INTERNATIONAL DESIGN CONFERENCE - DESIGN 2002
Dubrovnik, May 14 - 17, 2002.

A KBE SYSTEM TO MANAGE THE MODULE CONFIGURATION USING THE CORPORATE KNOWLEDGE

Ferruccio Mandorli, Sergio Berti and Michele Germani

Keywords: Knowledge-based Engineering, Modularity, Product Configuration

1. Introduction

The ever-increasing competition, due to the market globalisation, forces the industries (also SME industries) to modify their design and production strategies. A key point, today, is the development of products that fulfil the individual customer needs as close as possible. The market trend, hence, is towards the "mass customization". This can ideally lead to satisfy particular customer requirements while still maintaining the efficiency and the low development costs of the mass production. However, mass customization in itself introduces new demands on firms, these include improved product development processes, flexible manufacturing planning and control systems, and closer supply chain management.

While larger organisations by their nature can afford the efforts of applying complex methodologies, small and medium enterprises need a more simple and low risk approach. The goal of our research is the identification of an approach (method and technology) for the design of product families that can be configured for the needs of SME's. The objective of the present work, in particular, is the development of a KAE system application to shorten the time of module configuration phase (instantiation) aiding the designer decision-making activity. In fact, the great advantage of modular products development can be the rapid configuration of specific solutions (variants) to satisfy every new customer requirement. The KAE technology can support such process (a rapid and simultaneous analysis of many alternatives) and permits also an easy and effective reuse of the corporate knowledge (i.e. the designer experience).

2. Related work

Considerable research attention, in the recent years, has been dedicated to modularity and product family design [Otto and Wood 2001]. The product's platform architecture definition [Gonzales-Zugasti and Otto 2000] and the application of reconfigurable product platforms, the product family configuration [Tichem et al. 1999], the analysis of commonality to manage the product variety [Martin and Ishii 1997], are different aspects under investigation in the industrial and research world.

Other fundamental works are related to modularity definition and modules identification and configuration [Stone et al. 1998], [Allen and Carlson-Skalak 1998]. In this last context, the present paper will provide an approach to allow the integration between such theories and an automatic design system.

The designers, once established the modularity of a specific production and determined the functional modules (i.e. applying the Pahl and Beitz approach), must select the best modules assembly among many alternatives. They must configure, hence, every module instance, considering the impact that

each single selection has on the final product, in terms of costs, performances, assemblability and so on. Such task, generally, is intuitively performed on the basis of the expert's personal skill.
The module configuration can be divided in two steps: the determination of module configuration knowledge and the configuration of solution (approach similar to [Tiihonen et al. 1999] for the configurable products). In the first phase, the module design rules are defined and a knowledge model is obtained that specifies the set of correct configurations of the modules. In the configuration of solution phase a module is instantiated using the predefined knowledge model to satisfy the specific requirement. Such modules instances, then, can be managed in a high level model in which are defined the assembling rules (product architecture) to form the final product. In a previous work [Berti et al. 2001] such process has been supported with a low-cost CAD system linked with spreadsheet tables. The process has been applied in a simple test case (a low number of modules managed). But when the information grows (a complex machine with many hundreds functional modules), the system is hardly manageable.
In this work we have tried to overcome such problem developing a knowledge-based application to define the module configuration knowledge and to manage the specific configuration.

3. Method

Our approach to product family design can be enclosed in the bottom-up methodology [Simpson 2001], since we study a consolidated productive situation (in the specific case in collaboration with an industrial partner, Biesse S.p.a.) to standardise functional groups to improve economies of scale. In this scenario the primary goal is the module identification. On the basis of such analysis is possible to determine the product platform for the future production. The last step is the variants definition through the assembly of "intelligent" modules that encapsulate the configuration rules and the design parameters.

3.1 KBE systems for modular products

It is widely accepted that the design activity can be subdivided into four phases: specifications definition, conceptual design, embodiment design and detail design. The final result of the process is the generation of all documentation required for the production process (drawings, BOM, etc.). If the detailed geometry becomes important close to the end of the design process, in intermediate steps the reasoning of designer involves functions, sketches, schemas, simplified geometry. Such infomation implies an high level of abstraction. The traditional design systems (CAD systems) support a shape-oriented philosophy, but neglect the designer's way of thinking. The alternative is the use of knowledge-intensive design systems, in which shape is just one of the types of data the system must be able to manage. Knowledge-based engineering systems do not express designs with specific data instances as conventional computer-aided systems do, but with sets of rules that enable the design to apply to large classes of similar parts. The rules can be selectively executed to allow the design problems to be resolved [Sainter et al. 2000]. KBE systems can be also easily integrated (or interfaced) with a geometric modeller, hence, also the design documentation is easily manageable. Also, once the product knowledge has been acquired, it can be stored in a manner that allows product knowledge updates, reuse and sharing. Ideally, the corporate knowledge can be represented in the knowledge base implemented.
From practical point of view, the current technology available to develop a KBE applications are development environments (KAE shells) that provide functionality to handle a hierarchical product model structure including all types of data (shape, design rules, etc.). The main tools provided by typical KAE shells are: an object oriented language to define the product model, a tracer to support the debugging activity, a set of drivers to communicate with external databases, and an interactive environment to define a user-friendly interface to access the developed model [Mandorli and Bordegoni 2000].
The use of the object-oriented representation of products and processes is the fundamental characteristic that allow the correspondence between modular products and a KBE application. Every module can be considered an "object" that encapsulates the procedures and the rules for a self-configuration on the basis of specific design requirements. Such approach is discussed in the next paragraph.

3.2 Method framework and system architecture

The first problem in the methodology is to define a module. We adopt the following definition: "*an integral physical product sub-structure that has a one-to-one correspondence with a subset of a product's functional model*" [Ulrich and Tung 1991]. The determination of the functional structure of the product becomes the essential step to start the analysis.

The method proposed is based on a three-level model (corresponding to the design phases) with different levels of abstraction for the product representation [Bordegoni and Mandorli 2001].

Level 1: functional representation (task: determination of functional structure).

Level 2: modular representation (task: mapping of functions in well-defined modules based ona module library).

Level 3: modules instances (task: configuration of modules to arrive at the definitive detailed geometry).

Every level manages a different kind of information and it needs a different design support system, but such systems have to inherit the decisions and the structure defined at the upper level. Today is not available a technology that is able to manage such different data in a single product model. Our efforts, hence, have been concentrated on the third level. The other tasks have been executed on the basis of design teal experience.

The application of such approach has constrained to a preliminary analysis for the specific product typology. The task can be accomplished in four phases:

- analysis of current production (machines typologies and their main tasks) to identify the product variety;
- identification of modules by the functional models analysis of the machines;
- classification of modules to group the redundant functionalities;
- definition of information (design rules, interfaces, etc.) needed to configure the module instance and related components.

Such steps allow to define the knowledge base to implement a KBE application.

The three-level model proposed has been investigated only at the third level with the aim of automate the module instantiation phase. In the future, the idea is the development of a system able to manage simultaneously the data in all the levels.

The architecture of system developed is schematically illustrated in the Figure 1.

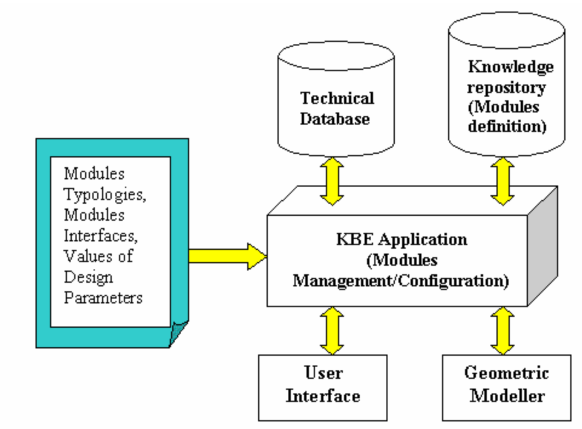

Figure 1. System architecture for module configuration

Currently the implementation of our system supports the configuration of modules and their assembly. The modular structure (modules typologies, modules interfaces and indipendent design parameters) is identified with a preliminary analysis. In the KBE application are implemented the rules that manage the parameters; the knowledge base is linked with a databse of standard components and with a knowledge repository in which are stored the modules definitions. The user can interact with the system using a user-friendly windows interface. The result is a detailed geometrical model of modules instances.

4. Practical example

4.1 The industrial test case

This work has been developed in collaboration with a manufacturer of woodworking machines. Such NC machines, called machining centers, are able to mill, drill, bore and tap the raw material with a single positioning.

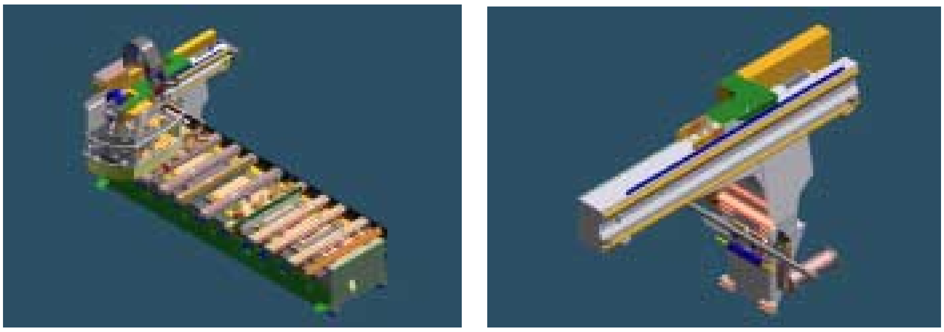

Figure 2. Three-dimensional model of a woodworking machine (left) and a movement axis (right)

The machine is equipped with two or more movement axis, with automatic blocking systems and reference systems.
Two main factors influence tha variability of machines: the different dimension (lenght, width and height) of wood components to be worked, the different shape of wood components. Such differences require an easy and rapid configuration of many machines typologies.
In the figure 2 is showed the 3d CAD model of a complete machine and the movement axis.

4.2 Method application

We have applied the method and the system described above to the design of a specific product family of three-axis machines. The different machines currently in production have been analysed. The result has been a group of function structures. Such diagrams have allowed the identification of the common functions and the specific functions (variants). The modules have been identified using an heuristics method based on the flows through the function structures [Stone et al. 1998].

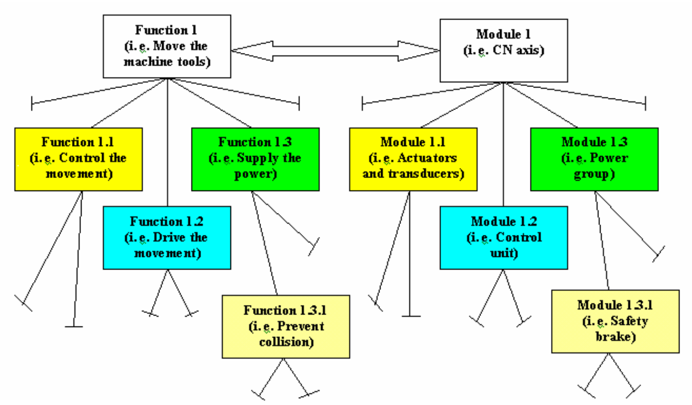

Figure 3. Partial structure function and corresponding modules for the movement axis

It has been obtained a hierarchical structure (similar to a tree) with a correspondence between the functions and modules (the first two level described in the section) 3. In Figure 3 a simplified schema is showed for the movement axis.
The next step has been the definition of the different type of knowledge required to configure every module. The use of an object-oriented system facilitate the identification of the correspondence between the module and the "object" to be implemented (Fig.4). In the knowledge base objects are

defined and stored on the basis of the design rules, formulas, tables, etc., used in the classical design process. Every module has been defined on the basis of chosen indipendent parameters (i.e. the module “hydraulic safety brake” is defined through oil typology, material, holes number, holes shape and cylinder diameter).
The other fundamental data are the modules interfaces. We can have two levels of interfaces: at high level to manage the flows conversion, at low level to allow the assembly of the physical components. The system allowed to handle them as particular objects properties.

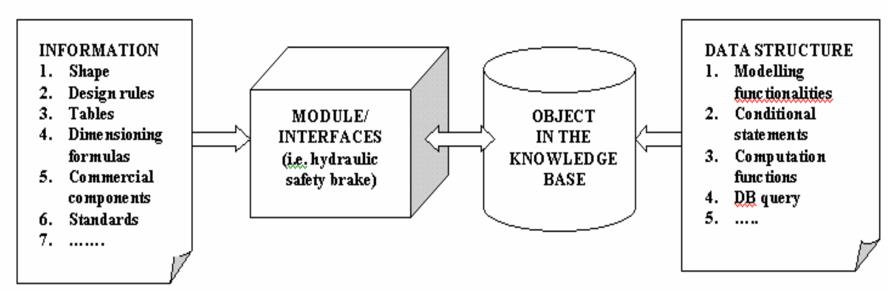

Figure 4. Correspondence between module and “object” in the KBE application

4.3 Results

As described above, the analysis of existing solutions and their main subsystems allowed the definition of the knowledge base needed to manage the configuration of solution. The knowledge base has been captured and formalised using an appropriate software tool (Selling Point by Oracle), to build a structure that makes up the computing core of the system. The computing core allows the automation of module configuration and generates a model of the specific module.

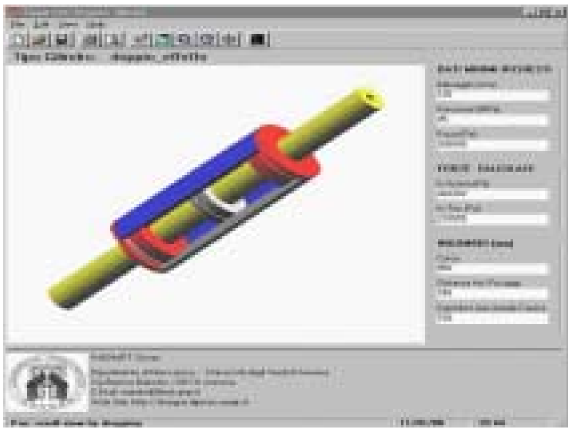

Figure 5. The KBE application user interface with a module instance of the safety brake

Currently we are working on a single high-level module (the movement axis) that is applied in every machine, but our aim is to implement all the needed modules to configure the complete woodworking machine. In the Figure 5 is illustrated the user-interface of software developed to generate a module instance (the hydraulic safety brake in the specific case) starting from the functional parameters and constraints.
The method application has showed many advantages in terms of time and usability; besides the object-oriented structure allows to manage complex configurations. The mechanism of encapsulation and inheritance could support, ideally, also the development of a multi-level design system to link the different product representation (functions, modules, instances/variants).
On the other hand two main drawbacks for small and medium enterprises can be highlighted: an initial great programming effort to set up the system, an high maintanance cost due to not easy upgrade of the system (every new solution implemented requires the expert's assistance).

5. Conclusions

In this paper has been proposed a design system application for the configuration of specific functional modules (systems related to the axial movement) used in the woodworking machines production. In particular, we have presented an approach to configure automatically the modules instances using a developed KBE system application based on the corporate design rules.
KBE systems has provided to the user the flexibility of an object oriented programming language to define the product model, containing all the different types of required knowledge. The representation object-oriented has resulted optimal for the module representation since allowed a mapping (one-to-one correspondence) between the modular level and the knowledge base of the system. The drawback is that the programming skill is required to implement the model (rarely present in SME industries).

Acknowledgement

This work has been conducted in collaboration with Biesse S.p.a. and we wish to thank the design team for the interesting discussions.

References

Allen, K., Carlson-Skalak, S., "Defining Product Architecture during Conceptual Design", Proceedings. of ASME 1998 Design Engineering Technical Conference DETC'98, Atlanta 13-16 September, 1998.

Berti, S., Germani, M., Mandorli, F., Otto, H., "Design of Product Families: An Example Within a Small and Medium Sized Enterprise", Proceedings of 13th International Conference on Engineering Design ICED'01, Vol. Design Methods for Performance and Sustainability, Glasgow 21-23 August, 2001, pp.507-514.

Bordegoni, M., Mandorli, F., "Multi-Level Modeling System for MCAD", Proceedings of XII ADM Conference on Design Tools and Methods, Rimini, 2001

Gonzales-Zugasti, J., Otto, K., "Modular Platform-Based Product Family Design", Proceedings. of ASME 2000 Design Engineering Technical Conference DETC'00, Baltimore 10-13 September, 2000.

Mandorli, F., Bordegoni, M., "Product Model Definition Support for Knowledge Aided Engineering Applications Development", Proceedings of ASME 2000 Design Engineering Technical Conference DETC'00, Baltimore 10-13 September, 2000.

Martin, M., Ishii, K., "Design for Variety: Development of Complexity Indices and Design charts", Proceedings of ASME 1997 Design Engineering Technical Conference DECT'97, 1997.

Otto, K., Wood, K., "Product Design: Techniques in Reverse Engineering and new Product Development", Prentice-Hall, New Jersey, 2001.

Sainter, P., Oldham, K., Larkin, A., Murton, A., Brimble, R., "Product Knowledge Management Within Knowledge-Based Engineering Systems", Proceedings of ASME 2000 Design Engineering Technical Conference DETC'00, Baltimore 10-13 September, 2000.

Simpson, T., Maier, J., Mistree, F., "Product Platform Design: Method and Application", Research in Engineering Design, Vol.13, 2001, pp.2-22.

Stone, R., Wood, K., Crawford, R., "A Heuristic Method to Identify Modules from a Functional Description of a Product", Proceedings of ASME 1998 Design Engineering Technical Conference DETC'98, Atlanta 13-16 September, 1998.

Tichem, M., Andreasen, M., Riitahuhta, A., "Design of Product Families", Proceedings of 12th International Conference on Engineering Design ICED'99, Vol.2, Munich 24-26 August, 1999, pp.1039-1042.

Tiihonen, J., Lehtonen, T., Soininen, T., Pulkkinen, A., Sulonen, R., Riitahuhta, A., "Modeling Configurable Product Families", Proceedings of 12th International Conference on Engineering Design ICED'99, Vol.2, Munich 24-26 August, 1999, pp.1139-1142

Ulrich, K., Tung, K., "Fundamentals of Product Modularity", Proceedings of the 1991 Winter Annual Meeting, DE-Vol.39, Atlanta, 1991.

Michele Germani, Post-Doctor Fellows
University of Ancona, Mechanical Department
Via Brecce Bianche, Ancona, Italy
www.dipmec.unian.it
++39-(0)71-2204969, ++39-(0)71-2204801,Email: m.germani@popcsi.unian.it