SUPPORTING THE CONFIGURATION OF NEW PRODUCT VARIANTS BY REUSING THE IMPLICIT KNOWLEDGE OF PAST SOLUTIONS

Malatesta, Marco (1); Cicconi, Paolo (1); Raffaeli, Roberto (2); Germani, Michele (1)
1: Università Politecnica delle Marche, Italy; 2: Università degli Studi eCampus, Italy

Abstract

The market globalization pushes for ever new products in order to reach new niches. In the household appliances field, the marketing specialists daily configure new combinations of numerous functional and product requirements seeking new product definitions. Each novel combination requires an assessment of technical and economic feasibility by the design departments.
This paper proposes a method for a preliminary validation of new configurations at the marketing stage. Indeed a tentative Bill of Materials (BOM) and a cost of the product are obtained. A knowledge base is derived by eliciting the requirement compatibilities from existing products. The approach is matrix based and it analyzes recurrent dependencies between requirements and components variants to determine which parts are most likely to appear in the BOM. Then, the knowledge base is integrated with rules that are input by experienced designers through a simple syntax.
The approach has been tested moving from the requirements of some instances of a family of cookers, and comparing the results obtained from the application of the method with the actual product BOM.

Keywords: Configuration, Design requirements, Knowledge management, Bill of Materials

Contact:
Dr.-Ing. Marco Malatesta
Università Politecnica delle Marche
Department of Industrial Engineering and Mathematical Sciences
Italy
m.malatesta@univpm.it

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 20th International Conference on Engineering Design (ICED15), Vol. nn: Title of Volume, Milan, Italy, 27.-30.07.2015

1 INTRODUCTION

Nowadays a globalized company experiences the necessity of offering a great variety of products. It is due to three main reasons: to be more appealing on the market through new products with different features and functions, to observe specifications, standards but also requirements of foreign market places, finally, to achieve economies of scale and strengthen the own market share applying strategies for taking over and merging with other firms and brands.

The main challenges to face this product variety affect both the marketing department, the design department and the production. The first one daily gathers data from the market, elaborates trends and conceives new products to be offered. Here, the main difficulties concern the preliminary technical and economical assessments of the new configurations that are defined by the operators. The marketing often lacks of technical skills, so continuous interaction with the design department is needed. On the production side, the problem is to handle the increasing number of variants coming from the design department (Jiao and Zhang, 2005) (ElMaraghy et al., 2013). In order to optimize the production, the number of components must be limited by reusing past solutions already developed in the extended company. Therefore, a crucial point is the knowledge regarding the product requirements, the compatibilities of the components and the production constraints. Such knowledge is distributed in the company and fragmented in many operators. The database (DB) formed by the already developed product variants, the relative list of marketing and functional requirements, as well as the Bill of Materials associated with them, form a large amount of data which can be exploit. Unfortunately, as it often happens, the use of ERP and PLM systems is restricted to punctual and unstructured searches to recall when a component was used and where.

In this context, the paper proposes a method to support the work of marketing operators in configuring new product instances. It concerns the elicitation of dependencies among product requirements to support the definition of new set of valid and compatible variant specifications. The formation of a tentative Bill of Material, given a valid set of requirements, through the application of a set of heuristics on the DB of existing product configurations. The recovering of the cost of single parts to calculate a possible production cost of a new variant.

2 STATE OF THE ART

The target of the mass customization (MC) is satisfying different customer needs with customized products trying to maintain the efficiency and cost of mass production (Pine, 1999). MC exploits the management theories of product variety. Variety or assortment is defined as a number or collection of different things of a particular class of the same general kind. Variant is an instance of a class that exhibits usually slight differences from the common type or norm (ElMaraghy et al., 2013).

Proposing on the market different features and functions of the same product typology, as well as opportunities for customization, generate more economic benefits. However, increasing too much the product variety actually results in lower sales (Wan et al., 2012). In fact, the offer of more product variants incurs higher expenses in design, production, inventory, selling and services. To raise the benefits and reduce the costs that stem from increasing product variety, Patel et al. (2014) highlight the need, especially for new ventures, to introduce modularity and flexibility. There are different approaches in the literature that debate how achieving an efficient product variety: product modularity and product configuration are the two main theories to support optimized management of the variants.

Modularity allows several products using standardized modules of components to be created. Therefore, product families are easily generated saving design time, production facilities and lowering the need of assembling new components (Simpson et al., 2001). On the other hand, configuration can be defined as the composition of a complex product from instances of a set of component types taking into account restrictions on the compatibility of those component types (Felfernig et al., 2011). That is, a product configuration is described as a system that is capable of creating, automatically or interactively, a product to satisfy both customers' needs and technical requirements without violating all constraints imposed on components due to technical and economic factors (Forza and Salvador, 2008). A configuration solution consists of the individual components, the assignment of values to their properties and the connection relations among components such that all constraints and customer requirements are satisfied (Yang et al., 2008). The application of product configuration systems avoids possible errors transferred between sale departments and engineering departments in manufacturing companies facilitating the sales-delivery process of products (Salvador and Forza, 2004).

If technical requirements are simply identifiable with engineering constraints, customer requirements are normally qualitative and tend to be imprecise. The latter are often included in functional requirements (FRs): the specifications of what a product should be able to do. FRs (Hauksdòttir et al., 2013) are often captured in use cases or work descriptions. On the contrary, non-FRs specify qualities that the product must have or criteria that the product must meet. They describe the spirit of its appearance, how easy it is to use, how secure it is, what standards apply to it. Non-FRs are often further specified as quality attributes. The most common are: look and feel, usability, reliability, performance, maintainability, portability, security and legal requirements. To understand and to specify this kind of requirements some methodologies are used, as well as the Quality function deployment (QFD) (Chan and Wu, 2005). QFD requires that customer needs are identified, quantified, translated into technical requirements and subsequently measured against how well the customer need is satisfied (Baxter et al., 2008). The implementation of different requirements and specifications is expressed in the company product portfolio. Its planning involves two main stages (Jiao and Zhang, 2005), the first one being the identification of the customer needs. The second is called product portfolio evaluation and selection and concern the optimal setup of these planned offerings with the objective of achieving best profit performance.
Finally, it is worth recalling that the amount of company data can be used to extract useful information for suggesting if new configurations of variants of product, that satisfy customer requirements, are also technically feasible. It is the field of data mining, which is defined as the process of extracting valid, previously unknown, and easily interpretable information from large DBs in order to improve and optimize engineering design and manufacturing process decisions (Braha 2001). The process of creating new variants starts from the marketing activities but, inevitably, it needs validation from the technical side. Connecting marketing with design and manufacturing becomes an important issue for what concern the rapidity and effectiveness of data exchange.

3 APPROACH AND ALGORITHMS DESCRIPTION

Figure 1 shows the flow of information among departments of a generic company producing consumer products or configurable products. The graph depicts the typical situation that is the scope of the proposed method. The marketing department gathers data from the market and the competitors. According to the company strategies and well consolidated approaches (for instance the QFD), the specifications for a new product are elaborated in terms of list of functionalities and performances.
The specifications included in the list can be numerous:

- Levels of required performance, for instance power, speed, weight
- Dimension and installation constraints, such as overall dimensions, fixture type, compatibility with selected environments
- Compliance with normative and standards, such as electrical tension and frequency, electromagnetic compatibility, temperature limits of exposed parts
- Functionalities, i.e. devices to provide specific functions to the user. For instance a wireless connection or a remote temperature control
- Aesthetic requirements, for instance appearance, material, color of a visible part of the product
- Assembly, maintenance or use requirements, such as consumptions, easy disassemblability, maintenance intervals, type of user interface (knobs, buttons, touch).

Such requirements could conflict each other and cannot be selected at the same time. For instance, a certain type of insulation solution could not be compliant with the normative of a nation or two functionalities cannot be technically available on the same product. The assessment of such constraints requires an evaluation by the design department since the marketing lacks of insights on technical aspects. This cause a first iteration loop and a criticality, since the marketing requires the evaluation of many product variants while the designers often perceive such requests as disturbing their everyday design activities. Once the set of requirements has been validated, the design department has to produce a preliminary product BOM. The purchasing department, thus, will be able to estimate the production cost in terms of bought parts, manufactured parts and assembly costs. The resulting cost comes back to the marketing that validates the compatibility with their expectations for the original target niche. In the worst cases, many iterations among the three departments are needed before having acceptable solutions. In this context, the paper seeks the minimization of such time wastes based on the following hypotheses. Produced product variants are stored in some company repositories as a list of requirement and/or product specifications. Technically, such data are usually found in spreadsheets, in custom software systems or in the ERP. Such DB is the source of the list of requirements that have been validated in the past. On the other hand, the ERP and the PLM systems contain the structure of the product variants, typically in the form of production BOMs while the design BOM may be absent. The BOM contains the list of the parts being used for a certain variant. From the ERP the cost of the single part is also available.

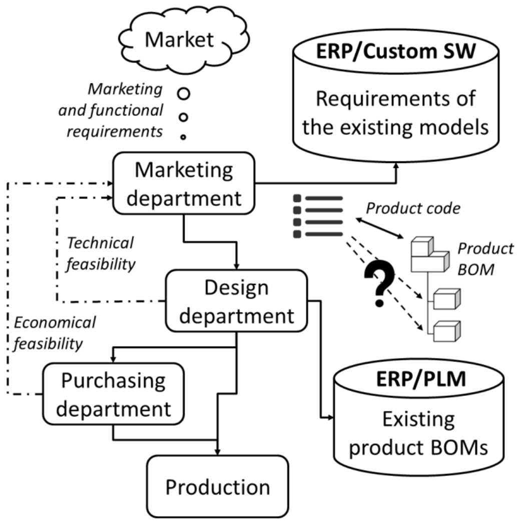

Figure 1. Scope of the approach: information flows, repositories and iterations

Given a certain product code, i.e. the identifier of a variant of a certain product family, the list of requirements and the BOM is therefore available. However, the connection between the single parts of product, as listed in the BOM, and the requirements is basically implicit and unknown. In fact, the heterogeneity of the requirements and the abstract level of the specifications do not allow a clear connection with the single part of the BOM. Such association requires the expertise of a senior designer. From these considerations, the proposed method aims to elicit the implicit configuration knowledge contained in the product variants that are already being produced to derive tentative BOMs. The goal is to provide the marketing department with a tool to estimate the BOM and the cost of a new variant and then reduce the iteration with the design and purchasing departments.

In particular, the proposed approach targets two main phases of the marketing department activity: the definition of a valid set of requirements for a new product variant and the computation of a tentative BOM given a set of requirements. Both the aims are achieved by mining data from the available company repositories.

3.1 Validating the requirement set of a new variant

The goal of validating the set of requirements has been faced analysing the existing lists. The past configurations are preliminarily organized as records in a DB table, namely the table of requirements, where the fields are represented by the requirements and the cells are the values assumed by each requirement for a certain product variant. Assuming to deal with configurations of n distinct requirements, four operators on the set of data are introduced:

- **Filter operator**: it filters a sub set of requirements to find similar configurations. Such operator is based on parametric queries on the table of requirements;
- **Distance operator**: it computes the Euclidean distance of two variants expressed in the n-dimensional space of the requirements. For each requirement, 1 is given if the value is equal and 0 if different. Such operator allows the most similar variants to be found and sorted in an automatic manner;
- **Compatibility evaluation**: it evaluates the compatibility of a pair of requirements by searching for a variant where the two requirements already coexist. This requires n*(n-1)/2 pairs to be considered to assess the global validity of a new configuration.
- **Rule-based exclusions**: it allows logic rules to be input to limit the values that can be assumed by a requirement. The syntax is simple and uses the logic operators AND, OR, NOT, EQUALS.

While the first two operators allow manual or assisted search to be performed, the other two allows a new variant to be assessed. In particular, the compatibility evaluation alerts the marketing expert on the novelty introduced by a new combination of requirements. On the other hand, the Rule-based exclusion operator aims to form a knowledge base that is grown by everyday usage. Although every rule must be formulated with the help and expertise of the design department, the growth of the rule DB progressively allows marketing operators to be independent. Moreover, the rule set forms a base of explicit configuration knowledge on the specific product family.

3.2 Computation of a tentative BOM

The second target of the approach aims to support the marketing specialist in the definition of a tentative BOM of the desired product configuration. The requirements of a product can be rarely mapped on specific product parts. It could happen just for limited situations. For instance, the request for the presence of a function in the product may be implemented with the inclusion of dedicated module. In this case, the mapping of the requirement with the physical components is straightforward. The paper introduces some heuristics to overcome this limitation. They are described as follows.

3.2.1 Parts classification and definition of a meta-product model

Since the method applies to a product family, from one variant to another some parts may disappear from the BOM while others could be present with different technical solutions, i.e. with different part codes. So *Part* is here defined as the category of a physical component, i.e. the set of characteristics and functionalities that makes a certain component being recognizable as that part. The parts of a family are arranged to form a Meta-model, that is a BOM including the different parts (i.e. kind of physical components) which may appear in any of the BOMs of the family members. Table 1 exemplifies the two introduced concepts. “Motor” is an example of part since it has no specific shape or reference in the company repositories. A “Motor” can be found in the product variants as “Induction motor code X100” or “Asynchronous motor code RFT345”, which are part codes, i.e. instances of the abstract “Motor” category. Meta-product model represents the product family and is given by the whole hierarchical set of possible parts. Each part of the meta-model may appear or not in a family member, and if it appears it assumes a specific part code.

Table 1. Representation of the concept of the product meta-model.

Meta-product model	Family member A		Family member B		Family member C	
	Presence	**Code**	**Presence**	**Code**	**Presence**	**Code**
Part 1	Yes	Code 1-1	No	-	Yes	Code 1-2
Part 2	No	-	Yes	Code 2-1	No	-
…	…	…	…	…	…	…
Part n	Yes	Code n-1	Yes	Code n-2	No	-

3.2.2 Population of the requirements-components dependency matrix

The knowledge of past solutions is encapsulated in the matrix that is presented in Table 2. A product requirement, for instance R1, is a specification of the product and can assume alternatives values, such as R1a, R1b, R1c. The requirements are listed as groups of table columns. A column is inserted for every value a requirement can assume. Each row of the table represents a part code (P1a, P1b, etc.). Rows are grouped by the part belonging to the meta-model (P1, P2, etc.). The cells of the table are filled with the identifier V1, V2, etc. of the product variants that are extracted from the company repositories. The presence of a variant in the cell means that a certain part code has been employed in combination with a certain requirement value. As a new variant of the family is introduced, a new set of requirements and the relative BOM are available. So the combinations in the table can be incremented with the new values.

Table 2. Structure of the requirements-components dependency matrix

		R1			**R2**			**…**
		R1a	**R1b**	**R1c**	**R2a**	**R2b**	**R2c**	**…**
P1	**P1a**	V1; V6	V2			V1; V2	V6	
	P1b		V3; V5		V5		V3	
	P1c							
P2	**P2a**	V1	V2			V1; V2		
	P2b	V6	V3				V3; V6	
	P2c		V5	V4	V4; V5			
…	**…**							

3.2.3 Definition of the product structure

Now it is possible to analyse original sets of requirements, which have been validated as in the previous section. The algorithm has to extract a product BOM from the meta-model, selecting the parts that are included in the new variant. Two heuristics are then defined.

Heuristic 1: *Only the columns corresponding to the values selected for the requirements are considered. A part is included in the product BOM if it is used at least for a product variant, in the form of any code, for a minimum number of requirements. Such threshold is chosen depending on the product family*.

Let consider a new variant with requirements R1c and R2a. The part P1 appears only under the column R2a so it is required only for the 50% of the requirements (1 out of 2). The part P2 is involved by both the requirements R1 and R2. Therefore, in the output BOM there will be the part P2.

3.2.4 Selection of the part codes

In this step, a code is selected for each part resulting from the previous phase. To this aim, the following rule applies:

Heuristic 2: *Only the columns corresponding to the values selected for the requirements are considered. Given a requirement R, a part P and a part code c, an index $I_{c,R}$ is defined as the ratio between the number of the variants $N_{c,R}$ that use such code and the total number of variants N_R.*

$$I_{c,R} = \frac{N_{c,R}}{N_R} \tag{1}$$

The part code c is selected as the one in P that maximizes the normalized sum I_c of the $I_{c,R}$ indices over the set of requirements*.

$$I_c = \frac{1}{R}\sum_R I_{c,R} \tag{2}$$

$$c^* \,|\, \max\{I_c\}, \forall c \in P \tag{3}$$

For example, let the part P1 in Table 2 to be considered for a new variant characterized by the requirements R1b and R2c. The indices I_{1a}, I_{1b} and I_{1c} for the three codes P1a, P1b and P1c would assume respectively the values of 0.417 ((0.333+0.5)/2), 0.584 ((0.667+0.5)/2) and 0. So the code P1b would be chosen for the new variant.

3.2.5 Computation of the variant cost

As the final step the cost of the new configuration can be computed as the sum of the cost of the single part codes being selected in the previous step. The index I_c represents a measure of the capacity of the selected code c* of satisfying the majority of requirements. Indeed, it represents a confidence index indicating how probable the choice of a certain part code is. In such a way, the marketing expert can highlight the areas of uncertainty and contact the design department for deeper analysis.

3.3 Description of the software tool

The proposed method has been implemented in the "Configurator" tool. This Windows based tool is a prototypical platform, developed in a research program focused on the management of product re-design. Particularly, the research context is on household appliances, such as, cookers, refrigerators and washing machines. The main research objective is to support marketing and technical users during the definition of new product variants. Thus, our tool aims to generate novel product configurations from the matching of different functional module variants, where a simple module can be new or already collected in a DB. The Configurator let different users (marketing, technical and cost analyst) add value to the configuration process. The base architecture is shown in Figure 2. The software kernel, programmed in Visual Basic.NET, can be accessed through three simple to use Graphical User Interfaces. A knowledge-based repository collects all compatibility rules that allow requirements to be validated. A DB system provides all technical and marketing data using a direct connection with ERP and PLM platforms. Each software interface provides a specific software functionality. The **Marketing Interface** offers the search functionality of existing products, where the filter setting is customized (Figure 3).The aim is to find the product variant closest to the customers' requirements. The marketing user can also define novel product configurations combining several functional modules. The assessment is provided by a knowledge-based solver that analyzes all validation rules. The **Technical Interface** provides the editing functionality for defining the validity rules. These rules verify the compatibility of any product requirements during the configuration of novel variants. Figure 4 shows an example of the simple syntax necessary to define the rules. Finally, the **Costing Interface** presents the functionality to generate and edit the early product BOM with cost values (Figure 5).

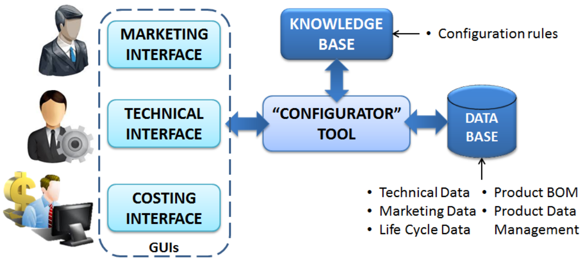

Figure 2. "Configurator" software architecture

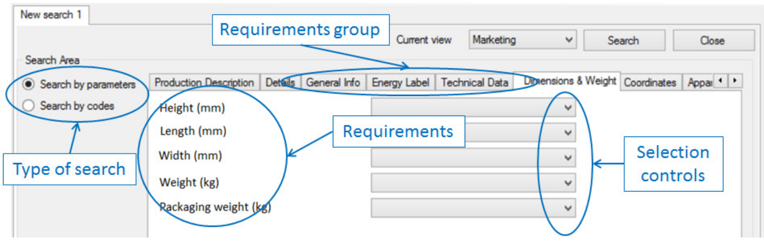

Figure 3 Marketing interface of the configurator software

IF ((Electronic Display = "A1") **OR** (Electronic Display = "A1+AUTO") **OR** (Auto Cooking = "TRUE") **OR** (Electronic Timer= "TRUE"); **THEN** Cooling Fan ="TRUE")

Figure 4. Example of a requirement validation rule

Code	Description	Quantity
24165		
W25241650010	K 342 G(W)/U	1
W21006052900	K 340 FG 2M CONT CL Z W	1
W21005765401	GR.DOKUM.KUCH.24165	1
W19501549304 (2)	STICKER - KUCHNIA NA GAZ G20(GZ50)	2
W19507227802	STICKER - ACCENSIONE PIANO MANUALE	1
W21005974300	PAKIET DOK.KUCH.24165	1
W21005767400	GR.POKRETEL C1,3 BIA.IND.	1
W14801875203	POKRE.ELE.C3 BIA	1
W14801875401 (5)	POKRE.GAZ.C3 BIA	5
W14802419200 (6)	KOL.POKRE.C1,3 SZA.JAS.	6
W20022211004 (6)	SPREZYNA POKR.50X60	6

Figure 5. Example of technical interface showing the BOM structure

4 APPLICATION TO THE HOUSEHOLD APPLIANCES FIELD

This section describes how the proposed tool supports the configuration of new product variants. A domestic cooker has been chosen as a test case to show how the developed platform works. This research-work has been developed in collaboration with Indesit SpA, a big Italian manufacturer of household appliances. A typical design workflow was reproduced for testing the prototypical software platform. It starts from customer requirements elaborated by the marketing unit. Traditional workflow provides a loop of many interactions between Marketing and R&D unit in order to obtain a technical validation of a product configuration. A user, with the Configurator Marketing Interface, can define autonomously a novel product variant. The resultant configuration set is validated by the rules implemented inside the knowledge-based system (Figure 2). This avoids any technical incompatibility. Finally, the flow ends with the cost analyst user that uses the Configurator Costing Interface to generate the product BOM and estimates the item cost. The platform efficiency has been evaluated in terms of reduction of time, BOM reliability and technical feasibility of resultant configurations.

4.1 Product introduction

A product family of freestanding cookers has been chosen for validating the proposed approach. This household appliance is characterized by high number variants and many SKUs (Stock Keeping Units). A standard domestic cooker consists of a cooking hob and an oven. A cooking hob is placed on the worktop, while the oven is on the middle-lower part. Cooking heat is provided by hob through combustion flames, electric resistance or electromagnetic induction. Each kind of cooking hob is characterized by further possible variants such as: burners number, grid type, performance and aesthetics. On the other hand, an oven is constituted of an insulated cavity where the cooking heat is provided by a gas burner grill or several electric resistances (base elements and electric grill). A gas oven consists of a gas burner grill on the bottom of the cavity, and an electric one on the top; while an electric oven has only electric resistances. A cooker may present twin or double cavity, not evaluated in this test case. Figure 6 (a, b, c) shows three examples of different product configurations regarding freestanding cookers: gas-electric, gas, and electric. The cooking system variability influences the product configuration, the technical solutions, and finally affects the cost and the BOM. Different possible combinations of heat sources are shown in Table 3.

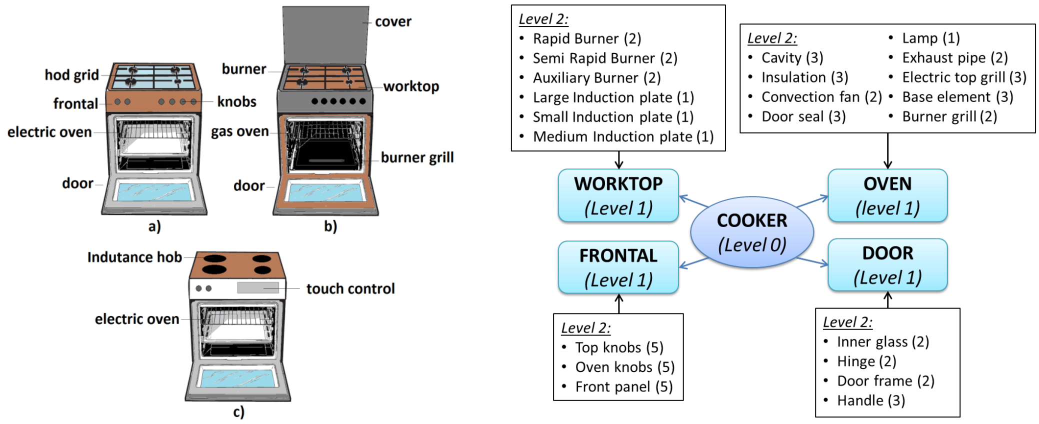

Figure 6. Convectional freestanding cookers: a) gas-electric; b) gas; c) electric. And a simplified cooker product structure (d)

Table 3. Example of different heat source combinations in cooker configuration.

System	Main component	Gas-Electric	Gas	Electric
Hob	Gas burners	x	x	
	Induction plates			x
Oven	Gas burner grill		x	
	Electric grill	x	x	x
	Electric resistance	x		x

4.2 Product structure definition

The full product structure of a cooker is organized in more than 10 subassembly levels and includes over 200 codes per single SKU. Three types of levels are recognized in a typical cooker structure: Level 0 that represents the SKU code, Level 1 that includes the list of main subassemblies, and finally Level 2 that includes the collection of all parts. In this paper, four representative subassembly groups are analyzed as shown in Figure 6 (d):

Worktop group: cooking hob with four burners or plates powered by gas, electromagnetic induction or electric resistance.

Oven group: cavity, insulation, convection fan, door seal, burner grill, lamp, exhaust pipe, base element and electric top grill.

Frontal group: top knobs, oven knobs, and front panel (aesthetics).

Door group: oven door with inner glass, hinges, doorframe and handle.

In the proposed test case, a reduced collection of components has been analyzed. The numeric value near each component in Figure 6 (d) indicates the number of variants here considered. In particular, the four described groups have been represented by 23 different types of components at Level 2. A potential of 54 different cookers can be configured using this reduced DB.

4.3 Product requirements

The full configuration of a freestanding cooker requires more than 50 different marketing and technical requirements. Here, only 10 representative requirements have been analyzed:

R1. Energy label: European energy efficiency class (from A to G);

R2. Market: the sales area (i.e. Italy, France, Great Britain, etc.);

R3. Oven capacity: it indicates the oven's inner volume in liters (i.e. 50, 58, etc.)

R4. Plug: specifications concerning the electric connection (i.e. Schuko, UK plug, etc.)

R5. Gas: fuel type (i.e. methane, LPG, etc.)

R6. Oven Power: maximum oven power in Watts (i.e. 2000W, 2100W, 2800W, 3100W, etc.)
R7. Frontal panel: aesthetics related to the frontal panel (i.e. stainless steel, stainless steel and glass, aluminum, varnished)
R8. Handle: aesthetics and shape of the door handle (i.e.: classic, restyling, tubular)
R9. User-interface: different types of user-interfaces (i.e. classic, touch control, etc.)
R10. Grid: aesthetics and shape of the hob grid (i.e. flat, cast iron, varnished, etc.)

4.4 Discussion of the obtained results

Table 4 shows eight existing product solutions named from V1 to V8 that have been used to populate the dependency matrix that includes requirements-components. As cited before, only 10 requirements are considered. Table 4 also contains the V* configuration that is the novel variant defined during the proposed workflow. Particularly, the new configuration V* refers to a cooker for the Italian market. This configuration was not present in the archive at the time of the experiment. Using the Configurator platform, all requirements were selected in the Marketing Interface for a DB querying. The query result confirmed that the desired combination was not present in the DB. Nevertheless, the marketing user was able to fill a new configuration set using proposed tools. Any incompatibility between features and components was avoided by design rules implemented beforehand. A product BOM was automatically built in less than 10 seconds for being reviewed by the cost analyst user.
Table 5 shows a comparison between the BOM computed by the Configurator tool and the final list of parts reviewed by the cost analyst. The developed algorithm fails in three cases: the cost of convection fan, inner glass and door frame.

Table 4. Examples of cooker configurations described by some product requirements.

		Configuration								
ID	**Requirement name**	V1 (217397)	V2 (217401)	V3 (269753)	V4 (269757)	V5 (293093)	V6 (293097)	V7 (323957)	V8 (359813)	V* (new)
R1	Energy label	C	B	C	B	C	A	A	A	A
R2	Market	Italy	France	Italy	France	France	France	France	Great Britain	Italy
R3	Oven capacity [lt]	65	58	58	58	65	58	58	50	65
R4	Plug	Schuko	Schuko	Schuko	Schuko	Schuko	Schuko	Schuko	UK	Schuko
R5	Gas	Methane	-	Methane	-	LPG	-	-	LPG	Methane
R6	Oven power [W]	3100	2100	3100	2100	3100	2100	2800	2000	2800
R7	Frontal panel	Inox	Inox	Inox	varnished	Inox	Varnished	Inox and glass	Aluminum	Inox and glass
R8	Handle	classic	classic	classic	classic	Restyling	Restyling	Tubular	Tubular	Restyling
R9	User interface	Basic	Basic	Basic	Basic	Basic	Basic	Touch control	Basic	Touch control
R10	Grid	Varnished	-	Cast iron	-	Varnished	-	-	Flat	Varnished

Table 5. Comparison between BOM computed by the configurator tool and the one defined by the design and costing departments. Cost details are not shown for confidentiality.

Group	Part	Computed BOM	Correct BOM	Part cost error [%]
Worktop	Rapid burner	P1b	P1b	-
	Semi burner	P2b	P2b	-
	Auxiliary burner	P3b	P3b	-
Oven	Cavity	P7a	P7a	-
	Insulation	P8c	P8c	-
	Convection fan	P9a	**P9b**	5,3%
	Door seal	P10c	P10c	-
	Lamp	P12a	P12a	-
	Exhaust pipe	P13a	P13a	-
	Base element	P14c	P14c	-
	Electric top grill	P15a	P15a	-
Frontal	Top knobs	P16d	P16d	-
	Oven knobs	P17d	P17d	-
	Front panel	P18a	P18a	-
Door	Inner glass	P19a	**P19b**	4,6%
	Hinge	P20a	P20a	-
	Door frame	P21b	**P21a**	2,1%
	handle	P22b	P22b	-

However, the average error cost on product BOM computation is about 5% on final cost considering a product assembly of 16 items. Considering the traditional design procedure, marketing users needed a lot of support from their colleagues working in the R&D unit (technical office). This means a continuous flow of sharing information that increased the total lead time. Also the cost analyst spent a lot of time on building the product BOM (up to 4 hours with some hundreds of parts). By formalizing technical aspects in rules and formulas, much time can be reduced using the Configurator platform. Time-savings of about 50% have been estimated during the proposed experimental testing which reproduced a typical design workflow based on the re-design of an existing configuration. The estimation of time-saving does not consider the time spent by technical users for filling the knowledge repository and DB with rules. In fact, this time is spent at the beginning of the Configurator implementation. Technical users should updates rules and the DB repository only when there are new parts or assembly to add. This strategy allows reducing time due to repetitive explanations and inquiries required by the marketing office.

5 CONCLUSIONS AND FUTURE DEVELOPMENTS

The fragmentation of know-how among the company departments as well as the key role of senior experienced people can be a strong limit to the business growth if not properly managed. IT systems can play a significant role in the communication and data sharing. However, it often happens that the information is implicitly stored and not available in a usable manner. The paper investigates a new software platform in the context of product variants definition, including requirements description and implementation. Companies continuously updating members of product families can benefit of data mining from repositories to recover rules, instantiate new members and derive tentative product BOMs to perform preliminary feasibility evaluations. Two main conditions are required. The first one regards the necessity of a consistent base of previous cases, so that the heuristics may converge to realistic solutions. The second one is that the new product definitions are obtained arranging previous technical solutions. Under such hypotheses the proposed method is expected to produces reliable results as shown by the preliminary test cases. As future work, it is necessary to test the approach on a larger base of data and to experiment the tool in the everyday activities of the marketing department. Moreover, the use could be extended to additional product families to check its robustness.

REFERENCES

Jiao, J. and Zhang, Y. (2005) Product portfolio identification based on association rule mining. Computer-Aided Design, Vol. 37, pp. 149-172.

ElMaraghy, H., Schuh, G., ElMaraghy, W., Piller, F. Schönsleben, P. Tseng, M. and Bernard, A. (2013) Product variety management. CIRP Annals - Manufacturing Technology, Vol. 62, No. 2, pp. 629-652.

Pine, B.J. (1999) Mass customization: the new frontier in business competition. USA: Harward Busin.Sch.Press.

Wan, X., Evers, P.T., Dresner, M.E. (2012) Too much of a good thing: The impact of product variety on operations and sales performance. Journal of Operations Management, Vol. 30, pp. 316–324.

Patel, P.C. and Jayaram, J. (2014) The antecedents and consequences of product variety in new ventures: an empirical study. Journal of Operations Management, Vol. 32, pp. 34–50.

Simpson, T.W., Maier, J.R.A. and Mistree, F. (2001) Product platform design: method and application. Research in Engineering Design, Vol. 13, No. 1, pp. 2-22.

Felfernig, A., Stumptner, A.M. and Tiihonen, J. (2011). Special Issue: Configuration. Artificial Intelligence for Engineering Design, Analysis and Manufacturing, Vol. 25, pp. 113-114.

Forza, C. and Salvador, F. (2008) Application support to product variety management. International Journal of Production Research, Vol. 46, No. 3, pp. 817–836.

Yang, D., Dong, M., Miao, R. (2008) Development of a product configuration system with an ontology-based approach. Computer-Aided Design, Vol. 40, pp. 863–878.

Salvador, F. and Forza, C. (2004) Configuring products to address the customization responsiveness squeeze: A survey of management issues and opportunities. I. J. of Production Economics, Vol. 91, No. 3, pp. 273–91.

Hauksdòttir, D., Mortensen, N.H., Nielsen, P.E. (2013) Identification of a reusable requirements structure for embedded products in a dynamic market environment. Computers in Industry, Vol. 64, pp. 351–362.

Chan, L., Wu, M. (2005) Quality function deployment: A literature review. E. J. of Op. Res., Vol.143, 463–497.

Baxter, D., Gao, J., Case, K., Harding, J., Young, B., Cochrane, S., Dani, S. (2008) A framework to integrate design knowledge reuse and requirements management in engineering design. Robotics and Computer-Integrated Manufacturing, Vol. 24, pp. 585–593.

Braha, D. (2001) Data mining for design and manufacturing: methods and applications. Dordrecht: Kluwer Academic.