QF²D: Quality Factor Through QFD application

Luigi Buglione *, Alain Abran**

Software Engineering Management Research Laboratory
Université du Québec à Montréal
C.P. 8888, Succ. Centre-Ville
Montréal, Québec, Canada
* *luigi.buglione@computer.org*
** *abran.alain@uqam.ca*

Abstract: In recent years, several Total Quality Management (TQM) techniques have been introduced for better Software Quality Management. One of these techniques is Quality Function Deployment (QFD). This papers shows how QFD has been applied to an existing software quality assessment technique, the Quality Factor technique (QF). This improved technique, QF²D, uses ISO/IEC 9126:2000 standards in its implementation. The procedure for deriving QF²D is presented, together with an example.

Résumé: Au cours de *ces dernières années plusieurs techniques de TQM ont été importées et utilisées pour une meilleure gestion de la qualité du logiciel. Une de ces techniques , le Déploiement de la Fonction de la Qualité (QFD - Quality Function Deployment), a été appliquée à une technique existante d'évaluation du logiciel, le Quality Factor (QF). Cette technique améliorée, dénommée QF²D, utilise de plus la nouvelle norme ISO/IEC 9126:2000, . La procédure pour l'utilisation du QF²D ,,est présenté, ainsi qu'un exemple.*

Key words: Software Quality, Quality Function Deployment, Quality Models, Quality Factor, ISO/IEC 9126-14598 series.

1. INTRODUCTION

Total Quality Management (TQM) is a term coined by the US Naval Air Systems Command in 1985 to describe the Japanese-style management approach to quality improvement. A central goal in TQM is to fully achieve customer satisfaction. Several techniques and approaches were developed in Japan, such as Just-In-Time (**JIT**), Quality Function Deployment (**QFD**), Design of Experiment (**DOE**) and Statistical Process Control (**SPC**). Some of these techniques, originally designed for the manufacturing industry, were "adapted" to the ICT world. In recent years, both academia and industry have focused on an application of TQM, which is software product quality (for example, the ISO/IEC9126 and 14598 series). At the QUALITA'99 conference, we proposed a technique called the Quality Factor (QF) technique for a qualitative evaluation of software [Buglione & Abran 99a]. QF returns a value for software product quality, derived from a comparison of users', developers' and managers' opinions about the project being measured. QF can also be used in conjunction with the QEST model and its extension to the whole software life cycle (SLC), referred to as the LIME model [Buglione & Abran 00a]. These models consider simultaneously three distinct but connected areas of interest, each of them representing a distinct dimension of performance:

- **economic dimension**, the perspective is the managers' viewpoint;
- **social dimension**, the perspective is the users' viewpoint;
- **technical dimension**, the perspective is the developers' viewpoint.

This paper reports on an update to the design of the Quality Factor technique (QF),, called **QF^{2}D** (*Quality Factor through QFD*): QFD is used to improve the original QF technique with a priori and a posteriori evaluations of the software product. In addition, its uses the most recent version of the list of quality sub-characteristics in the ISO/IEC 9126:2000 standard [ISO 99a,b,c,d].

Section 2 briefly summarises the foundations of the QF technique, identifies some limitations and the TQM domain investigated for improvement. Section 3 presents the technique selected, QFD, and its applications in the software field. Section 4 provides a description of the new, improved technique, QF^{2}D, together with an example. F Section 5 identifies some further uses for QF^{2}D.

2. QUALITY MODELS AND THE QUALITY FACTOR (QF)

A Quality Model (**QM**) is defined as "the set of characteristics and the relationships between them which provide the basis for specifying quality requirements and evaluating quality" [ISO 86]; alternatively, it is also defined as a "structured set of properties required for an object of a class to meet defined purposes" [Fusani 95].

The initial QF qualitative software evaluation technique, in our QEST quality model, was based on the 1991 version of the ISO 9126 standard [ISO 91], which included 6 software product characteristics, but only 21 sub-characteristics[1].

To obtain the **Q**uality **F**actor (**QF**) for software the following high-level procedure flow, , is used [Buglione & Abran 99a][2]: First, a questionnaire is submitted and filled out both by users by developers and managers to express their opinions on the quality of the software product. Then, the Measurement Working Group (MWG) responsible for the management of the procedure collects these opinions and, through the QF algorithmic procedure, derives the quality value for the project being assessed. This QF technique has been used in a major Italian company in the utilities sector.

Feedback from trials in industry have identified some limitations of the technique, such as an initial design only suitable for evaluation purposes, and a cumbersome procedure using multiple tables. To address these issues, an analysis was performed on the relevant quality practices and methodologies used in the Japanese manufacturing industry, such as Total Quality Management (TQM), based on the Japanese philosophy of **Kaizen** [Imai 86]. This led to the identification of tools/methods available for addressing the shortcomings of the initial QF design[3], one of which was Quality Function Deployment (QFD). The strategic focus of QFD required the identification, from the business requirements, of the most important items to develop in order to produce products aligned as closely as possible with customer requirements. A non-optimal

[1] In [Buglione & Abran 99a] a discussion is presented on the rationale for this choice among the main Quality Models (QMs).

[2] The complete method and references for the forms and questionnaire are described in [Buglione & Abran 99a] [Buglione 99].

[3] http://dfca.larc.nasa.gov/dfc/qtec.html is a NASA website containing an overview of quality technologies for Competitive Advantages, as well as a list of quality tools.

gathering and prioritisation of customer requirements can lead to a lower level of quality than expected, and in higher Time-To-Market (TTM) and overall production costs. Furthermore, QFD makes it also possible to summarise data from multiple viewpoints in a single table. That relevance is confirmed through analysis of the distribution of defects during the whole Software Life Cycle (SLC). For these reasons, QFD was chosen to improve QF.

3. QUALITY FUNCTION DEPLOYMENT (QFD)

Quality Function Deployment (**QFD**) was designed in the late ’60s in Japan as a means of translating customer requirements (the so-called “*Voice of the Customer*”) into appropriate technical requirements throughout the development and production of a product. In the ’80s, this method was introduced into North America, with an expansion of its initial focus on the product to the whole production chain (the “4 phases of QFD”: HoQ (or Design), Parts (or Details), Process, Production) [Crow] [Dean] [Herzwurm et al. 98] [Richardson 98] [Mizuno et al. 94].

QFD includes a series of matrices [Mizuno et al. 94], which are the tools used to represent data. The most commonly used matrices are the “House of Quality” matrix (HoQ – American Supplier Institute (ASI) terminology) or the “A1 Matrix” (Growth Opportunity Alliance of Lawrence GOAL/QPC terminology) described in [Hauser & Clausing 88]. This structure is presented in Figure 1. For a detailed description of QFD and its customisation for the software world, refer to [ESI 99]. Basically, the HoQ presents the intersection of two dimensions, the horizontal one (the WHATs) and the vertical one (the HOWs). In terms of product development, the WHATs identify the characteristics of the product and/or services desired by the customer, while the HOWs identify the way to achieve the WHATs[4].

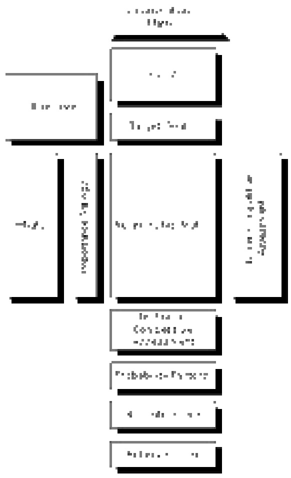


Figure 1: House of Quality (HoQ) structure.

The relationship matrix, the central part of the HoQ, represents the heart of the system, providing for the prioritisation of the WHATs through the attribution of importance ratings. These are calculated by multiplying each WHAT importance level (often measured on a Likert scale) by its intensity relationship with the HOWs, using symbols with associated weights.

4. QF^{2}D: MAIN FEATURES AND PROCEDURE

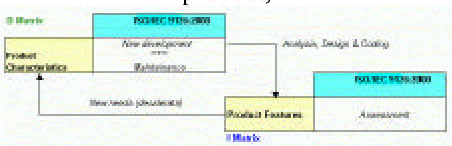
By integrating into QF the QFD concepts and matrix manipulation techniques, the **QF^{2}D** (**Q**uality **F**actor through **QFD**) was derived. The procedure for compiling the matrices is the same as in QFD, and the two are summarised together for comparison purposes. While [Buglione & Abran 00c] discuss main the advantages

[4] For a detailed explanation of the HoQ and how to fill out the matrix, refer to [Guinta & Praizler 93] [Herzwurm et al. 98] [MOC].

and differences relative to the original QF technique, this paper focuses on the procedure needed to calculate the QF^2D value and presents, in Appendix 1[5], a comparison with QF in terms of mathematical formulae.

QF^2D's new structure is based on a variable number of *requirements* in the development phase, or *features* in an a posteriori assessment of the software product implemented. Furthermore, QF^2D uses the updated ISO standard, where the number of software quality sub-characteristics has increased from 21 to 31 (ISO/IEC 9126:2000 series).

Figure 2 illustrates the framework of QF^2D for the evaluation of the software product,



with both the *Development* (D) matrix and the *Implemented product* (I) matrix.

Figure 2: QF^2D life cycle

The first step (D matrix) gives a quantification of the required level of product quality, starting from the desired product characteristics, each of them mapped onto the quality sub-characteristics proposed by ISO from the three viewpoints. These quality requirements must then be taken into account in the production of the software product. Once the software has designed and coded, it has to be tested and evaluated. Then, the product features implemented will be mapped onto the ISO sub-characteristics list to assess the quality of the software product, as implemented (I matrix). The comparison of values across the two matrices will provide feedback to the development phase. In this way, QF^2D gives an organization the opportunity to monitor the quality of a software product in a dynamic way throughout its life cycle. In the D matrix, for instance, the WHAT expresses the *requirements* of the three interest groups (E, S, T), giving each a priority following a Likert scale (from 1 to 5). In QF^2D, it is recommended that all the stakeholders in the three groups be considered, as in quality models like EFQM and the Malcolm Baldrige quality models. The HOW is expressed through the list of quality sub-characteristics from the new ISO/IEC 9126 standard (parts 2, 3 and 4). The relationship between *requirements* and quality sub-characteristics is expressed on the ISO/IEC 14598-1 [ISO 97] scale (from 0 to 3), instead of the QFD graphic symbols used in the HoQ, as proposed in Figure 4. Appendix 2 summarises the elements referred to in the two matrices.

The new procedure for calculating the quality value QF^2D is presented next in six steps:

1. Selection, by the respondents, of the most relevant requirements;
2. determination, for each requirement, of:
   - a priority level (1-5 scale);
   - which sub-characteristic(s) is(are) correlated to the requirement;
   - which rating (0-3 scale) for the sub-characteristic is associated with the requirement.
3. calculation of the sub-characteristic values (SCVs);
4. calculation of the each whole characteristic value (CV);
5. calculation of the Total Characteristics Value (TCV);
6. determination of the final QF^2D value (TCV/TCV_{max}).

A short example with five respondents (1 manager, 2 users and 2 developers) answering a quality questionnaire is presented below. After the QF^2D matrix

[5] The quality questionnaire, an Excel template for calculating the QF^2D value and Appendices can be downloaded at: http://www.geocities.com/lbu_measure/qf/qf2d.htm

had been filled in, the result derived is presented in Figure 3.

Figure 3: D matrix results (excerpt)

It is to be observed that the final aggregated result is derived by filling in a single table instead of the four needed in the initial design of the QF technique. In this example, and using the formulae in Appendix 1, QF is equal to 33.85% (calculated as the ratio between TCV=787 and TCV_{max}=2325).

The next level of analysis consists in decomposing it against the 6 quality characteristics and 31 sub-characteristics, and looking at the contribution of each single product quality aspect (Figure 4).

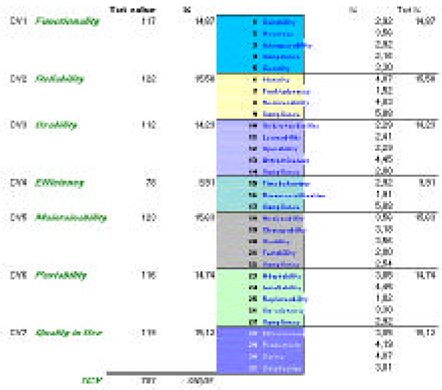


Figure 4: D matrix results analysis

In this example, Efficiency (CV4) is the characteristic with the lowest rating, and Replaceability (SCV25) the sub-characteristic with the lowest rating.

To evaluate the contribution of each viewpoint group (Economic, Technical and Social) to each sub-characteristic, it is sufficient to take into account the percentage of ratings per sub-characteristic in each group out of the total. For instance, referring to Figure 3, the rating assigned to sub-char #1 was 23 points, made up of 3 by the "E" group (13.04%), 14 by the "S" group (60.87%) and 6 by the "T" group (20.09%). These considerations will be useful for the second matrix (*I matrix*) application. Using the information from the D matrix, the development of the product can be modified in order to "fit" the initial requirements in a better way, taking into account all the stakeholders' viewpoints at the same time. Similarly, for next project of course: from the I matrix to the new D matrix for the new product being built.

The advantages of this new improved version of QF are as follows:

- use of the improve ISO/IEC 9126 standard series;
- not only a product evaluation, but also a new development/maintenance is provided, with feedback;
- use of the ISO/IEC 14589-1 evaluation scale to express relationships in the HoQ table;
- greater granularity in the whole product evaluation, at the sub-characteristic level;
- a simpler and quicker procedure for calculating the product quality value.

5. CONCLUSIONS & PROSPECTS

This paper has described the **QF^{2}D** (**Q**uality **F**actor through **QFD**) technique, a method for measuring software quality in a quantitative way. This method is based on a multi-dimensional view of software quality, using different stakeholders' viewpoints, and on the application of international standards, such as the ISO 9126 series, on software quality characteristics. Combining QF and QFD makes it possible to obtain an evaluation that covers the whole software product life cycle, returning useful information from the design to the production to the maintenance phase, through the use of two HoQ-based matrices.

The main changes to QF technique have been presented, as well as the new calculation procedure, leading to reduced time and complexity in determining the final value for software quality.

The QF^2D framework could also be used to focus on a particular perspective, for a more detailed analysis. For instance, the ISO 9241-11 standard on software usability can be used if the aim is to analyse the usability issue more closely, considering then every quality characteristic as a separate dimension.

Finally, the integration between the QF/QF^2D techniques and the QEST/LIME models must be noted [Buglione & Abran 99b]: QF^2D, can be used separately as QF, or jointly, with these two multi-dimensional software performance models..

6. REFERENCES

[Buglione 99] Buglione L, *Misurare il software. Quantità, qualità, standard e miglioramento di processo nell'Information Technology*, Franco Angeli, ISBN 88-464-1729-1, 1999

[Buglione & Abran 99a] Buglione L & Abran A., *A Quality Factor for Software*, Proceedings of QUALITA99, 3rd International Conference on Quality and Reliability, Paris, France, 25-26 March 1999, pp - 335-344.

[Buglione & Abran 99b] Buglione L. & Abran A., *Multidimensional Software Performance Measurement Models: A Tetrahedron-based Design*, in "Software Measurement: Current Trends in Research and Practice", R. Dumke/A. Abran (eds.), Deutscher Universitats Verlag GmbH, pp. 93-107, 1999

[Buglione & Abran 00a] Buglione L. & Abran A.: *LIME: A Three-dimensional Software Performance Measurement Model for Project Management.*, Proceedings of the 2nd World Congress on Software Quality (2WCSQ), September 25-29, 2000, Yokohama, Tokyo, Japan

[Buglione & Abran 00b] Buglione L & Abran A.., *Implementation of a Three-Dimensional Software Performance Measurement Model*, Technical Report, Université du Québec à Montréal (UQAM), to be published (2000)

[Buglione & Abran 00c] Buglione L & Abran A.., *QF^2D: a different way to measure Software Quality*, 10th Proceedings of the 10th International Workshop on Software Measurement (IWSM2000), October 4-6, 2000, Berlin, Germany

[Crow] Crow K., *Customer-Focused Development with QFD*, DRM Associates, URL: http://member.aol.com/drmassoc/QFD.html

[Dean] Dean E.B., *Comprehensive QFD*, URL: http://akao.larc.nasa.gov/dfc/qfd/cqfd.html

[Deming 00] Deming W.E, *Out of the Crises*, MIT Press, 2000, ISBN 0262541157

[ESI 99] Benguria G, Buglione L., Delgado E., Ferrer P. & Ostolaza E., *EFQM/SPICE Integrated Model version 2. Quality Function Deployment*, ESI-1999-TR-046, December 1999

[Fusani 95] Fusani M., Quality Models for Software Evolution Instruments, *International Seminar on Software Measuring & Testing*, IEI-CNR/Qualital/SSSUP "S.Anna", Pisa, Italia, December 12, 1995

[Guinta & Praizler 93] Guinta L.R. & Praizler N.C., *The QFD Book: The Team Approach to Solving Problems and Satisfying Customers through Quality Function Deployment*, Amacom Books, 1993, ASIN 081445139X

[Hauser & Clausing 88] Hauser J.R. & Clausing D., *The House of Quality*, Harvard Business Review, May-June 1988, pp.3-13

[Herzwurm & al 98] Herzwurm G., Ahlemeier G., Schockert S. & Mellis W., *Success Factors of QFD Projects*, Proceedings of the World Innovation and Strategy Conference, August 3-5, 1998, Sydney, Australia, pp. 27-41, URL: http://www.informatik.uni-koeln.de/winfo/prof.mellis/publications/qfdsf.zip

[Imai 86] IMAI M., *Kaizen: the key to Japan's competitive success*, McGraw Hill, 1986, ISBN 0-07-554332-X

[ISO 86] ISO, *International Standard 8402: Quality - Vocabulary*, 1986

[ISO 91] ISO/IEC, *International Standard 9126: Information Technology - Software product evaluation – Quality characteristics and guidelines for their use*, 1991

[ISO 97] ISO/IEC JTC1/SC7/WG6, *DIS 14598-1: Information Technology - Software product evaluation – Part 1: General Overview*, 3/6/97

[ISO 99a] ISO/IEC JTC1/SC7/WG6 FDIS *9126-1: Software Engineering Product Quality - Part 1: Quality Model*, N2228, 19/11/1999

[ISO 99b] ISO/IEC JTC1/SC7/WG6 *PDTR 9126-2: Software Engineering Product Quality - Part 2: External Metrics*, N2073, 10/03/1999

[ISO 99c] ISO/IEC JTC1/SC7/WG6 *PDTR 9126-3: Software Engineering Product Quality - Part 3: Internal Metrics*, N2083, 30/03/1999

[ISO 99d] ISO/IEC JTC1/SC7/WG6 *WD 9126-4: Software Engineering Product Quality - Part 4: Quality in Use Metrics*, N2208, 29/09/1999

[Mizuno & al 94] VV.AA., *QFD: The customer-driven approach to Quality Planning and Deployment*, S.Mizuno &Y.Akao (Eds.), APD, 1994

[MOC] The Matrix of Change (MOC) Homepage, http://ccs.mit.edu/moc/

[Richardson 98] Richardson I., *Using QFD to Develop Action Plans for Software Process Improvement*, SEPG98 Conference Proceedings, April 1998