interviews. The customer representatives were chosen as knowledgeable and influential opinion leaders in the application area. The interviews were carried out in international conferences. Each interviewing session took about 15 minutes. Approaching the interviewees personally and emphasizing the intention to undertake systematic development for the customer's own benefit was considered to lower the answering threshold.

The results were analyzed with the development team. The whole team with representatives from, for instance, electronics engineering, mechanical engineering, software engineering, and project management participated. The team was not particularly trained in QFD techniques, but *A* briefed the team and guided the sessions. The team discovered important requirements and the technical solutions for filling the requirements by cause–effect analysis. For instance, the product should have had a changeable sensor, but because of the sensor's small size and because detector was to be used in a relatively harsh environment, it was better to have the whole probe interchangeable. Also, it was found justifiable to have two product variants, a basic model and a model with connection utilities to external systems.

The application of the QFD was restricted to the initial or house of quality phase. The experiences were positive. In this case, the product development cycle took about two years from idea to market. *A* believes that QFD can help in shortening the development cycle; the costs are lower, because false prototypes are not being built. With QFD the development team established a shared vision of the product and the perspective was wider. The analysis supported the ideas of what the product should be.

Utilization and application of the method depends on the individuals, for whom it is important to get help and support. The first time takes more time and effort, but success in the first time is important. If the first time fails, it is possible that the method is never tried again. Instead of training ten people to QFD it might be more favorable to hire one experienced to guarantee success. Willingness and a positive attitude are important; people with negative experiences are a risk.

A considers the segment selection important. If the segment is too wide, the value of the QFD surveys stay close to zero. Segment selection helps in focusing the efforts; expansion to the new segments can come later. Difficulties with the questionnaires arise when trying to ensure that the team speaks the same language as the customer. For instance, in this case, one question was observed to be poorly formulated and was later skipped in the interviews. The questionnaire may prove valuable by giving marketing arguments and by better positioning the product, too.

A has plans to utilize QFD in successive projects, some sketches are already drafted, but they need further working. According to *A*, QFD has been also used elsewhere in the company. He has not considered other techniques or decision-support tools and is not aware of their being used in the company.

5.2.2 IT platform

The need for upgrading Vaisala's enterprise resource planning (ERP) system gave rise to an internal development project. In addition to the upgrading, this was seen as an opportunity to unify and standardize the information technology (IT) platforms corporate-wide. The strategic objective of the project was formulated in the management's own language at a relatively high level of abstraction. The purpose of applying QFD was to determine the operative goals for developing the IT platform.

B was in charge of the project. He is the project delivery process owner and characterizes himself as a process developer. *B* has earlier experience in QFD, dating back to the time when he was working in another company. He heard about the method many years ago and actively searched for more information. He has used the book written by Day (1993) and he rates the book very highly and as very readable.

The QFD questionnaire team members came from various functions of the company where the members had different roles. The team composed a list of customer requirements based on their own experiences. The formulation of the requirements, or questions, was chosen in such way that they would imply what functionality is wanted, not which system or product is preferred. The focus was on new functionality; requirements for existing features were left outside the scope. Developing the questions was considered fairly easy; the standpoint of each member was the process he was dealing with. Finally, the questions were grouped by the processes in a questionnaire. There were several groups; the number of questions within each group ranged from about ten to fifty; the total number of questions was over hundred.

The questionnaire form was implemented as a spreadsheet document. It was sent to a representative set of respondents from all departments of the corporation covering all processes and company premises. The purpose was to elicit evaluations and importance ratings for requirements and to see if the perceptions differed by processes, premises, or organizational units. The respondents were given an option to add missing requirements, but only a few new requirements were received. The return rate, coverage, and the results of the questionnaire were considered very good. Because questions were short, there was a short explanation attached to each question to avoid misunderstandings. There were only a few answers indicating that the respondent had not understood a question.

The team formed to fill in the technical portion of the QFD matrix consisted of IT specialists and process owners. The IT specialists knew the IT system solutions and products and the process owners represented the management. The purpose was to come up with a list of current and other potential tools and solutions that were known to have the necessary features. For very large systems, distinct subsets or modules were considered as candidates on the list, too.

After the technical solutions—software products, tools, and systems—were identified, the team began analyzing the relationships between the solutions and the requirements. The task was surprisingly laborious and took a long time. The beginning was very troublesome, with endless discussions and a lot of debate. After a few difficult sessions, the work started to progress; the members began to reach compromises and left out irrelevant details. Despite the troubles, *B* considers the analysis and the process fruitful in the end.

The evaluation and importance ratings were indicated on a scale from 1 to 5. The lowest value of the scale was practically unused, which is probably a result of focusing on the new features. The evaluation ratings differ greatly between processes or other classifying factors, but the importance rating patterns correlated better with those factors. The relationship scale used factor values 0, 1, 3, and 9. An observation was that this scale was quite steep, emphasizing the differences and dominating the calculations.

A steering group carried out a concept selection based on the results of the analysis. When the analysis results were presented, each process was examined for the tools covering the important requirements of the process. Sometimes some requirements were left uncovered, which was not a problem with less important requirements. Sometimes a requirement could be covered by an additional feature to an already chosen solution. The analysis revealed some uncovered functionality which initiated a process for acquiring a set of entirely new tools. It also revealed some emerging needs, which have been addressed in successive projects.

The QFD analysis created a shared view of the necessary components of the corporate-wide IT platform. The analysis gives justification to decisions and provides traceability to some extent, too. The IT manager in charge of the consequent implementation project appreciates the thorough assessment work and has a favorable attitude towards the QFD method. QFD has been utilized in some other tool selection projects, too.

A dedicated software product—QFD2000—was used in the analysis. The spreadsheet application was used in the implementation of the questionnaire because the QFD software implemented the necessary import functionality for the data; otherwise the QFD software would have had to be installed everywhere, or the questionnaire would hav had to be implemented in paper form. The particular QFD tool version was considered a bit archaic and it crashed occasionally. However, the tool was capable of handling the material and produced the necessary reports.

The teams applying the method did not participate in any formal training in QFD. *B* explained the necessary principles as the work progressed.

He thought that the QFD principles were easy to explain, the most difficult thing in the application being to avoid mixing solutions with requirements.

B considers the application of QFD in this project successful. QFD helped to make decisions quite quickly, although there was quite a lot of work. One problem was the growth of the matrices but a proper grouping of items helped their management. It would have been possible to carry out a more thorough assessment of the customer requirements, but that would have resulted in too long a questionnaire including evaluations of the current solutions. Laborious direct interviewing would have probably given more coverage, but there was also the risk of getting biased, interviewer-dependent results.

B holds the view that sufficient weight should be put on the concept design, analysis, and the selection in all development work. This would save time, avoid re-work, and avoid the problems resulting from fixating to concepts too early. *B* considers it difficult to introduce and utilize the QFD method in a company. The task requires a carrier of the idea; someone who is enthusiastic, relies on the technique, and has the persistence required to motivate and carry on through the troublesome phases. In this project, it helped a lot that the carrier of the idea was the leader of the project at the same time.

5.2.3 Multi-meter

Vaisala follows a strategy that positions the company in the segment of professional-level products; it does not envisage itself as a bulk manufacturer. The offerings in the professional-level product segment yield high margins but incur high financial risk. Generation after generation, the products must continuously be re-introduced with new innovations.

During a normal idea screening process, an organizational unit in Vaisala studied an opportunity to introduce a totally new product into markets new to the company. The planned product would be an innovative gauge device capable of measuring multiple meteorological quantities.

The development of a business plan and customer specifications involved a business unit manager and a product development manager, *C*.

In the initial analysis phase, a magnitude of a hundred possible customer segments was identified for the product. The scope was restricted to a couple of promising application areas.

C has an engineering background. Having contacts with the facilitator, *E*, who was the facilitator in the gas detector (section 5.2.1) project, too, *C* began considering QFD as a tool for composing the customer specifications. *E* was *C*'s main source of QFD knowledge, but *A*, from the gas detector project carried out earlier, became a contributor, too. *C* neither had earlier experience on QFD nor had he studied the QFD literature in depth.

C carried out the preparation of the customer questionnaire. He organized workgroups from different units and functions for formulating the questions. In the workgroups, there were people from projects developing similar types of products. *A*'s earlier questionnaire helped, and both *A* and *E* provided support. The workgroups utilized brainstorming and conventional meeting techniques; the resulting questionnaires were tested within the company. The total number of participants was a couple of dozens and the work took a relatively long period of time.

Construction of the questionnaire was a major effort. *C* emphasized the importance of the task. If the formulation of questions was bad, the analysis would suffer from the garbage-in–garbage-out effect; the results would be worthless.

The final questionnaire consisted of about 25 questions for capturing the customers' importance ratings and competitive evaluations, and half-a-dozen accompanying questions for assessing customers' views in technical and pricing issues. The questionnaire was delivered personally or by e-mail to customers in the intended segments. The pattern for the contact was the same in both cases: first the company and the people asking the questions were introduced, then there was an overall presentation of the planned product concept, and finally a request to answer the questionnaire. The result was a total number of about 50–60 completions of the questionnaire. Some of the customers were later also involved in the validation work of the customer and the technical requirements.

The questionnaire was delivered as a spreadsheet document. The answers were imported to a QFD software tool. The software is used for analysis, plotting out the results, and creating reports. A core team of about 4–5 people was in charge of the construction of the technical requirements, although several people from different functional units were involved. The customer requiremens were transferred to the technical requirements using the matrix approach and the work on the technical specifications continued.

QFD was not applied in the later phases, but the results of the analysis are used in designing marketing messages for the product. According to *C*, application of QFD has been successful in this project, although the product is not yet in the markets. The results have been valuable in justifying the design decisions, because the data is based on direct customer input. Likewise, the data has been used in communication with the management of the organizational unit.

C considers the scales for the importance, evaluation, and relationship ratings feasible. He regards the resulting weight values only as numbers giving direction, not as being imperative; their role is to help decision making. According to *C*, applying QFD was fairly simple. One should just hold onto the essentials and apply QFD lightly. *C* emphasizes the pragmatic approach and considers QFD very useful, particularly for developing new products. For the development of product derivatives, the overall cost–benefit ratio may be different.

C does not give any special importance on the management's role. The process descriptions of the company do not specify a method for drawing-up the customer specifications, but there are plans to suggest QFD for the task. Management does not necessarily know the method, although they have been impressed by the results. QFD is gaining some familiarity in the company. For instance, there are plans for analyzing the results of company-wide customer satisfaction surveys using QFD. *C* has not considered any other method for decision support, neither does he know about applying any other method within the company.

In the customer questionnaire in this case, the customers were asked to give a rating as to how well the current products perform in the field today. That is conventional, but the usage of the term "importance" is slightly unusual. The customers were asked to rate the importance to them of how well something should be supported. Usually the rating is supposed to indicate how significant fulfilling a need is to their satisfaction. In this questionnaire, it is as if the importance ratings were used in place of a planned target value.

5.2.4 Sensor

Vaisala offers a wide repertoire of devices for the surface weather measurements. The development of these products does not cease; the models must be constantly renewed to keep pace with the competition and to exploit new innovations. The sensor project was initiated for introducing a new generation of product family. The goal was to develop the first sensor device model in the product family. The device measures one quantity in surface weather.

D was assigned as the project manager for the sensor project. The company had existing products in this area. Of course, *D* could utilize the knowledge about those products, but he felt that the knowledge is often based on indirect sources and the intuition of engineers. *D* thought about ways to utilize more direct customer feedback in the product development. His manager told him that some method had been applied in two development projects recently. That is how *D* got into contact with *B* in the IT platform case (section 5.2.2) and *C* in the multi-meter case (section 5.2.3), and they reported that the method was QFD. *B* gave a short presentation of the method and handed *D* a book about the topic—the book written by Day (1993). *D* read the book, kept his own case in mind, and made himself familiar with the method.

The application of QFD began with brainstorming sessions and meetings about the questions that customers would consider important with respect to the product. The team thinking about the requirements consisted of *D*, product manager, and two or so people from product lines and marketing. The final formulation of questions into a questionnaire was carried out by *D* and the product manager. The questionnaire containing about 40 questions was tested with the help of a small number of people from the company and one customer.

The survey was carried out using the contact networks of the sales and technical support. The goal was to get a reasonably wide and representative sample of customers so that the answers could be collected quickly. It was not considered important to get a statistically ideal coverage. The chosen customers were contacted and with their permission the spreadsheet questionnaire was sent to them using e-mail. The customers answered the questionnaire and sent the results back. The sample size was two dozen.

The results were imported to the QFD2000 analysis software. According to *D*, there were minor problems in the data transfer, while the version of software suffered from occasional crashes. Anyway, the tool eventually performed the tasks it was supposed to do.

After importing the results into the analysis software, *D* determined the technical requirements by himself. By following the instructions of the book, he tried to maintain the appropriate abstraction-level of the requirements. *D* carried out the relationship analysis between the customer requirements and the technical requirements together with individual colleagues.

The resulting technical requirements were transferred into input data for the next level matrix for concept selection. There was a team of about ten people developing various technical solutions and approaches for implementing the product. In this work, the ideas, their architectures and essential technical solutions were sketched and brought to a level such that the comparison of the concepts was possible. Interesting factors in the comparisons were estimates of the manufacturing costs and some technical implications of design choices. The development of the concepts was quite informal.

When the concept choice set was considered complete, a concept selection session was arranged. A little over ten people were involved in the session, which took two and a half days. The session served also as a kind of a kick-off meeting for the actual development project. In the session, the concepts were evaluated and the one considered most appropriate was selected for the final implementation. The session really created and communicated a shared vision of the final product; the group understood the goals, the design decisions made, and the justification of the decisions.

D's conception is that the team has come up with good decisions. QFD has been applied in a small scale in some architectural issues in the later phases of the development, too. The product contains software, but *D* had not been thinking about how to apply QFD in software development. However, he could imagine the use of QFD for selection of software architectural concepts.

The final value of utilizing QFD in development cannot be assessed yet, because the development is still going on and it is impossible to get customer feedback. The QFD analysis from studying the method to the concept selection took about two and a half months of calendar time, but the load level was not 100%. *D* thinks that using some other approach would have taken more time, because it would have been necessary to open the already closed issues. Now they have managed to make good decisions; they have not had to make changes. QFD has helped in communication and the justifications are visible. He believes that the decisions benefit the customer and that it is unlikely that solutions led by intuition, for instance, would have been better at all.

D had no earlier experiences in QFD. In the beginning, he was a little doubtful, as to whether all this effort was worthwhile. However, the need for customer information was evident and QFD was a suitable tool for gathering it. *C* from another project gave support and reported that the life as a manager becomes easier when there are facts and justification for decision making.

D does not exclude the possibility of using some other techniques in the future projects, although he is currently unaware of any other method. In the next application he would probably put more effort into limiting the number of customer requirements, because a large number of them yield a large number of technical requirements. During this project, he already had to drop some less important requirements. *D* considers the method simple and easy to learn. The mechanisms are easy to understand and adapt, but the application of them took some thinking and training. He rates the book very highly.

According to *D*, the numeric scales used in the analysis were feasible.

He says that the purpose of the number is to give some importance-ordering and provide supporting information. As such they are sufficient; there are no absolute best solutions in product development.

D thinks that the use of the method is spreading in the company. So far it is not a standard operation and it might even be beneficial to allow some time for formulating of the practices and finding the suitable application areas. According to *D*, the management neither recommended the method nor resisted the use of it. Management support came in the form of trust. The management did give positive feedback from the resulting requirements, which were specified to a high standard.

5.3 Findings

QFD has been used in Vaisala for various purposes. Although all the cases described in this work involve dealing with or developing software, the projects were not software development projects. The application areas range from product development and implementation concept selection to software tool selection. Despite their varying scopes, the cases share some common characteristics.

The analysis of the cases was carried out by classifying the interviewees' answers and comments with respect to the topics they address. The topic categories were the same as to which the questions were classified. The material was then analyzed by considering answers and comments within each category for similarities and differences between the cases. The similarities that were found were candidates for generalizations.

In each case, QFD has been a new approach for the project group. The team leaders had come to adopt a favorable and enthusiastic attitude towards QFD. Only one of them had earlier experience of the method. The other team leaders were supported by practitioners from earlier QFD projects or an otherwise capable facilitator. The supporters helped the team leaders to get acquainted with the techniques; they also used the book written by Day (1993) as an information source. The team members did not get any classroom training in QFD either. Instead, the team leaders and facilitators explained the goals and procedures before each phase of work.

The method was considered simple to learn. Some team members suffered from fatigue and considered the techniques laborious to apply. The learning took place at a very practical level, interleaved with application of the method in practice. Software tools were utilized in gathering the information and analyzing the results.

The customer questionnaire had been composed carefully. Extra care had been taken to ensure that the customers understood the questions in the intended way. The importance ratings have been used rather unusually. At least in one project, the importance was seen as the respondent's perception of how well something should be supported. In other words, the importance rating was used as a customer-specified target value, not as the customer's perception of the importance itself. The interviewee in at least one other project talked about analyzing the gap between the importance and the current values, so obviously this definition of importance has been used systematically. The effect of this matter may be only minor; QFD techniques have provided results, which support intuition; it is not just the plain numbers that were considered in the decision making.

In the cases described, the basic set of ratings and factors were used in the QFD matrix processing. Additional factors like marketing argumentation points or estimations of complexity were not used. Most of the analysis was carried out in the basic house of quality matrix, although the approach has been applied to concept and tool selection.

The QFD team sizes were under ten members. More people have been involved, but in such cases the team leader arranged the meetings for smaller groups at any one time. The dimensions of the QFD matrices were under 30 customer and technical requirements in general, but in some cases a larger number of issues were managed by using a justified way of grouping. The time taken by the QFD planning and analysis was two or so months in each project. The impression is that the motivation in utilizing QFD was the individual's wish to get their job done better. There were no formal instructions to use QFD.

All the interviewees consider the application of QFD successful. A general perception is that the use of QFD is gaining popularity in the organization. The general awareness of the method and skills in the techniques are increasing, while the news of successful application of QFD is spreading within the organization. Some opponents are known exist, too; the adoption of the method is seen as gradual rather than a sudden rush.

Chapter 6

Application guidelines

This chapter gives suggestions to practitioners about things that might be worth considering in utilizing and applying QFD in projects. The guidelines are synthesized from information found in literature sources, experiences from the analyzed cases, and general observations from real-life software development projects.

6.1 Choosing a method

The purpose of using a method is to establish or improve a process and bring competitive advantage. This requires a certain degree of maturity in an organization. Often the exact selection of method is not that important, sometimes the outcomes of the processes improve just by thinking about and discussing them. Many problems are result of inadequate communication. Better communication and systemacy help in focusing the efforts. A common method can be a way of reaching the goal, because it fosters communication and systemacy.

Development projects are unique and the development process is also a human and social activity. There are not precise algorithms for reaching the goals. QFD is one approach to organizating the efforts; it is not a scientific process or a magic formula for success. The method is relatively easy to adapt and the techniques are highly flexible and customizable. The flexibility is sometimes a burden, because it can make the application of the method harder, due to the high level of freedom.

Although knowledge about QFD method can be gained from books, it is not always easy to know where to start. Reality often differs from book examples, while a practitioner must find her or his own ways of applying the method. The experiences of other practitioners can help at the beginning, but the skills and knowledge improve only by trying the method. There is no shortcut, and as the saying goes, experience is what you get when you get something you did not want.

6.2 Applying QFD

There are innumerable ways to adapt and apply QFD in development processes. Some variations involve making strategic level changes in the organizational goals and processes, some require just small changes in the way a development team organizes its efforts. Adapting QFD may be a minor but significant move in a small or medium size company. In the following guidelines, the viewpoint of small and medium scale organizations is particularly kept in mind. Of course, a larger company can adapt QFD using fairly discreet and small moves, too.

This work presents the application issues and guidelines in three contexts: organization, people, and implementation. Risk issues are addressed in addition to these three. The division is somewhat arbitrary, because the issues are often interconnected and some of them cross the category boundaries.

6.2.1 Organization

A prerequisite for successful application of QFD at the organization level is that the organization is customer oriented and committed to delivering customer satisfaction. This requires some thought of who the customers really are. It is important particularly when the other organizational values, objectives, and strategies are not clearly expressed, as in many cases with small and medium scale enterprises.

Organizational support for adapting QFD is considered important.

The importance of support is emphasized when QFD is the policy of the organization or it is becoming the policy. In smaller-scale cases the change is typically not so dramatic. Often it is sufficient that the organization and existing practices do not hinder an individual willing to voluntarily adapt the QFD techniques. The practices should allow some flexibility and the organization should tolerate failures and use them for learning.

Management support may be necessary in forming the cross-functional teams. The forming of teams is likely to require some authority to get the necessary people together, especially when the people come from different organizational units. The simplest case is of course when the QFD initiator in the team is already in a position that authorizes him to ask people to work in the team.

Management should recognize the position of the cross-functional team and grant the necessary resources. The scarcest resource is typically time. Participating in the teamwork must not be an extraneous burden to the members; the members' other responsibilities should be relieved accordingly. Compensation in the form of money is seldom a viable solution in the long term.

6.2.2 People

Successful utilization of QFD at the project level requires that there is at least one enthusiastic and capable individual to guide the application process. Things may be easier if this person is the leader of the team or owner of the overall process. It is not necessary that this QFD champion is the team leader; the champion can be an additional team member or an experienced outsider. Having a facilitator or peer support does not mean that the team leader should not be committed to the QFD policies.

The motivation of the cross-functional team is important. People need justification for changing their work practices, especially when QFD is applied for the first time. Gains like "our company will prosper" may be too abstract or too impersonal. For some people, just getting their job done better or feeling a sense of accomplishment gives the motivation. Somebody may need financial incentives for contributing to the teamwork.

People in a cross-functional team are likely to have different attitudes and different compensation arrangements, for instance.

There are times when the QFD work is likely to progress slowly. This is totally to be expected. The team leader or the QFD champion should have the persistence to carry on and keep up the motivation. Motivating the team may involve discussions about the goals, applied techniques, and the necessary level of details in reaching consensus.

The team members should get acquainted with the QFD techniques before trying to apply them. Having the whole group trained in QFD, possibly in an external training company, is naturally a fine, but often very expensive, solution. In the initial phases of adapting to the method, it is usually sufficient to have one person in a team either trained in QFD or having experience in QFD. Such a champion can teach the other team members, and tell and motivate them about what they are trying to accomplish as the work progresses. This kind of a targeted just-in-time training arrangement can save time, money, and effort, particularly when the alternative is more general training, which might not even take the QFD application policies of the project into account.

6.2.3 Implementation

Contributing to customer satisfaction emjamces all activities in the successful implementation of QFD processes. Finding out the customer needs and wants, and developing good solutions to them in timely and effective ways, creates value to both customer and company. When left alone, a team developing product or service gets easily drifted and starts generating non-essential features that do not add value. Instead, the team should find out what the customer considers important.

When resolving the customer importance ratings, the right question is: How important is fulfilling a need to contributing to customer satisfaction? Asking questions like "How much a customer is willing to pay for fulfilling a need?" should be postponed until the later phases of analysis, otherwise the analysis will be severely misguided. Questions must be formulated carefully. The focus is on customer needs and wants; the questions should not reflect the intended solutions.

The customer or customer groups should be identified. The QFD process should be carried out keeping the identified customers in mind. Software systems are complex and often involve various stakeholders. Applying QFD to software development can benefit from classifying the stakeholders according their characteristics and importance. A dedicated matrix type can be used for the stakeholder analysis.

Keeping the customer requirements at the same level of abstraction in the matrix may not be easy. Particularly in developing software, the requirements may have very distinct characteristics. Besides, the means of fulfilling the requirements may differ in fundamental ways. It is worthwhile considering having separate matrices for functional, non-functional, and project requirements. In some cases, it is useful to have an additional matrix for analyzing relationships between functional and non-functional requirements. Having separate matrices for distinct types of requirements helps in keeping the number of items in a matrix lower.

The practical limit for matrix dimensions is 20–30. Exceeding this number makes the analysis a laborious task. When analyzing the relationships between customer requirements and technical requirements—the "whats" and the "hows"—the work should proceed in the order of technical requirements. If the customer requirements are considered first, an unnecessarily large number of relationships are likely to result.

Although it is safe to follow one or other of the documented disciplines in applying QFD, practitioners are encouraged not to abandon the use of common sense. QFD is flexible and customizable for the purposes of a particular project. Sometimes it does not make sense to apply all features or phases of QFD. Practitioners should also consider other methodologies and tools like unified modeling language (UML) in the software development. QFD may be suitable, for instance, in the initial phases of requirements engineering and in software architecture selection. Matrix techniques are practical for cause–effect analysis, as they are in risk assessment. The rest may deal better with other approaches. One should not waste effort in making the problem fit the method; tools should be chosen according to the problem.

There are software tools available for carrying out the analysis in the QFD matrices. In addition, the tools have functionality for interfacing with the other software, like spreadsheets and other office applications. Simple application cases do not necessarily need a dedicated software product; the matrix could be managed with a spreadsheet application or by hand, for instance. However, dedicated QFD software can essentially help the routine work, reduce errors, and provide reports in the analysis. Investing in a software tool is likely to pay back in terms of saving work.

6.2.4 Risks

If the scope of a QFD project is not clearly defined, the success of the project is put at risk. The analysis becomes difficult and the results are likely to be worthless if the project tries to offer everything to everybody. An analysis should focus on a relevant stakeholder group and avoid considering multiple products at the same time. Wider cases can be split to subanalyses or separate cases. Segmenting and targeting is usually beneficial; a case can be expanded and new segments can be selected in the successive projects or phases. In new product development, the incremental or iterative approach reduces risks by providing comparable reference products or services for customer competitive assessments.

The key to applying QFD is capturing the voice of the customer. However, there is a risk of having fake customers and consequently, incorrect results. For instance in requirements engineering, the software development team may use its own specialists as customer representatives. The emphasis of a potential customer may be totally different from that of a specialist who has an insider's view of the product. It is essential to capture the opinions of real customers in the target group or at least get as close to them as possible.

A risk to the progress of a QFD project is the team's getting stuck at a particular level of detail. The idea is to construct a shared vision of the product or service. A shared view is a result of taking various, possibly conflicting, conceptions into account and composing a consensus of their importance and implications. The team must allow for skipping some details in the process, because not all of them are meaningful to the vision of the product. Engineers may have a slightly too-scrupulous attitude to the method; QFD is not a scientific process or a magic formula. Some details can be better agreed on in later design phases when there is more knowledge about the issue and the overall goal is clear.

The goal of QFD becoming an established practice in an organization is put at risk, if the first application of QFD fails. People tend to have a certain degree of resistance against change and utilizing QFD in their work implies a change of practice. The effect of human resistance will be amplified if the change is associated with experience of a failure. If the application of QFD fails in the first try, people may become so hesitant that they will never try it again. To avoid this deadlock situation, the application project, QFD champion, and the team should be chosen carefully. It is important to emphasize a certain tolerance of failures and the learning aspect of the project.

Chapter 7

Conclusions and future work

In the beginning, four objectives were stated for this thesis. Chapters 2 and 3 contribute to the first objective of creating understanding about the quality function deployment method by presenting the concepts and procedures of QFD. These chapters describe the QFD method and how it is applied in the industry. The description covers the basic elements of QFD and the phases of the application process. Literature sources are used to give examples of QFD variants and approaches in different application areas including software engineering. QFD is also compared to some other decision-support methods.

Chapter 4 contributes to the second objective, summarizing the preconditions for successful application of QFD. The chapter discusses the meaning of success and gives an overall analysis of QFD, while creating an understanding of what QFD actually promises and what is necessary in order to keep those promises. The observations and conclusions are derived from the literature.

Chapter 5 contributes to the third objective of gathering experiences of QFD in real-life projects. The chapter presents direct observations from QFD application cases. Material for the iterative case-based analysis has been gathered by interviewing practitioners for their experiences and findings from the cases that are summarized.

Finally, Chapter 6 contributes by giving suggestions to a practitioner about things that might be worth considering in utilizing and applying

QFD in projects. The guidelines are synthesized from information found in literature sources, experiences from the analyzed cases, and general observations from real-life software development projects.

By endeavoring to these objectives, the thesis has attempted to approach the goal of surveying QFD experiences in the literature and in the real-life application cases and synthetizing a set of guidelines for applying QFD in requirements engineering. QFD is a systematic approach for capturing and communicating the issues that customers consider valuable in the development process. It serves the purposes of requirements engineering. The basic techniques are relatively easy to learn and understand. The method is a framework for organizing efforts in a meaningful way; it is not a scientific process or a formula with a deterministic outcome. Guidelines for applying the method have been given, but because the development process is a human activity and because each instance of the process is unique, the goal of summarizing all the correct, sufficient, and necessary recommendations has not been fully met.

For the enthusiastic researcher, the future holds several interesting challenges. For instance, this work synthesizes some guidelines for a practitioner who is planning to utilize QFD in a small or medium sized organization. It would certainly be interesting to validate the guidelines and their value in real-life projects. Another interesting topic would be the utilization QFD in software development. Although there are various approaches to using QFD in software engineering and the usefulness of the method in that area is supported by several claims, QFD has not been widely adopted in requirements engineering. Maybe the right way of using QFD in software development remains yet to be discovered.

Bibliography

Beskow, C., Johansson, J. & Norell, M. (1998), Implementation of QFD: Identifying success factors, *in* L. S. Peters, ed., 'IEMC '98 Proceedings, International Conference on Engineering and Technology Management, San Juan, Puerto Rico, USA, October 11–13, 1998', Institute of Electrical and Electronics Engineers, Inc., Piscataway, NJ, pp. 179–184.

Brandt, S. (1996), Applying QFD in software engineering, Master's thesis, Tampereen teknillinen korkeakoulu, Tampere, Finland.

Cristiano, J. J., Liker, J. K. & White, III, C. C. (2001), 'Key factors in the successful application of quality function deployment (QFD)', *IEEE Transactions on Engineering Management* **48**(1), 81–95.

Day, R. G. (1993), *Quality Function Deployment: Linking a Company with Its Customers*, ASQC Quality Press, Milwaukee, Wisconsin.

Green, P. E. & Rao, V. R. (1971), 'Conjoint measurement for quantifying judgemental data', *Journal of Marketing Research* **8**(3), 355–363.

Green, P. E. & Srinivasan, V. (1990), 'Conjoint analysis in marketing: New developments with implications for research and practice', *Journal of Marketing* **54**(4), 3–19.

Gustafsson, A., Ekdahl, F. & Bergman, B. (1999), 'Conjoint analysis: a useful tool in the design process', *Total Quality Management* **10**(3), 327–343.

Haag, S., Raja, M. K. & Schakade, L. L. (1996), 'Quality function deployment usage in software development', *Communications of the ACM* **39**(1), 41–49.

Herzwurm, G., Ahlemeier, G., Schockert, S. & Mellis, W. (1998), Success factors of QFD projects, *in* R. L. Chapman & R. Hunt, eds, 'Proceedings of the World Innovation and Strategy Conference WISC98, Sydney, Australia, August 2–5, 1998', University of Western Sydney, Campbelltown, Australia, pp. 27–41.

Herzwurm, G., Schockert, S. & Pietsch, W. (2003), QFD for customer-focused requirements engineering, *in* 'Proceedings of the 11th IEEE International Requirements Engineering Conference RE'03, Monterey Bay, California, USA, September 8–12, 2003', Institute of Electrical and Electronics Engineers, Inc., Piscataway, NJ, pp. 330–338.

Hofstede, G. (1991), *Cultures and Organizations: Software of the Mind*, McGraw–Hill, London.

Hrones, Jr, J. A., Jedrey, Jr, B. C. & Zaaf, D. (1993), 'Defining global requirements with distributed QFD', *Digital Technical Journal* **5**(4), 36–46.

Hurri, J. (2000), Using decision tools in deciding system product requirements: literature review and a behaviourally motivated lightweight tool, Licentiate thesis, Helsinki University of Technology, Espoo, Finland.

Karlsson, J. (1997), 'Managing software requirements using quality function deployment', *Software Quality Journal* **6**(4), 311–325.

Karlsson, J. & Ryan, K. (1997), 'A cost–value approach for prioritizing requirements', *IEEE Software* **16**(5), 67–74.

Krogstie, J. (1999), Using quality function deployment in software requirements specification, *in* A. L. Opdahl, K. Pohl & E. Dubois, eds, 'Proceedings of the Fifth International Workshop on Requirements Engineering REFSQ'99, Heidelberg, Germany, June 14–15, 1999', University of Namur, Namur, Belgium, pp. 171–185.

Lehtola, L. (2003), Käyttäjä- ja asiakasvaatimusten priorisointi: Priorisointikäytännöt todellisessa tuotekehitysympäristössä, Master's thesis, Teknillinen korkeakoulu, Espoo, Finland.

Liu, X. F. (2000), 'Software quality function deployment', *IEEE Potentials* **19**(5), 14–16.

Mazur, G. (1998), *QFD and designing software*, in ReVelle et al. (1998), chapter 10, pp. 139–162.

Pullman, M. E., Moore, W. L. & Wardell, D. G. (2002), 'A comparison of quality function deployment and conjoint analysis in new product design', *Journal of Product Innovation Management* **19**(5), 354–364.

ReVelle, J. B., Moran, J. W. & Cox, C. A. (1998), *The QFD Handbook*, John Wiley & Sons, Inc., New York.

Saaty, T. L. (1980), *The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation*, McGraw–Hill, New York.

Salo, A. A. & Hämäläinen, R. P. (1997), 'On the measurement of preferences in the analytic hierarchy process', *Journal of Multi-Criteria Decision Analysis* **6**(6), 309–319.

Standish Group (1995), The CHAOS report (1994). Retrieved December 5, 2003, from http://www.standishgroup.com/sample_research/chaos_1994_1.php.

Total Quality Software (2003), QFD2000 software presentation material. Retrieved December 5, 2003, from http://www.qfd2000.co.uk/featexample.htm.

Vaisala Group (2003), Vaisala Group presentation material. Retrieved December 5, 2003, from http://www.vaisala.com/.

Zultner, R. E. (1998), *QFD and designing software*, in ReVelle et al. (1998), chapter 11, pp. 163–191.

Zultner, R. E. (2000), Blitz QFD, software QFD for very rapid development. Retrieved December 5, 2003, from http://som.utdallas.edu/c4isn/slides/C4ISN_zultner_bqfd.pdf.