Engineering Part IIB

Paper 4C4 - Design Methods

SUMMARY NOTES

Quality Function Deployment (QFD)

Dr Nathan Crilly

Dr James Moultrie

Version 1.1: March 2014

Quality function deployment (QFD)

QFD is a tool intended to assist a development team in focussing on the needs of their customers. It helps to transform customer requirements (the voice of the customer) into engineering characteristics (the voice of the designer). QFD provides a means of prioritising which aspects of the product demand the most attention, and it can help facilitate communication across organisational functions.

1. Background

QFD originates in Japan in the mid-1960s where it was developed to assist in the planning, development and quality assurance of new products. It was adopted in the United States in the mid-1980s, and was further developed and disseminated by organisations such as Ford Motor Company and Black & Decker. Although originating in the manufacturing industries, it is now used not just for developing products, but also systems, processes and services.

2. Terminology

Attempts to understand exactly what the term "quality function deployment" could possibly mean are unlikely to be illuminating. The term is one possible translation of a set of Japanese characters; other translations would have yielded another set of (possibly still inscrutable) words. A more complete translation of the original Japanese terms might be: "strategic arrangement (*deployment*) of appropriate characteristics (*qualities*) throughout all aspects of product (*functions*) according to customer demands". In other words, QFD involves designing for the customer.

In addition to its name, QFD involves a number of different terms, initialisations and symbols. We will strive here to keep this special vocabulary to a minimum and to remain consistent; however, the introduction of some new standard terms and symbols will still be required. Whilst these terms and symbols might not always seem well chosen, they *are* conventional and should assist you in interpreting others' QFD charts.

3. The House of Quality

QFD is a tool that requires the production and interpretation of many interrelated matrices. We will concentrate on the most widely used of these matrices, *The House of Quality*. The house of quality is simply named after the house-like shape that the matrices form. QFD in its most comprehensive form includes matrices other than the house of quality, but they will not be considered here, and in general, we will refer to QFD and The House of Quality interchangeably (as is common practice even though it's not strictly correct).

The House of Quality assists us in defining HOW (according to technical performance measures) we can achieve WHAT (according to the customer) is most important. This definition of a formal relationship between the HOW and the WHAT is at the centre of QFD. In doing so, QFD formalises and structures a relationship that is often considered only informally or intuitively.

We'll consider a simplified version of 'textbook' QFD, as it is typically described and widely used for new product development. The tool should of course be modified if it is being applied to atypical situations. This might involve refining or omitting certain elements as necessary, or completing the matrix in some different order. However, for this course we will assume that QFD should be done as follows…

The House of Quality is built up from 6 interrelated matrices. We will discuss each matrix in turn, indicating how the output from one matrix contributes to the next.

3.1. The Customer Attributes

QFD promotes the separation (and then relation) of customer needs to product specifications. The *Customer Attributes* matrix (really just a structured list) represents 'the voice of the customer', telling us WHAT has to be achieved. In order to determine who your customers are, and what needs they have, some market research is necessary. This might typically involve the analysis of guarantee returns, customer interviews and complaints feedback. From this information a list of needs/benefits is defined. This list can be interrogated by asking questions such as WHO is it that needs WHAT and WHEN do they need it? WHERE will they use the product, and WHY and HOW will they do so?

This list of needs represents the voice of the customer in QFD, and wherever possible the actual words that customers use should be retained. However, the objective here is to define needs, rather than product attributes, so some translation may be required. For example, if customers say they want tinted glass in their vehicles, some further enquiry might be required to determine whether this is a need for privacy or a need for protection from the sun. Either way, it is these (relatively solution-neutral) needs rather than product characteristics that are required.

Tree diagrams and affinity diagrams (see the *Creativity* notes) can then be used to explore the relationships between the different needs that have been identified, and to define further needs and benefits that have been thus far omitted. What results should be a structured list of the customer needs expressed in qualitative (rather than quantitative) terms. Further market research or internal brainstorming can then be used to assign a priority weighting to each need/benefit in accordance to how important it is to the customer. Higher priority values indicate attributes that are more important. The output is a structured and weighted list of needs that represents 'the Voice of the Customer'.

3.2. The Engineering Characteristics

The *Engineering Characteristics* matrix (which is again just a list) describes HOW the product may achieve its required performance in general terms which are not solution-specific. There are three basic options to how these engineering characteristics are defined: i) they can be defined in terms of top level performance characteristics (for example, voltage, mass, time); ii) they can be defined in terms of product functions (for example, stabilise workpiece, remove dust); or they can be defined in terms of product components or subsystems (for example, a bicycle wheel contains a hub, a rim and some spokes). Which of these options is preferable depends on factors such as the type of product that is being developed, the stage of development and the organisation of the team. We will concentrate here on the first of these options which is quite suitable for projects at the early stages where detailed functions and components have not yet been defined.

For each customer attribute define one or more engineering characteristics that are influential. Ensure that these characteristics are measurable *during* development (not just after), and that they can be controlled by the development team. Again, tree diagrams and affinity diagrams can help to ensure that comprehensive coverage of the required engineering characteristics is achieved. The output is a structured or ordered list of engineering characteristics that represents 'the Voice of the Designer'.

3.3. Relationships

The *Relationships* matrix is formed by the intersection of the two lists discussed in the preceding two sections. For each cell in the matrix the development team must decide the strength of relationship that the associated engineering characteristics has on the associated customer attribute. This involves asking what effect making a change to the value of an engineering characteristic has on a particular customer attribute. The convention here is to place different symbols within each cell to indicate the strength of the relationship. Two concentric circles indicate that there is a strong relationship, a single circle indicates a moderate relationship, and a triangle indicates a weak relationship (or that such a relationship is only possible, not certain). If there is no relationship, then the cell is left blank. Numerical values are assigned to each symbol. By convention, these values are 9 for strong relationship, 3 for moderate relationship, 1 for weak relationship and 0 (zero) for no relationship.

The use of these symbols permits the development team to understand at a glance the extent to which any single customer attributes is being addressed by the range of engineering characteristics that have been defined. An empty column indicates an engineering characteristic which customers do not require, or indicates that the customer requirements list is incomplete. An empty row indicates a customer characteristic that is not being met, or indicates that the engineering characteristics list is incomplete.

Note: It is of course possible to get cells that have negative relationship values. This occurs when an engineering characteristic has been defined with respect to one customer attribute, but also affects other customer attributes in the opposite sense. For example, increasing the thickness of a car's door panels increases safety (need X) but decreases fuel economy (need Y). Negative signs in front of associated required symbols and numerical values should then be used. Such instances often highlight key opportunities for technical breakthroughs.

3.4. Technical matrix

With the Relationships matrix complete, it is possible to determine a prioritisation of the engineering characteristics. This is done in the *Technical* matrix by calculating the relative contribution that each engineering characteristic makes to the customer's overall satisfaction. For each cell, the relationship value is multiplied by the prioritisation value assigned to the associated customer attribute. Summing these products (i.e. products of multiplication) for any given column yields the technical prioritisation of the engineering attributes. Just as with the customer attributes priorities, higher technical priority values indicate attributes that are more important.

Consideration of the technical priorities prompts the team to define design targets associated with each engineering attribute. These might include a specification of, for example, maximum weight, minimum shelf life and minimum mean time between failures. All of these targets should be quantifiable, and this is why it is important to define engineering characteristics that are measurable. Design targets might also be informed by competitive benchmarking (discussed in the *Planning Matrix* section).

3.5. Technical Correlations

The Engineering Characteristics of a given product are rarely entirely independent of each other. The *Technical Correlations* matrix aims to make the relationships between engineering characteristics explicit, and the strength of those relationships explicit also. The Technical Correlations matrix forms the 'roof' of The House of Quality, and indicates the technical areas where close communication and collaboration are needed.

Above the engineering characteristics matrix an indication is made of the direction in which that characteristic should be optimised. An upwards arrow is used for "more is better", a downwards facing arrow is used for "less is better" and the letter 'T' is used for "the target value is best" (from the customer's perspective). The technical correlations matrix provides a cell for every possible combination of the engineering characteristics. Within each cell, the development team indicates the effect (or correlation) that a change in one engineering characteristic is likely to have on another. A strong positive correlation is indicated with two concentric circles, a positive correlation is indicated with a single circle, a negative correlation is indicated with a cross, and a strong negative correlation is indicated with a double cross (like a "#"). For example, increasing the rigidity of a component (desirable) and decreasing its mass (also desirable) are likely to be negatively correlated (possibly strongly negative). The result of this process is a record of how the different engineering characteristics are mutually reinforcing or mutually contradictory.

3.6. Planning Matrix

The *Planning Matrix* offers space for the comparison of marketplace competitors, either with each other, or with the product that the development team are responsible for. This requires quantitative market data for each of the customer attributes. The rows of the planning matrix are defined by the customer attributes matrix. And the columns of the planning matrix are defined by a numerical rating system that indicates the degree to which any given product satisfies each item in the customer requirements matrix. Values for each cell of the planning matrix can be based on user research, competitive analysis or team assessment. Either way, it is the customers' ratings (as measured or as expected) that are important. This matrix assists with strategic planning because it defines basic market opportunities in response to the relative strengths of the competition.

4. Strengths and benefits

The strengths and benefits of QFD can be briefly summarised as follows:

- Encourages cross-functional teamwork throughout the product development process, encouraging communication and cooperation
- provides a mechanism for increasing the use of marketing inputs
- Focuses the development team's minds on what they don't know
- Focuses on customer needs, not product features
- Prevents wild unqualified assumptions
- Ensures that a wide range of issues are considered, including customer requirements, technical characteristics and competitive products.

5. Limitations and drawbacks

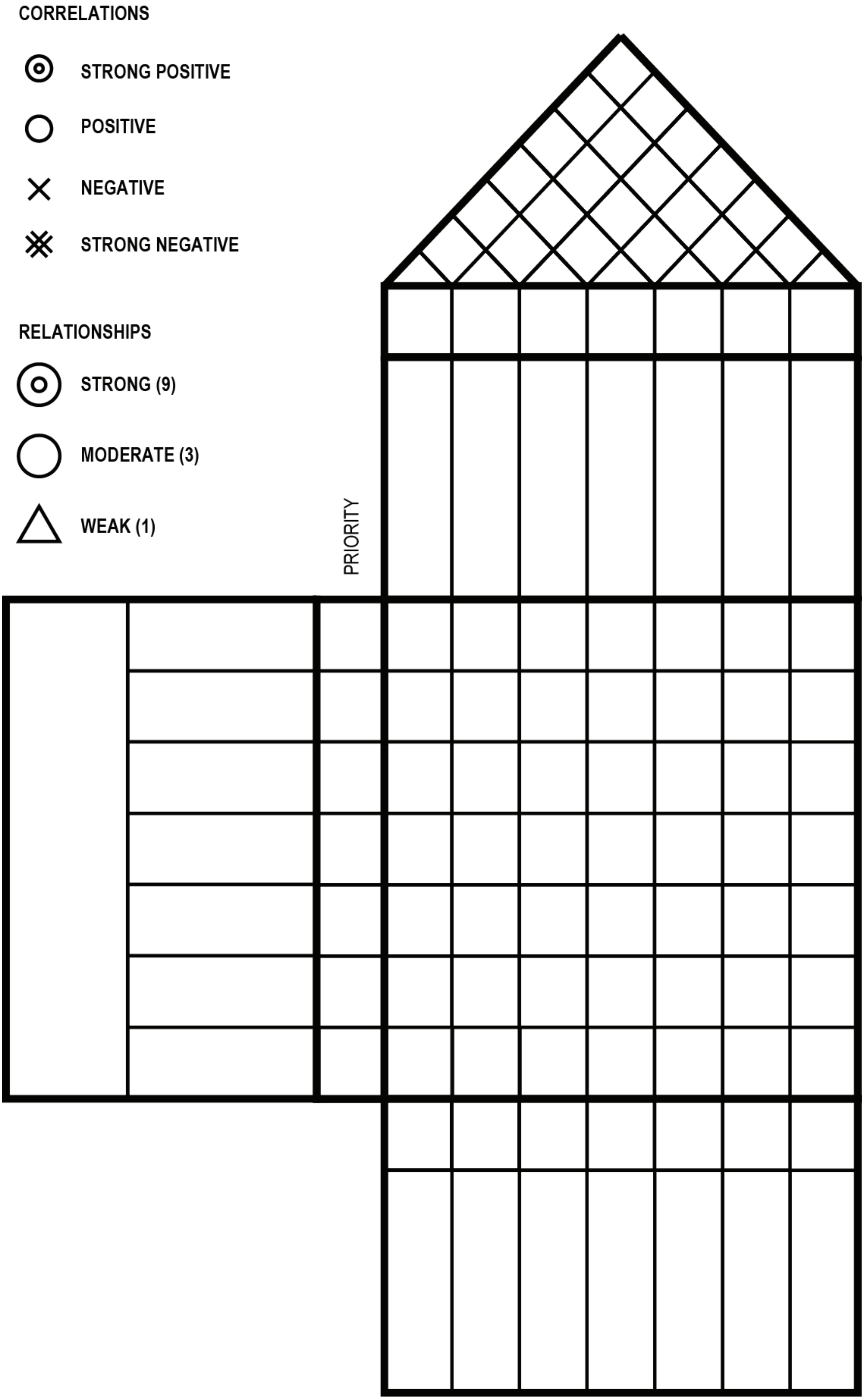
The limitations and drawbacks of QFD can be briefly summarised as follows:

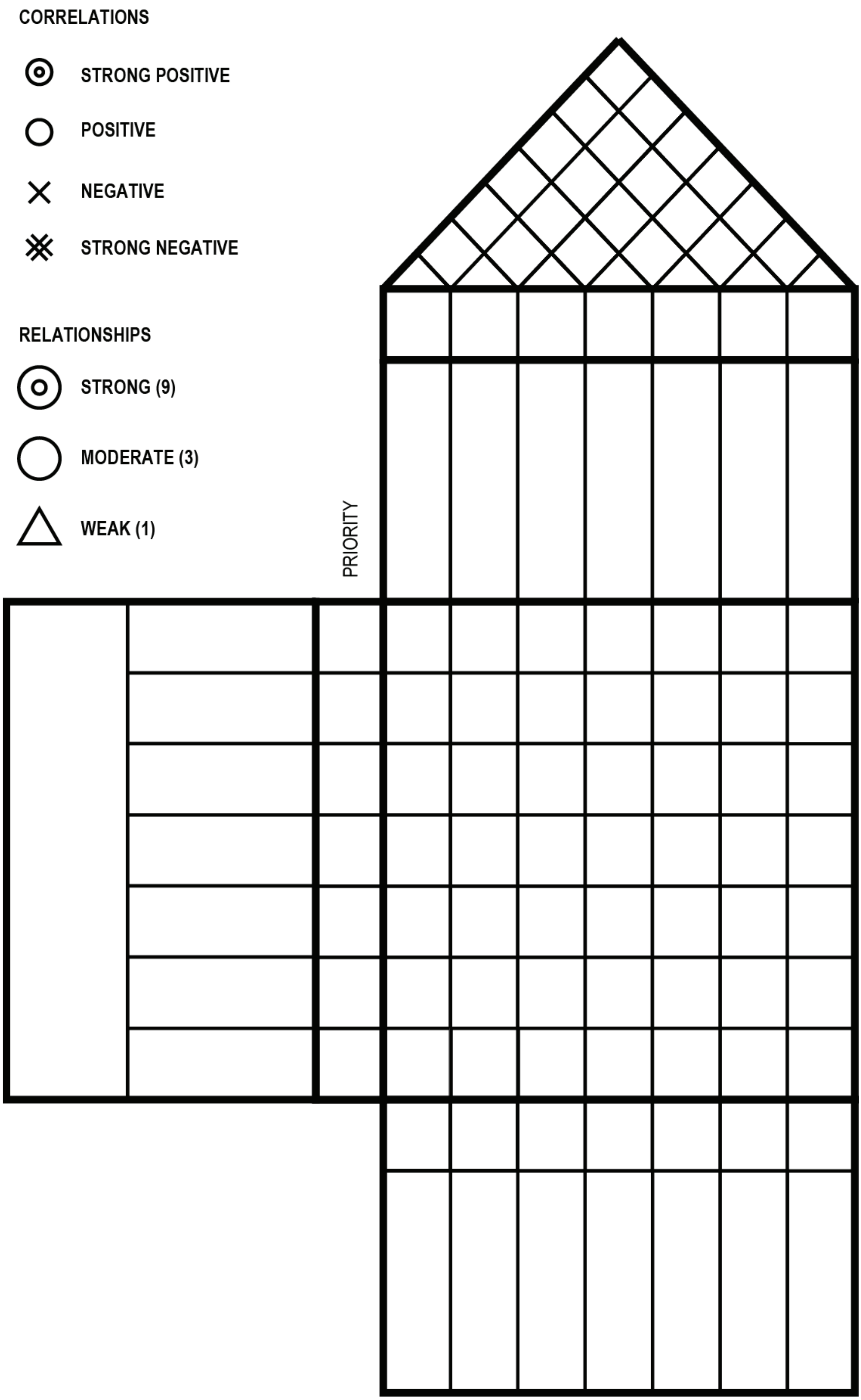
- Requires a cross functional team, including, for example, representatives from marketing, engineering and manufacturing
- Can be exceedingly complex, time consuming and tedious
- Can be too analytical - a numerical answer can be treated as a 'right' answer
- Requires some training and strong facilitation initially
- There are many variants of QFD, each with their own terminology
- There is the risk that attention is devoted not to serving the customer, but to serving the matrices.

6. Further reading:

Cross, N. (2000), *Engineering design methods: strategies for product design*, 3rd edition, Chichester, UK: Wiley. [Chapter 8 gives a brief overview with examples]

Cohen, L. (1995), *Quality function deployment: how to make QFD work for you*, Reading, MA: Addison-Wesley. [A definitive guide]


CORRELATIONS
STRONG POSITIVE
POSITIVE
NEGATIVE
STRONG NEGATIVE
RELATIONSHIPS
STRONG (9)
MODERATE (3)
WEAK (1)
PRIORITY


CORRELATIONS
STRONG POSITIVE
POSITIVE
NEGATIVE
STRONG NEGATIVE
RELATIONSHIPS
STRONG (9)
MODERATE (3)
WEAK (1)
PRIORITY