# Defining Specifications for Custom Products: A Multi-Attribute Negotiation Approach

S.L. Chen[1], M.M. Tseng[1] (1)

[1]Advanced Manufacturing Institute, Hong Kong University of Science and Technology, Hong Kong

**Abstract**

Defining product specifications so that customers' needs could be well matched with suppliers' capabilities is a very challenging task in custom product design. This research links technical requirements on product design with product management issues and formulates specification definition as multi-attribute negotiation. Specification negotiation is modeled as an iterative process of give and take in search of a mutually acceptable configuration. A negotiation support system is developed based on a product configuration system to identify among the multiple attributes what to give, what to take, and by how much so as to facilitate specification negotiation operationally.

**Keywords**:

Design, Customization, Specification

## 1 INTRODUCTION

In recent years, rapid growth in custom products has become a trend in many industries. Typically, custom products are designed and developed by a supplier specifically for a customer based on agreed specifications (specs). Specs are of critical importance for both customers and suppliers in economic terms. Specs are direct expression of customers' needs hence an indicator of the value of the product that customers are willing to pay; and since specs serve as a legal contract to guide and bind suppliers' operations including design, development, production, delivery etc., specs indicate the cost as well as profitability of the product to suppliers.

Defining specs for complex products is a very challenging task. Firstly, specs contain multiple attributes and there are complex interrelationships between different attributes. Secondly, specs involve multiple parties with diverse interests and functional specialties both within and across firm boundaries. Thirdly, specs have complex implications on commercial terms like price, order quantity, delivery lead time etc. Consequently, complexity and conflicts abound concerning spec definition. A noted problem in practice is that customers spend a lot of time and efforts to compile a set of specs, which, however, are often found by suppliers to be inconsistent or infeasible after painstaking checking. Customers and suppliers have to go through laborious back-and-forth iterations before agreement can be reached [1]. The situation is further compounded by the so called 'customization-responsiveness squeeze' [2]. Basically, suppliers not only need to provide the right customization but also need to provide it responsively. The time pressure considerably aggravates the difficulty of spec definition.

In response, manufacturers are moving strategically into collaboration on spec definition in hope that customers' needs could be well matched with suppliers' capabilities over the specs so that customers' value and suppliers' profitability can be achieved simultaneously. One approach is to have engineers from both sides work together on spec definition. However, in many cases, this is not a justifiable option because it consumes enormous engineering resources. Furthermore, since engineers are technology oriented, sound technical solutions may not make sound economic sense. Another approach is to include spec definition in business negotiation, which is usually carried out between the Purchasing from the customer and the Sales from the supplier. But Purchasing and Sales personnel normally lack in up-to-date technical details of the product. Consequently, they either rely on experience, which often results in over promising or being too conservative when making commitments over specs; or they go back to engineers for help, which consumes time and resources and may turn away customers due to the customization-responsiveness squeeze.

The key problem with current approaches for collaborative spec definition lies in the loose integration of product knowledge across different functional departments in both the customer and supplier organizations. Recent advances in engineering design, particularly design by configuration, open up new possibilities to handle the spec definition problem. Meanwhile, there's been significant progress in negotiation theory and its application in engineering design [3]. This research builds on top of these advances and attempts to find a systematic way to facilitate spec definition in custom product design by organizing product knowledge from both the engineering and the business domains into a common framework. Both specs and the associated commercial terms are treated as attributes to be negotiated and spec definition is formulated as multi-attribute negotiation. Under this formulation, a spec negotiation support system is developed based on a product configuration system.

## 2 SPEC DEFINITION IN CUSTOM PRODUCT DESIGN

The critical importance of spec definition on the success of product design has been well recognized by researchers in engineering design community. According to Darlington and Culley [4], spec definition is a process of capture and transformation of customers' needs into a representation that design can follow. Using Suh's [5] axiomatic design model, spec definition is equivalent to the mapping of functional requirements (FRs) onto design parameters (DPs). Other models and methods include quality function deployment (QFD), Kano model, conjoint analysis etc. [6].

Within the context of custom product design, a major limitation of these models is that spec definition is taken as a unidirectional, translational process instead of a bidirectional, interactive process. Consequently, suppliers' capabilities are not proactively leveraged to match with customers' needs.

Custom products are essentially new products developed for specific customers, but they differ in many ways from the new products in traditional sense. Generally speaking, custom products often do not involve drastic changes in design; instead, sharing commonality and reusing previous design knowledge are encouraged. Well recognized design methodologies for custom products include modular design, product platform based design, product family oriented design etc. [7] [8]. In parallel, there's has been advances in areas like product modeling, product data management, and software configuration etc. The convergence of these developments leads to product design by configuration, which has been recognized as a strategy to address the customization-responsiveness squeeze [9].

With a configurator, product design is reduced to selecting and arranging combinations of predefined components to satisfy given specs [10]. In addition, a configurator provides a means to centralize segmented product knowledge within different functional departments. Consequently, customers' requests can be responded to with quick feedback, which facilitates spec definition with efficient communication [9].

However, most configurators are essentially coded with existing product knowledge. Given the heterogeneity of customers' requirements, the complexity of the product and the rapid changes in product technology, difficulties in maintenance become a major hindrance for the efficiency and effectiveness of configurators if specs are defined in a unidirectional way without proactively taking suppliers' capabilities into consideration. Negotiation, the natural discourse of give and take between buyers and sellers, provides a basis of interaction for both sides to acquire information and to deal with conflicts. This research formulates spec definition in custom product design as a negotiation process.

## 3 A GENERAL FRAMEWORK

Raiffa et al. [11] defined negotiation as a process of joint decision making, which entails joint consequences, or payoffs, for each individual. Based on this definition, there are two elements for a negotiation to take place: first, a channel for communication through which decisions can be made jointly; second, a mechanism for each individual to evaluate the consequences (or payoffs) of the joint decision so that alternatives can be compared and negotiation can move towards predefined objectives. Based on this proposition, a general framework for spec negotiation is developed as in Figure 1.

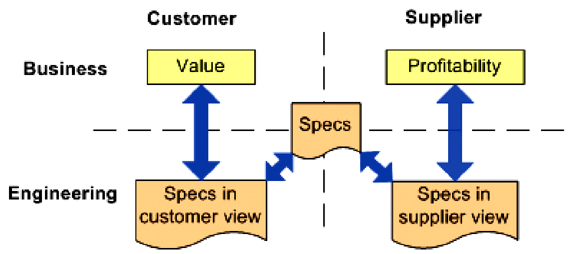


Figure 1: A general framework for spec negotiation

At the core of the framework are the specs to be negotiated. In the engineering domain, there are two versions of specs, one from customers' view and the other from suppliers' view, because customers and suppliers are usually situated in different knowledge domains, they view specs from different perspectives and use different 'languages' when describing the same specs. The mappings in between establish a channel for communication. Evaluation of joint decisions (i.e. specs) goes beyond engineering and falls into the business domain. Generally speaking, customers aim to maximize the value while suppliers aim to maximize profitability. Value and profitability are usually calculated with variables such as price (cost), quantity, lead time etc. The values of these variables hinge on the product to be designed, which is described with specs. Tracing the dependencies, we set the implications of the specs on (customers') value and (suppliers') profitability as the other two pillars in the spec negotiation framework.

In general, this framework bridges the technical aspects with product management issues, customers' view with suppliers' view concerning spec definition. Under this framework, specs as well as delivery variables (e.g. price, quantity, lead time etc.) are treated as attributes to be negotiated and spec definition is formulated as multi-attribute negotiation. Under this formulation, there are generally two approaches: one is customer oriented, i.e. maximizing customers' value within suppliers' affordability; the other is supplier oriented, i.e. maximizing suppliers' profitability on the condition that customers' requirements are fulfilled.

The customer oriented approach is in alignment with the popular notion of customer centric enterprises. Following this approach, Bichler et al. [12] developed a decision support system for the Representation and Evaluation of Configurable Offers (RECO). The backbone of this approach lies in utility theory [13]. Utility indicates the overall value of the product perceived by the customer. However, within the context of spec definition for custom products where the products are complex, uncertainty is high, and requirement on accuracy is high, utility is hard to implement in practice because of its subjectivity.

This paper follows the supplier oriented approach in consideration that in many sourcing situations customers are not looking for products with the best possible performance but products with good-enough performance and the lowest price, and price depends mainly on cost and cost links with suppliers' capabilities and has been thoroughly studied and well formulated.

## 4 MULTI-ATTRIBUTE NEGOTIATION

Negotiation is about exchange for mutual benefits. Different parties often have different values attached to the same attributes because of different perspectives. Such discrepancy of valuation makes exchange attractive if and only if both parties think what they take is more valuable than what they give. In this sense, discrepancy of valuation makes win-win solutions possible and means opportunity to reach agreement. When there're multiple attributes, discrepancy of valuation will emerge on multiple dimensions, which means multiple degrees of freedom in negotiation and high potential of reaching agreement.

However, negotiation is also about conflict resolution. Each party seeks to maximize his/her own benefits from the exchange but this often conflicts with the other side's benefit maximization pursuit. When there're multiple attributes in negotiation, conflicts will emerge on multiple dimensions, which will obscure the intrinsic valuation of attributes and obstruct reaching agreement.

In practice, spec definition as multi-attribute negotiation is an ill-structured problem with incomplete information. It requires a process of discovery to make explicit the valuation discrepancies and exploration of technical capabilities. Such a process can be taken as a dynamic process of revelation and concession making with collaborative problem solving in finding creative solutions. In the case of spec negotiation for custom products, the attributes for negotiation are usually agreed upon in advance. Thus, spec negotiation can be modeled as a concession making process of give and take over the values of attributes. Then a key to explore and exploit the valuation discrepancies over multiple attributes is being able to identify among the multiple attributes what to give, what to take, and by how much.

## 5 A SPEC NEGOTIATION SUPPORT SYSTEM

This paper takes a prescriptive [11] approach from supplies' perspective and develops a spec negotiation support system (SNSS) based on a product configurator. SNSS consists of two subsystems: an optimizer to configure the optimal product given the specs, and a recommender to suggest what and how to negotiate. Basically, the optimizer aims to find the optimal solution within a given solution space; while the recommender aims to reconfigure the boundary of the solution space so as to include better solutions.

### 5.1 The optimizer

The optimizer is adapted from RECO [12] but takes the supplier oriented approach and takes into consideration of product design issues.

*Configuration representation*

A product configuration is modeled as a combination of components arranged according to certain product architecture. For each component, there could be several options. For example, a PC is composed of components like hard disk, CPU etc.; for the hard disk, the size could be 20GB, 40GB etc. In this section, a single product family is considered, i.e. the product architecture is fixed and the task of configuration is to select the right components. The components for a configuration can be represented as a set: $\{v_{ij} \mid i = 1,...,I; j = 1,...,J; v_{ij} \in V_i\}$ where $V_i$ represents the options for the *i*th component on the product architecture, $v_{ij}$ represents the *j*th option for the *i*th component.

*Configuration approach*

Configuration is approached as a rule based constraint satisfaction problem [10], i.e. selecting and arranging components to satisfy certain constraints. And configuration does not start from scratch but from a base configuration, which is defined as the best previous model within the same product family. Configuration is to replace or update some components of the base configuration according to customers' specific requirements.

*The objective function*

The unit cost of the configuration is to be minimized. It can be calculated with the unit cost of the base configuration plus cost deviation caused by new components selected:

$$C = C_b(q,l) + \sum_i \sum_j m_{ij} x_{ij} \qquad (1)$$

where $C_b(q,l)$ is the unit cost of the base configuration updated in terms of order quantity ($q$) and delivery lead time ($l$), $x_{ij} \in \{1,0\}$ indicates whether $v_{ij}$ is selected or not, $m_{ij}$ represents the cost markup from the base configuration if $v_{ij}$ is selected.

*Constraints representation*

Basically, there are three categories of constraints:

1. Configuration rules, which are inherent to the configurator and fixed given current technology, e.g. compatibility between components.

2. Commercial terms, e.g. discount can be granted to some components if quantity reaches a certain level;

3. Constraints imposed by specs. A well established axiomatic design model [5] is assumed to map FRs within the specs onto DPs.

These constraints can be first described as general logical expressions, then transformed into conjunctive normal form (CNF) representation, and further converted into mathematical inequalities [12].

*Problem formulation*

Product configuration is formulated as an integer programming (IP) problem as in Table 1.

Given: A base configuration

Find: A new configuration

Satisfy:

Configuration rules;

Commercial terms;

Constraints imposed by specs.

Objective:

Minimize the cost of the new configuration

Table 1: Problem formulation for the optimizer

### 5.2 The recommender

When configuring a product, constraints imposed by configuration rules are fixed, or hard, while constraints imposed by specs and commercial terms are negotiable, or soft. Relaxing soft constraints means enlarging the pool of components for selection, which implies possibility of better solutions for the optimizer. With an optimal configuration identified with the optimizer, there are generally two scenarios for the soft constraints:

1. Some constraints are tight while some are slack.

2. All constraints are tight.

In scenario 1, relaxing tight constraints helps to decrease the lower bound of cost, while tightening slack constraints within slackness does not increase it. Correspondingly, in spec negotiation, the attributes linked with tight constraints need to be 'taken' while the attributes linked with slack constraints can be 'given'. Figure 2 illustrates the take and give scheme in this scenario with 2 attributes. $A_1$ and $A_2$ axes represent the range of values for attribute 1 and 2 respectively. Cost is modeled as a convex function inversely related to $A_1$ and $A_2$. Constraints on $A_1$ and $A_2$ are slack and tight respectively. By giving on $A_1$ from $a_1$ to $a_1$', while taking on $A_2$ from $a_2$ to $a_2$', the optimal solution moves in the direction of decreasing cost.

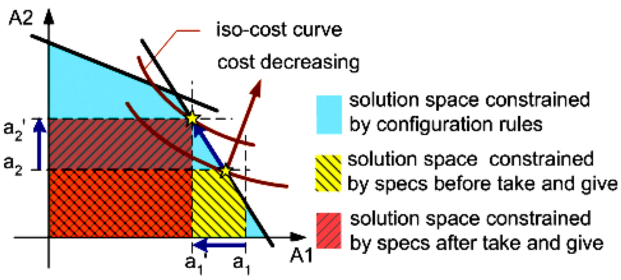


Figure 2: Take and give in scenario 1

When multiple soft constraints are simultaneously tight, a checklist based on cost markup is devised to order the constraint relaxation priority. For example, constraints on both CPU and hard disk are tight, but since CPU costs more so it ranks higher than hard disk on the checklist. Such a checklist may not be optimal, but it provides a pragmatic way to handle the problem.

The criterion for take is to relax the constraint until a new configuration with lower cost is included into the solution space, otherwise leave the constraint intact and go on to the next tight constraint on the checklist. The criterion for give is that compromise should be made within slackness so as not to affect the optimal solution.

The take and give process described above results in all constraints becoming tight, i.e. scenario 2, which is an equilibrium state with no incentive to compromise over any single attribute. Breaking the equilibrium could lead to another equilibrium (Figure 3(a)), or return to a state in scenario 1 (Figure 3(b)).

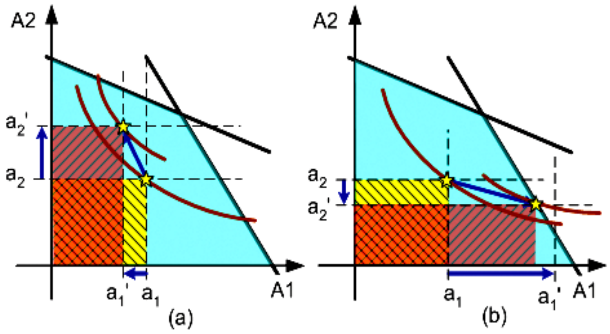


Figure 3: Take and give in scenario 2

The recommender works in between of the two scenarios with cost as the guiding criterion. There's not an explicit stopping criterion. The process stops when agreement is reached or declared a failure after certain efforts. The algorithm for the recommender is summarized in Table 2:

1. Identify slack and tight constraints;
2. Take (give) over the attributes corresponding to tight (slack) constraints;
3. Repeat step 2 until equilibrium is reached;
4. Break away from equilibrium, if some constraints are slack, return to step 2; otherwise, repeat step 4.

Table 2: The algorithm for the recommender

### 5.3 System architecture

Architecture for the implementation of SNSS is developed. The architecture integrates the parties involved in spec definition and takes into consideration of product architecture identification because suppliers normally offer products that belong to different product families. Specs are classified into primary specs and secondary specs, which link to product architecture and technical details respectively. The data flow within the system architecture is structured as in figure 4.

## 6 SUMMARY

The rapid increase in custom products in recent years calls for a systematic approach to expedite agreement making over spec definition between customers and suppliers. This paper links the technical requirements on product design with product management issues and formulates spec definition as multi-attribute negotiation. The give and take negotiation process is modeled as an iterative process of exploring and exploiting the valuation discrepancies between customers and suppliers over the attributes. Such a process provides a basis to supplement product configuration by tracking the partial information unfolded during negotiation. Development of a spec negotiation support system (SNSS) from suppliers' perspective is also reported.

## 7 ACKNOWLEDGEMENT

The authors gratefully acknowledge the financial support of ITC (Innovation and Technology Commission) of HKSAR under grant UIM/86 and Esquel Enterprises Ltd.

## 8 REFERENCES

[1] Nellore, R., 2001, Managing Buyer-Supplier Relations: the Winning Edge through Specification Management, London, Routledge.

[2] McCutcheon, D. M., et al., 1994, The Customization-Responsiveness Squeeze, Sloan Management Review 35/2: 89-99.

[3] CIRP ECN (engineering as collaborative negotiation) website, http://mspde.usc.edu/ecn/.

[4] Darlington, M.J., Culley, S.J., 2002, Current Research in the Engineering Design Requirement, Proceedings of the Institution of Mechanical Engineers 216/3: 375-388.

[5] Suh, N.P., 1990, The Principles of Design, Oxford University Press, New York, NY.

[6] Dahan, E., Hauser, J., 2002, Product Development: Managing a Dispersed Process, Handbook of Marketing, London, Thousand Oaks: SAGE.

[7] Tseng, M.M., Jiao, J., 1996, Design for Mass Customization, Annals of the CIRP, 45/1:153-156.

[8] Tseng, M.M., Kjellberg, T., Lu, S.C.-Y., 2003, Design in the New e-Commerce Era, Annals of the CIRP, 52/2:509-520.

[9] Salvador, F., Forza, C., 2004, Configuring Products to Address the Customization-Responsiveness Squeeze: A Survey of Management Issues and Opportunities, Int. J. Production Economics, 91:273-291.

[10] Sabin, D., Weigel, R., 1998, Product Configuration Frameworks - A Survey, IEEE Intelligent Systems, 13/4: 42-49.

[11] Raiffa, H., Richardson, J., Metcalfe, D., 2003, Negotiation Analysis: The Science and Art of Collaborative Decision Making, Belknap Press.

[12] Bichler, M., Kalagnanam, J.R., Lee, H.S., 2002, RECO: Representation and Evaluation of Configurable Offers, IBM Research Report RC22288.

[13] Thurston, D.L, 1991, A Formal Method for Subjective Design Evaluation with Multiple Attributes, Research in Engineering Design, 3/2:105-122.

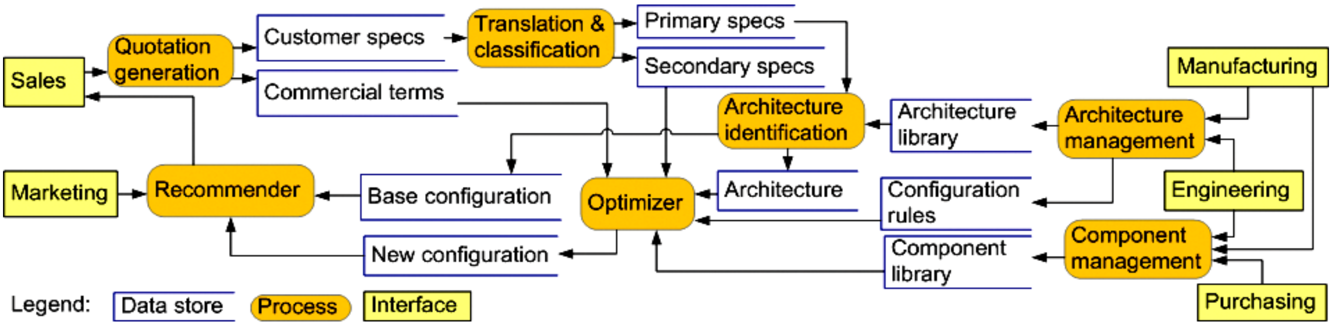


Figure 4: Data flow within the SNSS architecture