

Recent developments in the configuration of multiple-variant products: Application orientation and vagueness in customer requirements

Stephan Schwarze and Paul Schönsleben
Institute of Industrial Engineering and Management (BWI)
Swiss Federal Institute of Technology ETH Zürich
Zürichbergstrasse 18, CH-8028 Zürich, Switzerland
Tel: +41 1 632 05 10 Fax: +41 1 632 10 40
schoensleben@bwi.bepi.ethz.ch

Abstract

To face the problems of mass production the configuration of multiple-variant products becomes more and more important. In this paper an approach is presented for improving the configuration process. Therefore the process is split into three stages. The important stage to make configuration services more customer friendly is the Specification Mapping. As a basis for the configuration process a common configuration data model exists which contains functional application-oriented knowledge in addition to conventional technical data. Furthermore vague knowledge is integrated into a product configuration, and finally a short realization guideline is presented.

Keywords

Configuration of Multiple-Variant Products, Functional Knowledge, Product Modeling, Customer Orientation, IMS

1. INTRODUCTION

In recent years, customers have become more critical and choosy. Individual customers have specific requirements for offered products. Customer orientation is becoming an important phrase for many companies; the customer must be the focus of all activities within the company. As a result, one-of-a-kind production has emerged as a new trend. Compared to mass production, the number of product variants increases drastically. Furthermore, the competition between companies increases quickly. Speed and flexibility in design and manufacturing are very important for the success of companies.

A product configurator that can use customer requirements directly as input is a necessary tool for any company that wants to be able to offer engineered-to-order or assembled-to-order products to individual customers. The necessity of a product configurator for customer-oriented companies becomes obvious if they ask themselves the following questions (Fazio, 1994):

What do you do when your product expert suddenly leaves the company?

What happens if you have more configuration problems to solve than your product experts are able to solve?

Existing configurators still have many deficiencies. Therefore, an improved customer-oriented configuration approach will benefit many companies.

Before actually manufacturing a product, it is necessary to configure it correctly. During the configuration process the product is tailored to meet the customer's needs. It is determined, for example, what the detailed components of a product are, how the final product is assembled and what processes are necessary to manufacture the product. To configure a product variant correctly, company-internal knowledge (from designers and engineers) is essential as is the customer's functional application-oriented knowledge. This functional knowledge can include the intended use of the product, its desired functionality or its features. The company's knowledge mainly covers the technical and structural aspects of the products, e.g., technological know-how, production facilities, knowledge about parts and components, connecting possibilities, etc.

Ideally, the initial input for product configuration stems from the customer's functional requirements. The customer gives a functional specification covering the desired features. This specification has to be satisfied by the configured product. Frequent problems inherent to such functional requirements are that

- the customer's input is incomplete, and
- the customer's requirements are vague and uncertain.

Today's configuration approaches mostly ask for a detailed (technical) product specification as input. Customers must be familiar with the technological details of

the desired product and often have to specify technical parameter values. However, frequently the customer (or the salesperson) does not have sufficient knowledge to give such a precise specification and can only provide his specification as qualitative, functional requirements.

This functional (possibly vague or incomplete) specification has to be transformed into appropriate technical parameter values. Here, individual preferences and misinterpretations have to be avoided, and even for complex products all input knowledge has to be used. Individuals mostly cannot provide a reliable configuration service manually. The solution is a software-supported product configuration.

The first stage in product configuration, the Specification Mapping, is a central point for the integration of customer knowledge into the configuration process (see Figure 1). In this stage the functional, application-oriented input is transformed into technical or structural knowledge.

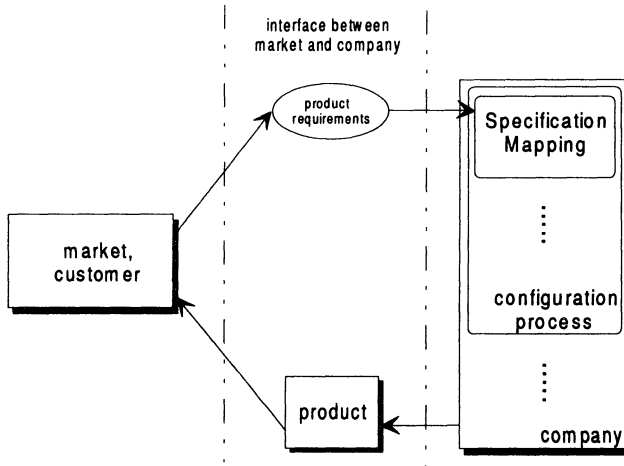


Figure 1 Product as interface between market and company

Configuration problems exist not only for industrial companies, but also for insurance companies and banks that recognize a high individualization potential in developing customer specific services (products) (Link and Hildebrand, 1995). Therefore, a method of customer-oriented product configuration can also be applied to products in the service sector (service products).

2. THE PRODUCT CONFIGURATION PROCESS

The main idea that is presented in the following is the possibility of separating the process of product configuration into three stages (Schwarze 1994). This approach avoids the problem of solving separate subtasks at the same time. Each stage concentrates on a special phase within the configuration process. Another argument for several stages is that the feature of rule-based systems does not allow for a clear separation between knowledge and control. With rules, not only is a relationship between components expressed, a sequence of actions is also determined (Weigel and Faltings, 1994). Thus the maintenance becomes easier if subtasks are treated separately and the respective knowledge is kept in different knowledge bases.

The three stages in a product configuration process are

1. Specification Mapping,
2. Technical Configuration and
3. Choice and Optimization.

Figure 2 shows this new concept. A bicycle serves as an example of a product in the figure. The basis for the three stages is one common configuration data model (see Section 3) in which any information that is relevant in any configuration stage is modeled. In practice, the information in this common configuration data model can be kept in several knowledge bases.

To simplify the configuration process, specifications that can be changed should be separated from those that cannot. These firm decisions are preferably made in the Specification Mapping stage.

A special situation exists for simply-structured products in the service sector, like bank credits. Their structural constraints are mostly very simple whereas customer requirements can be complicated and extensive. Therefore such products can be configured in a single stage that is very similar to the Specification Mapping stage in an engineering product configuration process. The principles are the same as those discussed in Section 4.

3. CONFIGURATION DATA MODEL

Product information can be split roughly into functions and physical principles (Chen and Menq, 1992). Pure physical models are very popular, but they do not consider functional product aspects that are only indirectly product-related but rather application-oriented. Such functional requirements are types of abstract information.

A precondition to developing a modern configuration approach is that product functionalities can be modeled. This task must cover two aspects:

- The functionality², in particular the application, of a product must be modeled.

- The relation between a product's functionality and its structure must be considered.

Sembugamoorthy and Chandrasekaran (1992) require the task of functional modeling to support the understanding of complex devices. Representing a device's functionality

- requires a description of "what a device is", achieved by recursively decomposing it into subdevices linked with *component of* relations, and
- a description of "how a device works" by identifying functions of subdevices and describing their behaviors.

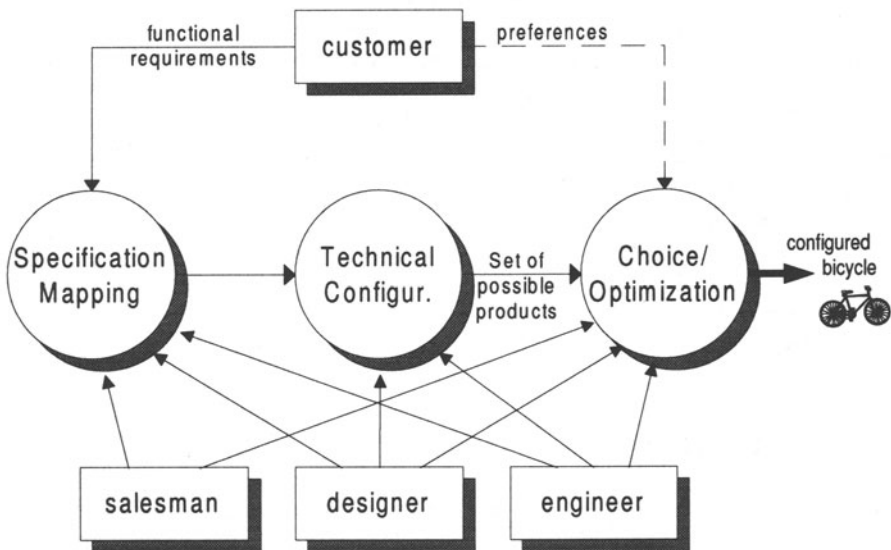


Figure 2 Three stages in the configuration process

Functional descriptions are naturally modular, which means that a device can be replaced by a totally different device that accomplishes the same functions. Thus, a graph that is a hierarchy based on functional properties can be drawn. For a company, the representation of functions in the configuration data model leads to the following advantages (Chawla and Sangal, 1992):

*Functionality in this context does not mean the technical functions of a product but the application- or requirement-oriented view. Its purpose is to integrate the customer's view of a product into the configuration data model.

- It guides the task of searching for a certain product.
- It can be used for decomposing problems into finer details.
- It supports finding all parts that satisfy a particular function.
- It enables one to formulate restrictions that have to be fulfilled in order to carry out a function.

For a configuration problem, this means that a functional decomposition can consist either of a function being decomposed into other functions, or components that carry out the function being determined. Instead of integrating variables that represent functions and variables that represent components as suggested in (Mittal, 1990), it is better to distinguish between a functional and a structural (or technical) hierarchical decomposition of a product. Requirements appear at the leaves of a functional decomposition tree. They are not decomposed into subrequirements, but they are put in relation with components by formulating rules.

To fulfil the requirement of modeling a product's functionality, Schönsleben's product model (Schönsleben, 1994) has been extended to meet the requirements for a configuration data model which can be used as a base for any of the three configuration stages. Details about the configuration data model and the new classes are given in (Schwarze, 1996). An earlier version of this extension was developed through the research project IMS/GNOSIS (IMS/Gnosis, 1995; Gosset and Massotte, 1994; Schönsleben and Schwarze, 1995) and was guided by industrial requirements.

4. SPECIFICATION MAPPING

The Specification Mapping is the first stage in product configuration. The goal is to transform the functional knowledge provided by the customer to a structurally concrete and precise product specification [KuSz92]. *External* knowledge from the customer is transformed to meet company-*internal* requirements. Internally, technical, production-oriented knowledge is most important. The Specification Mapping covers the functional aspects of product configuration and is usually done by salespeople in companies. Very often it is neglected, and this neglect can have a negative influence on the final configuration result because an incorrect Specification Mapping can never lead to correct product configuration.

Major problems in this stage are

- that uncertain, rough and tacit knowledge has to be used and
- that the input data does not have a fixed format but varies widely in appearance.

Equivalent information can be given by different customers in different formats. Nevertheless, this input has to be interpreted correctly so that the same product

specification is created. The data that is treated can be of unsure origin. For example, a customer often gives information about the performance, the area of use or the quality of the required product: he describes the required functionality of the product, and this functional description usually is more application-oriented than product-related. Therefore, this stage is the most risky part of a configuration system.

It is a demanding task to get this Specification Mapping done automatically. Expert system techniques are an appropriate means. But if the input information and the input format should remain free, the automation of the Specification Mapping can be very problematic. Even in a rather small domain, the expert-knowledge database has to be huge to cover all possible input values (Schönsleben, 1988; Schönsleben and Oldenkott, 1992, van Veen, 1992; van Veen and Wortman, 1992).

If the information about the required product is not sufficient, it has to be completed. The customer has to provide more input and must describe the product with more details, possibly by answering concrete questions. This is also necessary to limit the number of possible configurations.

It is desirable that the customer gets as much freedom as possible when specifying the requirements for a product. He should be free in what kind of information he provides about the desired product: functional information, technical information, structural information, etc.

In general, two different situations can be distinguished when a customer specifies his requirements (see Figure 3).

1. The customer has the ability to specify the technical parameters directly. In this case it is possible to abstain from a Specification Mapping.
2. The customer has knowledge only about the functionality of the required product. Then the Specification Mapping is necessary to transform this knowledge into a technical product specification.

Obviously, the procedure of product configuration is much easier in the first situation. In this case the risk of errors is much lower because concrete parameters and features are specified directly by the customer. But the first situation only exists for very competent customers.

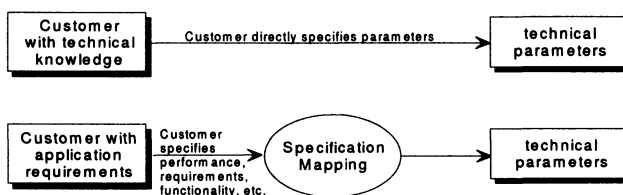


Figure 3 Necessity of the Specification Mapping

5. CONSIDERATION OF VAGUENESS IN A PRODUCT CONFIGURATION

Frequently, vague knowledge appears in a knowledge source for a configuration system. Mainly two points where the problem of uncertainty becomes obvious exist:

- Often the customer can only provide a rough and vague description of what he wants.
- The configuration or sales experts that have to interpret the functional requirements are used to formulating their knowledge by using natural language terms.

Both aspects are described in more detail in the following and have to be considered in a modern configuration system.

5.1 Vague Product Descriptions

When specifying the desired product, a customer very often uses vague, fuzzy terms (Chen and Menq, 1992; Bode et al., 1994). He describes the functionality of a product by using words like “often”, “sometimes”, “never”, etc., or qualitative input like “big”, “cheap”, “fast”, etc. This is the easiest way for a customer to specify what he wants because he can use the same words as in a shop.

Therefore, a requirement for a configuration system is that such blurred, colloquial terms are allowed as input information. This makes the systems much more user-friendly, and the acceptance for using a configurator will be higher. The customer can use the same notation for specifying a product that he has always used. The other advantage is that no knowledge will be lost. The user provides the product specification that he really has in mind directly. If he had to transform his own colloquial product specification into a formalized description that could be understood by the system, a lot of information might get lost during this formalization. Furthermore, he can provide the information on his own and does not need a knowledge engineer to transform it correctly.

5.2 Vague Expert Knowledge

In an expert system for executing a configuration, most of the knowledge for interpreting the functional product specification and creating a technical specification will come from configuration experts or from sales or marketing people (Popp, 1994). These people are used to formulating their knowledge in colloquial terms. For example they formulate rules like

if a bike is *often* used in races
then it *should* have *thin* tires,

where the terms "often", "should" and "thin" are natural language expressions.

Again, a configuration system should accomodate the use of such terms. The objective is to make expert heuristics usable for machines. The consequences are the same as given above: the acceptance of the system will be much higher because experts can directly use natural language.

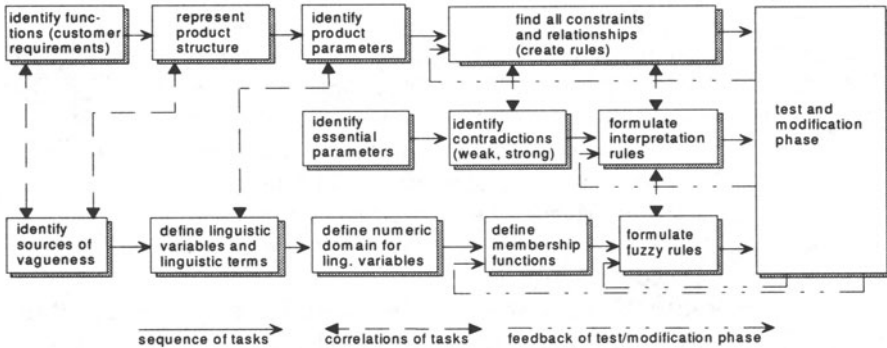


Figure 4 Task structure for realizing a Specification Mapping system

Little information will be lost due to formalizing rules into another notation. The configuration expert should be able to implement the rules on his own without being dependent on a knowledge engineer.

A further advantage is that the knowledge is immediately understandable to new employees. They do not have to interpret an unknown notation. Finally, it is easier within the system to interpret the customer's requirements if identical (colloquial) terms are used both for data input and in the rules.

Both sources of vague knowledge are linguistic uncertainty. Therefore fuzzy logic is an adequate methodology to handle this vagueness. A concrete approach for integrating fuzzy logic into a configuration system is described in (Schwarze, 1996).

6. REALIZATION

Figure 4 shows the structure of tasks that have to be executed for an implementation (Schwarze, 1996). The separate tasks stem from aspects mentioned in the previous sections. Obviously, some tasks cannot be undertaken independently, but correlate with other tasks as indicated below.

Before the Specification Mapping system can be used in practice, it has to be tested for a set of configuration examples. During this *test and modification phase* the system has to be modified. In particular, rules must be modified, extended or newly created. For the fuzzy terms of linguistic variables, membership functions may have to be changed.

7. CONCLUSION

7.1 Summary

An approach for making the product configuration process more customer-friendly is presented. The entire product configuration process is split into three stages. The Specification Mapping stage maps functional requirements to structural data, the Technical Configuration stage determines a set of final products and checks consistency, and the Optimization stage decides on an optimal product.

Since the Specification Mapping is neglected in most existing configuration systems, a configuration data model in which functional, application-oriented knowledge is integrated has been developed.

In the Specification Mapping stage, vague knowledge appears frequently in individual customer requirements as well as in product expert knowledge. Since both types of uncertainty are linguistic, fuzzy logic is used for integrating vague knowledge into the Specification Mapping system. Therefore, vague knowledge is also integrated into the configuration data model, and linguistic variables with associated linguistic terms have to be defined.

The Specification Mapping approach is applicable for service products as well as for engineering products. Since for some service products, for example an individual bank credit, the product structure is simpler but the functional requirements are more complex, the Specification Mapping stage is even more important. Furthermore, the input to service product specifications is commonly vague knowledge.

7.2 Outlook

Important future work for the Specification Mapping system itself is

- to improve the user interface,
- to merge the system with the other configuration stages and
- to integrate the system into existing information systems in a company.

The integration of a Specification Mapping system into a production planning and control system (PPC system) and a product data management system (PDM system) is particularly important because these systems capture, record and manage all product-related data and documents. With this integration the

maintenance, not only of structural product knowledge but also of the configuration knowledge described in the configuration data model can be guaranteed.

8. REFERENCES

- Bode, Jürgen; Xinli, Zhao; Shouju, Ren: Quality Function Deployment: Extensions and fuzzy concepts in an integrated CSCW environment, in: Proceedings of the International Conference on Data and Knowledge Systems for Manufacturing and Engineering, Hong Kong, May 2-4, 1994, p.246-251.
- Chen, Baosheng; Menq, Chia-Hsiang: Initial attempts on the characterization of functional requirements of mechanical products, in: Dutta, D.; Woo, A.C.; Chandrashekar, S.; Bailey, S.; Allen, M. (eds.): Concurrent Engineering - 1992, The Winter Annual Meeting of the ASME, Anaheim, California, November 8-13, 1992, p.315-329.
- Chawla, A.; Sangal, R.: Functional Reasoning in Configuration Design, in: Gero, J.S. (ed.): Artificial Intelligence in Design '92, Kluwer Academic Publishers, Dordrecht, 1992, p.41-60.
- Fazio, Gregory J.: Getting the Facts on Rule-Based Product Configuration, in: 1994 APICS Conference Proceedings, p.E12-E13.
- Gosset, Paul; Massotte, Pierre: Knowledge Systematization and Sharing - GNOSIS (IMS) Impact on Industry, CIM Europe Conference Copenhagen, October 5th-7th, 1994.
- IMS/GNOSIS Consortium: Report on the IMS/GNOSIS test case: Configuration Systems for Knowledge Systematization, 1995.
- Link, Jörg; Hildebrand, Volker G.: EDV in Marketing und Vertrieb strategisch nutzen, in: IO Management Zeitschrift, 64(1995), No.3, p.85-89.
- Mittal, Sanjay: Hierarchical Reasoning Model for Configuration Design Problems, Research Paper at Xerox PARC, 1990.
- Popp, Heribert: Anwendungen der Fuzzy-Set-Theorie in Industrie- und Handelsbetrieben, in: Wirtschaftsinformatik, 36 (1994) 3, p.268-285.
- Sembugamoorthy, V.; Chandrasekaran, B.: Functional Representation of Devices and Compilation of Diagnostic Problem-Solving Systems, in: Artificial Intelligence in Engineering, 1992, p.47-74.
- Schönsleben, Paul: Expertensysteme als Hilfsmittel der variantenreichen Produktkonfiguration, in: Informatik Forschung und Entwicklung, No. 3, 1988, p.117-127.
- Schönsleben, Paul: Praktische Betriebsinformatik, Springer, Berlin, 1994.
- Schönsleben, Paul; Oldenkott, Hendrik: Enlarging CAD and Interfaces between PPC and CAD to respond to product configuration requirements, in: IFIP Transactions B-7, North-Holland, Amsterdam, 1992.

- Schönsleben, Paul; Schwarze, Stephan: Literature Survey: Product Configuration for Products with many variants, Research Paper at BWI/ETHZ, March 1994.
- Schönsleben, Paul; Leuzinger, Ruth: Innovative Gestaltung von Versicherungsprodukten, Gabler, Mainz, 1996.
- Schwarze, Stephan: The Procedure of Product Configuration and Handling the Configuration Knowledge, in: Burke, L. (ed.): Proceedings of the Third IERC, Atlanta, May 1994, p.220-225.
- Schwarze, Stephan: Configuration of Multiple-Variant Products: Application Orientation and Vagueness in Customer Requirements, vdf Verlag, Zurich, 1996.
- Veen, E.A. van: Modelling Product Structures by Generic Bills-of-Materials, Manufacturing Research and Technology, Vol. 13, Elsevier, Amsterdam, 1992.
- Veen, E.A. van; Wortmann, J.C.: Generative bill of material processing systems, in: Production Planning & Control, 1992, Volume 3, No. 3, p.314-326.
- Weigel, Rainer; Faltings, Boi: Constraint-based knowledge representation for configuration systems, Technical Report No. TR-94/54, Ecole Polytechnique Federale de Lausanne, Department d'Informatique, 1994.

9. BIOGRAPHY

Paul Schoensleben, 1952, studied Mathematics and Operation Research at the Swiss Federal Institute of Technology ETH in Zurich. He obtained a master's degree in 1977 and a Ph.D. degree in 1980. Several years of responsibilities for Computer and Organization departments of industrial companies followed. In parallel, he developed and introduced a PPC package, today called Expert/400. From 1983 to 1991 he has been ordinary professor for Business Informatics at the University of Neuchatel, Switzerland. Since 1991 he is ordinary professor for Industrial Engineering and Management at the Swiss Federal Institute of Technology ETH in Zurich. He is co-founder and actually vice president of ASGI/SGBW, the Swiss partner of APICS.

Stephan Schwarze, 1967, studied Computer Science at the Technical University of Karlsruhe (Germany) and Victoria University in Wellington (New Zealand). He graduated with an M.Sc. in 1992. Between 1993 and 1996 he worked at the Institute of Industrial Engineering and Management at Swiss Federal Institute of Technology (ETH) in Zurich, Switzerland. In the group of Professor Paul Schoensleben, his main research focus was on product configuration and product modeling. He received his Ph.D. in Technical Sciences from ETH Zurich in 1996. Since 1996, he works as consultant and model developer for Trilogy Development Group in Austin, Texas (US).