



**HAL**  
open science

## Generic bill of functions, materials, and operations for SAP<sup>2</sup> configuration

Linda Zhang, Élise Vareilles, Michel Aldanondo

► **To cite this version:**

Linda Zhang, Élise Vareilles, Michel Aldanondo. Generic bill of functions, materials, and operations for SAP<sup>2</sup> configuration. *International Journal of Production Research*, 2013, 51 (2), p.465-478. 10.1080/00207543.2011.652745 . hal-01773873

**HAL Id: hal-01773873**

**<https://hal.science/hal-01773873>**

Submitted on 8 Nov 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Generic bill of functions, materials, and operations for SAP<sup>2</sup> configuration

Linda L. Zhang<sup>a\*</sup>, Elise Vareilles<sup>b</sup> and Michel Aldanondo<sup>b</sup>

<sup>a</sup>*IESEG School of Management (LEM-CNRS), Catholic University of Lille, Lille, France;*

<sup>b</sup>*Toulouse University – Mines Albi, Albi, France*

Most available studies on configuration focus on either sales configuration specifying functional features or production configuration addressing product components. It has been well recognised that automating most of the activities associated with specification, engineering, and process planning of customised products and their interactions is one key in achieving product customisation. Thus, treating sales configuration and product configuration separately may not contribute to product customisation from a systematic view although they may lead to the improvement of individual stages. Recognising this limitation of existing studies, in this paper, we propose integrated SAles, Product and Production (SAP<sup>2</sup>) configuration, which helps achieve product customisation from a holistic view. Its rationale lies in automating consistently sales, product and production configuration activities in one system. In view of the importance of configuration models, we focus on the model underpinning SAP<sup>2</sup> configuration called generic bill of functions, materials and operations (GBoFMO) and discuss it in detail. As the core of SAP<sup>2</sup> configuration, GBoFMO can provide companies with an insight into organising the large volumes of data and knowledge in the life cycle of product family development. We also report a case study of light passenger aircrafts to illustrate the GBoFMO.

**Keywords:** sales configuration; product configuration; production configuration; generic bill of functions, materials and operations

## 1. Introduction

Not only do today's customers increasingly impose their individual requirements on products, but they also demand fast deliveries and low prices. In response to such customer expectations, many manufacturing companies strive to implement product customisation, in attempting to offer diverse products as expected while using the available design and manufacturing capabilities. Configuration – specifying a customised product as a combination of a set of pre-designed components – has been recognised as one of the promising approaches to facilitating product customisation (e.g. Tiihonen *et al.* 1996, Hong *et al.* 2008, Song and Kusiak 2009, Trentin *et al.* 2011). Typical pre-designed components are parts, modules, and assemblies. As a special design activity, configuration is performed after fundamental design activities and capitalises on the results of these activities. Such results include function specifications, component design, and relationships between functions and components (Stumptner 1997, Stumptner *et al.* 1994). For given customer requirements, configuration determines functional features, components and component arrangement of the corresponding customised product while ensuring sufficient product performance. The main advantages of configuration include accommodating sales force automation, providing a right amount of product variety while fulfilling a wide range of customer requirements, coping with a large number of customised products, and shortening lead times in the sales-delivery process, as summarised in a number of studies (e.g. Sulonen *et al.* 1998, Bourke 2000, Forza and Salvador 2002).

A myriad of studies on configuration have been reported addressing various issues, such as constraint-based configuration formulation (e.g. Mittal and Frayman 1989, Xie *et al.* 2005, Aldanondo and Vareilles 2008, Jiang *et al.* 2011), configuration systems (e.g. Slater 1999, Ong *et al.* 2006, Chen and Wang 2009), knowledge representation (e.g. McGuinness and Wright 1998, Mailharro 1998), and configuration reasoning (e.g. Pu and Purvis 1994, Viswanathan and Allada 2001, Tseng *et al.* 2005). While the existing studies approach configuration from different perspectives and introduce different problem-solving methods and methodologies, they focus on separate issues (e.g. sales configuration, product configuration) while lacking a systematic view where interactions among issues are considered. While sales configuration handles the determination of functional features that

\*Corresponding author. Email: l.zhang@ieseg.fr

describe customised products, product configuration deals with the configuration of product components that technically define customised products. This focus is also true for all major ERP systems that include a configuration module (e.g. SAP, Baan) and to many other software solutions for small and medium sized companies (Tiihonen *et al.* 1996). Nevertheless, to different degrees, the available studies and systems contribute to the management of product variety and the quick definition of customised products. However, it is well understood that successful product customisation relies on the efficiency in both configuring and producing products (Hong *et al.* 2008, Zhang and Rodrigues 2009a).

A production process is formed by a number of operations with precedence relationships. Each operation is associated with one or more input items, an output item, manufacturing resources (i.e. machines, tools, fixtures and setups) and a cycle time. Because they materialise abstract design concepts (i.e. design specifications) into physical products, production processes play a major role in achieving production efficiency on shop-floors. To help obtain production efficiency of diverse products based on the available manufacturing resources, a concept of production configuration is discussed in several studies (e.g. Zhang 2007, Zhang and Rodrigues 2009b). Production configuration addresses the configuration of production processes for customised products by integrating the principles of product configuration and process planning. In determining production processes, production configuration utilises design similarity and commonality inherent in product variety available in a company's product offerings. The design similarity and commonality contributes to the configuration of such production processes that help maintain production stability of customised products by eliminating unnecessary production changeovers.

In response to the limited focus of existing configuration-related solutions and in view of the importance of production configuration in sustaining production efficiency, in this study, we propose a Sales, Product and Production (SAP<sup>2</sup>) configuration, in attempting to facilitate product customisation from a holistic view. More specifically, in accordance with customer requirements, SAP<sup>2</sup> configuration deals with sales, product and production configurations in one system. By consistently configuring functional features, product components and further operations and manufacturing resources, SAP<sup>2</sup> configuration is expected to reduce time and cost to fulfil a customer order from design to production while meeting the expected product performance. It is also expected to maintain product quality by keeping production as stable as possible based on production processes configured.

Fundamental to any configuration system, a configuration model details data and their relationships, thus representing all potentially configurable customised products (Krebs *et al.* 2003, Krebs 2006). In view of this important role of configuration models, in this study, we focus on the model underpinning SAP<sup>2</sup> configuration, called generic bill of functions, materials and operations (GBoFMO). In practice, companies implicitly use the relationships either between features and component design or between component design and process elements without clearly documenting such knowledge and the way to organise such knowledge. GBoFMO explicitly presents the knowledge about sales, design and process planning and how such knowledge can be organised. Another reason for us to address the configuration model, instead of other issues in SAP<sup>2</sup> configuration, such as configuration process modelling, system architecture design and configuration formulation, is that these issues are research topics on their own and deserve separate studies.

The rest of the paper is organised as follows. Section 2 presents a review of the related work, supporting our proposal of SAP<sup>2</sup> configuration. Section 3 discusses SAP<sup>2</sup> configuration with respect to its unique functions and its positioning in product customisation. We conceptualise the GBoFMO in detail in Section 4 by highlighting domains involved, views and their mapping and concept implications. A family of light passenger aircrafts is adopted in Section 5 to demonstrate the GBoFMO. We end this paper in Section 6 by pointing out the future research.

## **2. Related work**

Since the early 1990s, many commercial configuration systems and research oriented prototypes have been developed. Perhaps the most famous and best documented two are XCON (Barker and O'Connor 1989) and VT (Marcus *et al.* 1992). XCON is developed at the Digital Equipment Corporation for configuring VAX computer systems, while VT is developed at an Elevator company in New Jersey for configuring elevators. Some other examples are COCOS for configuring digital cross-connect systems in telephony (Stumptner *et al.* 1994), PROSE for configuring telecommunications hardware (Wright *et al.* 1993), Pconfig for configuring computers (Slater 1999), Lava for configuring digital switching systems (Fleischanderl *et al.* 1998), XKL for configuring passenger cabins for the aircraft AIRBUS A340 (Kopisch and Gunter 1992), and DSD for configuring complex drive systems

(Ranze *et al.* 2002). Besides commercial configurators, many prototypical configuration systems have been introduced by researchers, e.g. a web-centric configuration prototypical system for configuring bicycles (Ong *et al.* 2006) and a neural network-based configuration system for computers (Chen and Wang 2009). In addition, major ERP systems have been developed to include a configuration module, such as SalesPLUS in Baan (Yu and Skovgaard 1998, <http://www.buci.org/company.htm>), SalesBUILDER in Trilogy's Selling Chain software suite (Hales 1992; <http://www.trilogy.com/legalterms.php>), and SAP configurator (SAP, 1994). In spite of the variety of application areas, one feature common to the available systems, prototypes, and modules is that they deal with either sales configuration or product configuration while leaving production configuration unaddressed (Chen and Wang 2009). This motivates us to put forward SAP<sup>2</sup> configuration in this study.

Representing product components and their relationships, configuration models are fundamental inputs for developing and applying configuration systems per se (Krebs 2006). Recognising this importance, researchers have addressed configuration models from different perspectives. Understanding the difficulties in managing configuration models, Männistö and Sulonen (1999) introduce a set of concepts to capture configuration evolution, such as Is-instance-of relation, Is-a relation, and Has-part relation. Krebs *et al.* (2003) discuss how a configuration model is influenced by evolution and how configuration models evolve. Krebs (2006) reports a method to ensure correctness of configuration models when they evolve. Felfernig *et al.* (2001) adopt the unified modelling language to model a type of typical customised products: computers. McGuinness and Wright (1998) apply description logics to formally represent configuration models. They use an assembly for electronics gear for circuit packs as an example to demonstrate the proposed description logic-based modelling. Soinen *et al.* (1998) develop a generalised ontology to represent a configuration model and the knowledge embedded. Similarly, as a starting point for future research on SAP<sup>2</sup> configuration, in this study, we address the underlying configuration model with respect to its conceptualisation.

### 3. SAP<sup>2</sup> configuration overview

SAP<sup>2</sup> configuration is proposed to complement existing solutions by addressing production configuration and further integrating sales, product, and production configurations in one system. To achieve this, a wide range of data and knowledge about sales, marketing, design, planning, process, production, manufacturing resources, and costing are required. They collaboratively support SAP<sup>2</sup> configuration, as shown in Figure 1. Being modelled in a number of possible formats, such as rules, decision trees, formulas, equations, and mathematical models, knowledge acts as a control to instruct system operation. It assists in assessing customer input requirements and evaluating the selected configuration elements (e.g. functional features, components, operations). For given customer inputs, SAP<sup>2</sup> configuration outputs sales configuration result: a set of features describing a customised products and other order-related information. It also outputs the product's technical design represented by bills of materials (BOMs) and production processes represented by bills of operations (BOOs), as shown in Figure 1.

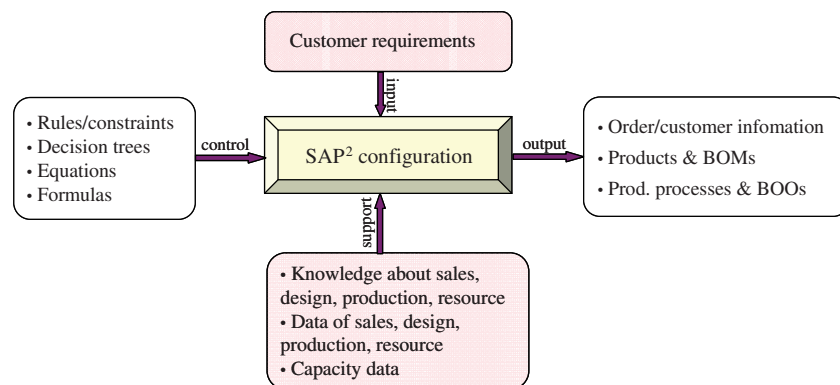


Figure 1. Overview of SAP<sup>2</sup> configuration.

### 3.1 Main functions of SAP<sup>2</sup> configuration

Common functions and the corresponding modules available in most reported software solutions and prototypical systems include customer requirements collection (performed by a user interface module), customer requirements assessment (carried out by an input evaluation module) and quotation preparation (performed by a quotation module). Besides these common functions, to achieve the expected outputs in Figure 1, SAP<sup>2</sup> configuration has three unique functions: integrated sales, product and production configuration, BOM and BOO generation, and configuration evaluation.

#### 3.1.1 Integrated sales, product, and production configuration

To achieve sales, product, and production configurations in one system, SAP<sup>2</sup> configuration necessitates all sales, design, planning, production, and process-related knowledge and data involved in a product family. To effectively organise various knowledge and data from different aspects and the interconnections both within and between aspects, we put forward GBoFMO as the underpinning configuration model (see details in Section 4).

Built upon the GBoFMO, three subfunctions contribute to the integrated configuration function, including sales configuration, product configuration and production configuration. Each of these subfunctions can be performed by a corresponding submodule. Carrying out sales configuration, the sales configuration submodule configures the set of compatible functional features that can meet the evaluated customer requirements. Based on the sales configuration result, the product configuration submodule aims to determine technical specifications of the customised product. It selects appropriate component types, subsequently determines component attributes and their values, and finally specifies design parameters and the corresponding values for defining components. The result of product configuration includes several product alternatives, each of which consists of a number of specific components and their parent-child relationships. For each product alternative, the production configuration submodule configures the corresponding production processes. Each production process is formed by a number of operations, operations precedence and machines together with other manufacturing resources that are used to perform operations. Owing to machine flexibility and operations flexibility, usually more than one production process is feasible to produce the same product. Since the GBoFMO contains planning and process knowledge, the production configuration submodule determines several production process alternatives for each product alternative.

#### 3.1.2 Configuration evaluation

Unlike available systems and prototypes, SAP<sup>2</sup> configuration explicitly deals with evaluation of configured product and production process alternatives. A configuration evaluation module can perform such evaluation function. First, it takes the result of the integrated configuration module: pairs of configured product and production process alternatives as input. It then evaluates each pair with respect to, e.g. production cost and completion time. Finally, it outputs the computed cost and completion time for each pair of product and production process alternatives. Based on such a result, a company can make a decision on the product to be offered by considering the trade-off between cost and delivery time, the relationship with the customer, its strategic objectives, and other factors.

To ensure accuracy, configuration evaluation should be based on the latest data about manufacturing resource availability that are obtained from existing production planning and scheduling systems and shop-floor execution systems. In this regard, a system based on SAP<sup>2</sup> configuration should be designed to enable the integration with company's legacy systems.

#### 3.1.3 BOM and BOO generation

The importance of error-free BOMs and BOOs for smooth production, high product quality, and reduced production lead time has been well understood (Ariano and Dagnino 1996). Some companies turn to configuration systems for solutions for BOM generation (Forza and Salvador 2002). In response to the lack of studies in BOO generation, SAP<sup>2</sup> configuration offers BOM and BOO generation function. In this regard, an SAP<sup>2</sup> configuration system should be designed to automatically generate both BOMs and BOOs for the configured products and production processes. The BOM and BOO generation module performs this function. The module generates BOMs and BOOs for the final products and production processes after configuration evaluation. To enable such generation, knowledge should be organised to guide the arrangement of components, operations,

and manufacturing resources. Furthermore, to accommodate future configuration, BOM and BOO data should be saved in corresponding databases in the SAP<sup>2</sup> configuration system.

### 3.2 Positioning of SAP<sup>2</sup> configuration

The importance in integrating processes between and within functional units for achieving product customisation has been well recognised (Forza and Salvador 2002, Yang *et al.* 2005). Configuration systems are expected to facilitate such integration (Bourke 2000). Owing to the inherent functionalities that they are designed for, most of the existing configuration systems do not address integration among all the major processes. SAP<sup>2</sup> configuration, as shown in Figure 2, attempts to fill up this gap by addressing a number of important issues in product family development based on artificial intelligence, Internet and computer technologies.

Based on the results of fundamental activities in (re)designing a product family, SAP<sup>2</sup> configuration explicitly models product and option selection rules, which are subsequently utilised to identify and validate customer requirements, features, components and operations within a generic unified structure: the GBoFMO. Thus, it accommodates identification of customer requirements, specification of functional features and processing of customer orders and configuration of products and production processes. In addition, it is proposed to generate products' technical documents (i.e. BOMs and BOOs) and customer- and order-related documents. Such a move away from manual process towards integrated automatic solutions provides a necessary step in reducing the time taken to produce products. Moreover, such computerised generation eliminates many human-related errors when manually creating these documents, especially BOMs and BOOs, thus ensuring a high level of correctness. In turn, the presence of such BOMs and BOOs that are virtually without errors increases material/capacity planning reliability, avoids production stoppages, and reduces the risk of delayed deliveries and the level of safety inventories.

In configuring, visualising products and generating documents, SAP<sup>2</sup> configuration should interact with legacy systems, such as CAPP, CAAP, CRM, and PDM, as shown in the figure. In addition to getting data from these systems, it inputs data and information (e.g. customer order) to the relevant systems (e.g. material/capacity planning system) so that all activities/processes involved in developing new products are consistently carried out. Finally, it contributes to managing the relationships with customers by correctly identifying, documenting their requirements and by getting the right products that can better meet these requirements. In summary, SAP<sup>2</sup> configuration transforms competencies and knowledge previously kept in individual experts from different fields into a unified knowledge structure embedded in the configuration system, resulting in organisation competencies and knowledge. By utilising such organisation competencies and knowledge while capitalising on technology advancements, SAP<sup>2</sup> configuration ties product development processes with all supporting processes, thus facilitating product customisation.

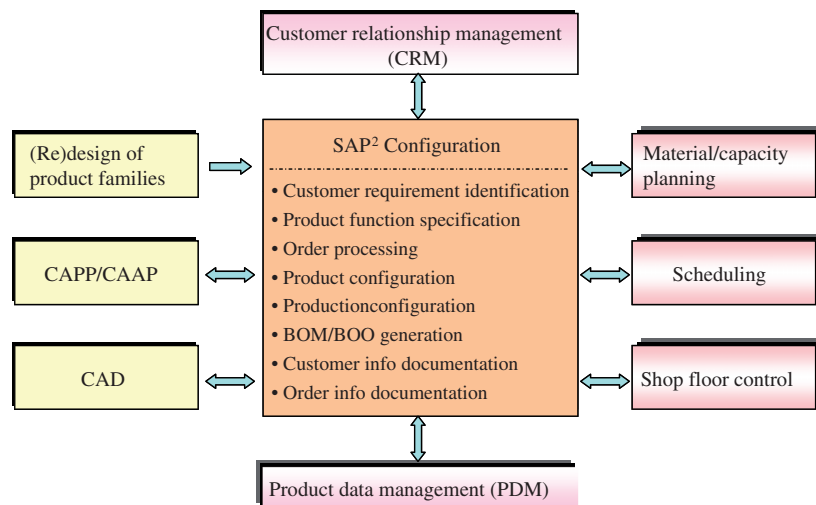


Figure 2. Positioning of SAP<sup>2</sup> configuration.



#### 4. GBoFMO conceptualisation

In line with the current practice in family-based product development, a GBoFMO in relation to a product family is proposed in this study as the underlying configuration model. It organises family-related sales, design, planning and process data and knowledge. In a product-development process, the concept of domains is widely applied to depict the transformation of a product from its conceptualisation to the fulfilment stage (Suh 2001). In accordance with this domain concept, the GBoFMO encompasses data from three domains, namely the functional, design, and process domains. In practice, the functional, design, and process data present products from three different views: the sales, design, and production views. Hence, a GBoFMO consists of these three views along with the mapping between different views.

##### 4.1 Domains involved in GBoFMO

As shown in Figure 3, this domain model helps organise elements in SAP<sup>2</sup> configuration by associating the characteristics of domain data to these configuration elements. For instance, the process domain interprets operations and manufacturing resources necessary to produce output items with given input items. The interlocking relationships between domains are indicated by arrows. The inter-domain connections together with these in each individual domain are the basis for mapping between two views.

Embodying customer perceptions on product offerings available in a product family, the functional domain captures a set of functional features and their values. These features collectively indicate what products can do and how they look like. Basically, two types of features can be distinguished, namely the atomic and composite features. An atomic feature (e.g. Feature 2 in the figure) is such a feature that individually defines a product function or appearance without interacting with other features. For instance, colour is an atomic feature describing a bicycle and assumes a number of values. On the contrary, a composite feature (e.g. Feature 1) describes a system function or appearance at a higher level and is decomposed into several child features. (For instance, as a parent feature, computer memory has two child features: temporary memory and permanent memory.) In turn, a child feature (e.g. CF11) can be further decomposed into its child features. Such decomposition ends with all the features being atomic ones and results in a hierarchical structure. The combination of values of child features involved leads to a value instance of the parent feature (i.e. a specific feature involved in a customised product). This is applied to parent and child features at any two adjacent levels of the hierarchy. Both the combination of different features and the combination of values of a same set of features contribute to variety in the product family to be configured. In other words, the customised products differ with one another in either different features or different values of a same set of features.

While the functional domain describes products in terms of features, the design domain represents products' technical specifications, more specifically components, their attributes and design parameters. In this regard,

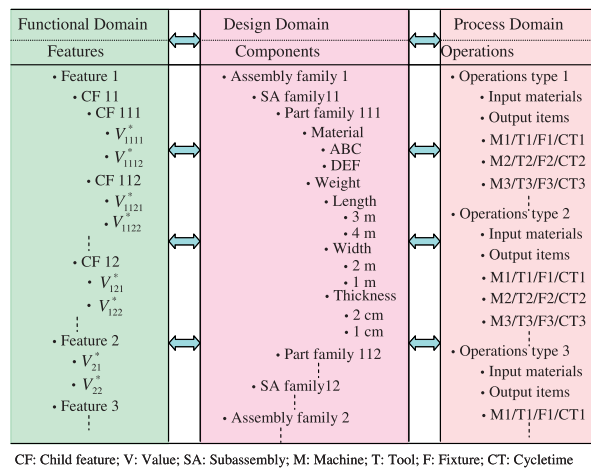


Figure 3. Three domains involved in GBoFMO.

features in the functional domain are delivered by components in the design domain. A component represents either an assembly or a part. To deliver variants of the same features (i.e. specific features), variants of the same component types are available, forming component families. Such component families are either assembly families or part families. In relation to the hierarchical structure of features in the functional domain, assemblies (e.g. Assembly family 1 in the figure) are formed by immediate child assemblies (e.g. SA family 12) and/or child parts. These immediate child assemblies (e.g. SA family 11) have their child assemblies and/or child parts (e.g. Part family 112). Unlike assemblies, parts do not possess child components, thus being basic components.

As with features describing customised products, attributes coupled with the corresponding values characterise specific components, be they assembly variants or part variants. These component variants are then defined by pairs of design parameters and values. Different with design parameters, an attribute shows a certain property of a component, indicating component uniqueness. It can be the derived result of a combination of multiple design parameters; it can also be a design parameter (depending on the design parameter). For example, as an attribute of a steel block, weight is determined by a number of design parameters: mass, length, width, and thickness; the attribute: colour is a design parameter at the same time.

While components are defined in the design domain, they are produced using elements available in the process domain. These elements include operations, manufacturing resources, and cycle times. Similarly, to produce component variants, an operation type includes a number of operations variants, each of which differs with one another in some detailed process parameters (Zhang and Xu 2010). In addition, different machines along with other resources can perform the exactly same operations while incurring different cycle times, costs and/or different operations precedence, as shown in the figure. The operations and manufacturing resources along with operations precedence for producing a customised products form a production process. Further, in connection with component assemblies and parts, which are not outsourced, a number of sub-production processes can be identified in the production process.

#### 4.2 Views and their mapping

In practice, based on their own interests and knowledge, people from different functions in a company interpret the same products from different views. For example, from the sales view, sales people see a product as a set of functional features; from the design view, designers see the product as a list of components; and from the production view, production personnel see the product as a list of ordered operations together with manufacturing resources. Since the GBoFMO organises the above data and knowledge both within and between the respective domains discussed above, it enables the integration of several business functions, such as the sales, design, and production functions, in a context coherent framework for configuring product families. It accomplishes this by establishing the mapping relationships between different views of a product family, as shown in Figure 4.

In the sales view, the GBoFMO exhibits the product line that is perceived by customers from different market niches. Thus, it manifests a variety of functional features and their interrelationships that involve both decomposition and dependency. In accordance with the different preference of customers, the sales view also

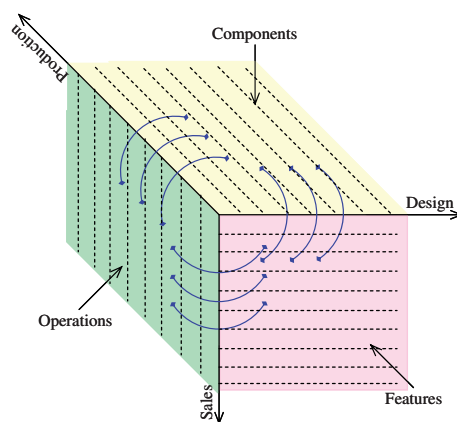


Figure 4. Three views and their mapping in GBoFMO.



embodies the relative importance (or priority) of functional features. In relation to features identified in the sales view, the design view reveals technology application to a product design. It describes the product design using components and their hierarchical structure. The hierarchical structure describes not only the decomposition of components into smaller units but also the interrelationships between components (e.g. require, exclude). Further, corresponding to specific features in the sales view, component attributes are defined to highlight differentiation and variety in configuration. In accordance with these attributes, design parameters are determined to uniquely define component variants. Production variations resulting from configuration differentiation in the design domain are addressed by the production view of the GBoFMO. This view represents product information by a description of physical realisation of the product configuration, thus concerning construction of the product. More specifically, it consists of manufacturing and assembly operations along with the necessary manufacturing resources.

As shown in Figure 4, mapping relationships exist between two views of the GBoFMO. First, different customer requirements are identified by a number of functional features in the sales view. Essentially, this identification entails sales configuration activities. Various functional features are mapped from the sales view to component attributes in the design view. In accordance with these component attributes, the technical specifications of components (design parameters and their interconnections) are further mapped. This mapping from features to component attributes and finally to technical specifications embodies product configuration activities. The mapping between the design and production views reflects considerations of manufacturing and logistics such that production processes are configured to produce customised products by assessing available manufacturing capabilities, production cost and lead time. The mapping between the production and sales views captures the correspondence between a physical product structure and its functionality, thus providing necessary information about cost and lead time to facilitate quotation preparation.

### 4.3 Concept implications

The GBoFMO entails a conceptual structure and overall logical organisation of a product family from the sales, design and production viewpoints. Such a conceptual structure and overall logical organisation acts as a generic umbrella, under which each new sales, product, and production configurations can be determined to fulfil individual customer requirements. In this regard, the GBoFMO involves two aspects: (1) a unified common structure within which variations in functions, technical design and production processes for customised products belonging to a family can be differentiated and (2) the configuration of specific functional features, components and operations from the unified common structure.

Figure 5 shows a representation example of the unified GBoFMO structure. Each node in the structure, be it product-related or process-related, is a generic concept in the sense that it represents a family of item variants of the same type. Consistent with the existing literature and in line with the discussion earlier, a product family is characterised by a set of features. Shown as mapping relationships in the figure, there are many interconnections among features, feature values, as well as features and feature values. These mapping relationships can be *require* meaning the selection of one feature (or a feature value) requires the selection of another, *exclude* indicating the selection of one feature (or a feature value) excludes the selection of another, and *or* denoting either one feature (or a feature value) among a number of alternatives can be selected. By satisfying these mapping relationships, appropriate features and the corresponding values can be determined, resulting in the sales configuration of a customised product.

The child component families involved in a product family have multiple descriptive attributes, each of which can assume a number of values. Mapping relationships exist between features of a product family and child components. Consistent with practice, by nature, such mapping relationships are many to one, indicating a component is described by one or more than one feature. As an example, Figure 5 shows such mapping relationships between End Product Family and Assembly Family 1. As a result of these mapping relationships, the determination of component inclusion and component attribute values is based on features. While specific components are described by a set of attribute values, they are technically defined by design parameter and value pairs. In this regard, there are mapping relationships between component attributes and design parameters, as shown in the figure. In addition, mapping relationships exist among design parameters, parameter values, as well as parameters and values. These mapping relationships are applicable to same component families but also among different component families. Figure 5 gives some examples of these mapping relationships between Assembly Family 2 and Assembly Family 3, between Assembly Family 2 and Part Family 22, and between Part Family 21 and

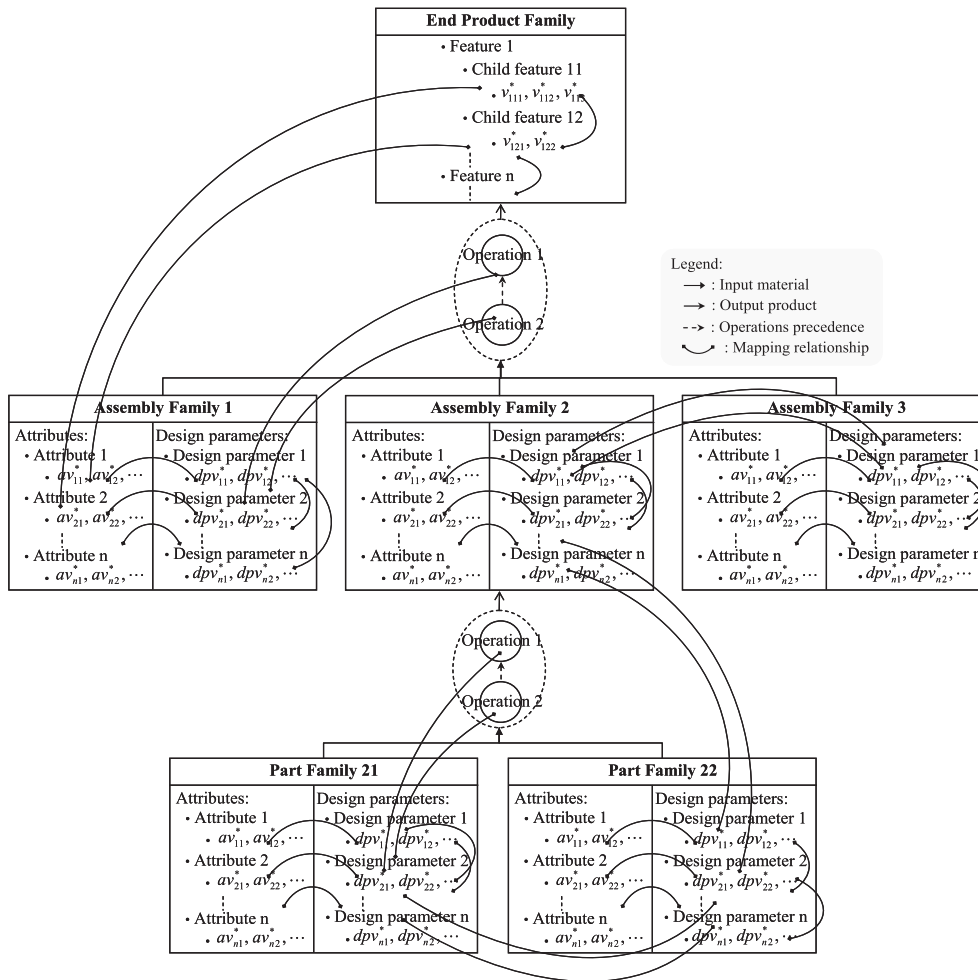


Figure 5. GBoFMO representation.

Part Family 22. This is consistent with the fact that the design of parent components influences that of child components and that the design of sibling components influences the design of other sibling components. As with the combination of specific features leading to a customised product, a set of compatible design parameters and the respective values uniquely define a component variant.

As shown in the figure, to produce parent components from immediate child components, operations, be they assembly or manufacturing operations, along with their precedence relationships form sub-production processes. (These operations are either assembly operations or manufacturing operations.) Connecting all these sub-production processes by following the parent-child relationships among components leads to a production process to produce a customised product. Similarly, mapping relationships exist between component design parameters and operations, as shown in the figure. In accordance with practice, such mapping relationships are one to many, denoting one design parameter may require one or more than one operation. They determine specific operations, manufacturing resources and operations precedence for given child component variants.

With all the above mapping relationships embedded in the GBoFMO, given a sales configuration describing a customised product, the product configuration in terms of components and their relationships can be obtained. Subsequently, as a result of production configuration, the production process consisting of ordered operations, manufacturing resources and cycle times can also be determined. In view of its hierarchical structure, the GBoFO concept is more suitable for products with a modular structure, instead of an integral one. Corresponding to such modular product structures, the production processes configured are modularised as well.

## 5. An illustrative example

In the light of the fact that aircrafts are typical configurable products and our experiences of working with a local aircraft company, we use light passenger aircrafts to demonstrate the GBoFMO concept. Unlike the airplanes, such as Boeing 777 and Airbus A380 which are used to transport a large number of passengers across two distanced locations, light passenger aircrafts are used to transport a small number of passengers from one location to a nearby location. Figure 6 shows an example of a light passenger aircraft and its major assemblies: engine, wings, cabin and electronics.

The family of light passenger aircrafts under investigation can be described from three views: the sales, design and production views. Table 1 gives the sales view of the aircraft family including features and values that these features can assume. There are four features distinguishing one aircraft from another, namely the number of passengers (NPA), the finishing level (FIN), the speed (SPE) and the flight range (FRA). Each of them can take on a specific value for a customised aircraft. For example, the feature: the number of passengers can be 4 if the customer prefers a small aircraft. The combination of different values of the five features results in different aircrafts. Such valid combination is determined by the relationships among features. An example of such relationships is given in

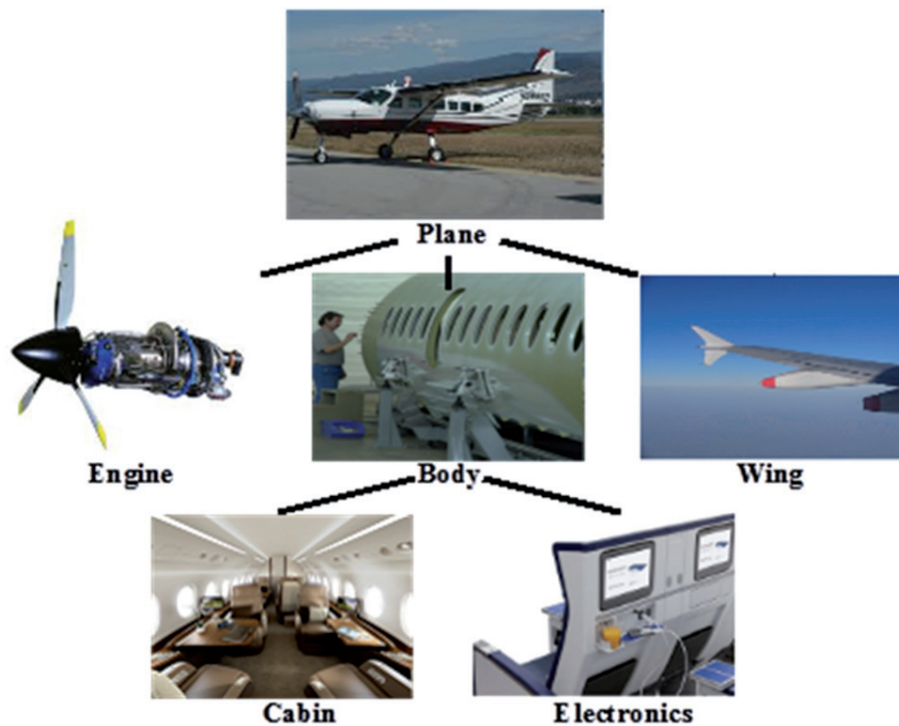


Figure 6. Light passenger aircraft and some major assemblies.

Table 1. Sales view of the light passenger aircraft family.

Features	Feature values
FIN: finishing level	Low, medium, high
NPA: number of passengers	4, 6, 8, 10
SPE: speed (km/h)	300, 400, 500, 600
FRA: Flight range (km)	300, 400, 500, 600, 800, 1000
Relationships among features	If speed = [400, 600], the flight range is $\leq 600$

the table: if the speed is between 400 (km/h) and 600 (km/h), the flight range is less than 600 (km). These relationships *de facto* reflect configuration constraints in the sales view.

The design view of the aircraft family is detailed by aircraft components, component attributes, and design parameters. Table 2 shows the design view with respect to the major assemblies including engine, body, wings, cabin, and electronics. Each of these assemblies is described by a number of attributes and technically defined by design parameters. While the configuration of attributes is influenced by the combination of feature values of aircrafts, it determines design parameter values of aircraft components. In this regard, relationships exist across the sales and design views, and within the design view as well. For example, a body is characterised by an attribute weight and is technically defined by a length and a width. Table 2 also gives the possible values that component attributes and design parameters can assume and some examples of relationships between component attributes and design parameters.

The production view of the aircraft family is shown in Table 3. Generally speaking, for assembling an aircraft, two assembly operations, namely assemble wings and mount engine, are necessary. In addition, assemble wings must be carried out before mount engine. The specific manufacturing resources required and the estimated cycle times to be incurred are influenced by the design parameters of input components. This is the same for assembling the body from a cabin and some electronics.

Upon analysing the sales, design, planning and production data, and knowledge pertaining to the aircraft family, the GBoFMO is constructed, as shown in Figure 7. The GBoFMO represents the sales data (e.g. features of the aircraft family), the design data (e.g. wing attributes and design parameters), and the production data (e.g. the assembly operations to assemble an aircraft) in one unified structure. Thus, it enables a corresponding configuration system to consistently configure the set of features of a customised aircraft, component designing aircrafts, and operations producing aircraft.

## 6. Conclusions

Complementing existing configuration-related studies, this study proposed SAP<sup>2</sup> configuration to facilitate product customisation from a holistic view. Its rationale lies in automating most of activities associated with specification, engineering and production of customised products. In this regard, SAP<sup>2</sup> configuration offers three unique functions: integrated sales, product and production configuration, BOM and BOO generation,

Table 2. Design view of the light passenger aircraft family.

Components	Attributes and values		Design parameters and values	
Engine	Gas-need (l)	400, 600, 800	Tank-size (l)	400, 800, 1200
	Engine-power (hp)	400, 500, . . . , 1200	Chamber size (l)	0.4, 0.8, 1.2
			Number of compressor stages	2, 4, 6
Body	Body-weight (kg)	[1960, 4840]	Body-length (m)	[8, 16]
			Body-width (m)	[4, 8]
Wings	Cargo-volume (m <sup>3</sup> )	[0, 2]	Number of seat supports	2, 3, 4, 5
	Portance	[10 000, 50 000]	Shape	Delta, straight, medium
	Wing-weight (kg)	[500, 1000]	Wing-length (m)	[3, 5]
Cabin	Cabin-weight (kg)	[1000, 2000]	Material	Aluminium, carbon
			Number of standard seats	4, 6, 8, 10
			Number of luxury seats	4, 6
Electronics	Natural light degree	Low, medium, high	Number of portholes	2, 4, 6
	Electric power (kw)	[1, 10]	Cabin equipment level	Low, medium, high
			Pilot equipment level	Low, medium, high
	Autonomy (mn)	10, 20, 30	Battery capacity (A)	20, 30, 40, 60, 80
Relationships among attributes and design parameters	If engine-power is 400, the number of compressor stages is 2 and chamber size is 0.4 If engine-power is 500, the number of compressor stages is 4 and chamber size is 0.8 If engine-power is 1100, the number of compressor stages is 6 and chamber size is 1.2 Body-weight = 1000 + 30 × body-length × body-width			

Table 3. Production view of the light passenger aircraft family.

Operations		Manufacturing resources	Cycle times
For assembling an aircraft			
Sequence 1	Assemble wings	Workstation A, Overhead crane-2T, Overhead crane-4T, Human-win-A1, Human-win-A2	[4, 8]
Sequence 2	Mount engine	Workstation B, Jig_l, Jig_r, Jack_l, Jack_r, Human-eng-B1, Human-eng-B2	[1, 3]
For assembling a body			
Sequence 1	Assemble body	Workstation C, Overhead crane 1T, Overhead crane 2T, Mech-TollCase-2140, Human-mech-ass-C1, Human-mech-ass-C2	[10, 42]
Sequence 2	Mount electronics system	Workstation D, Scope-tool-11T, Electr-ToolCase-1140, Human-mech-elec-D3, Human-mech-elec-D4	[8, 36]
Relationships manufacturing resources	If material is Carbon, the overhead crane-2T must be used for assembling wings If Overhead crane-2T is used for assembling wings, the cycle time is within the range of 4–6 h If Overhead crane-2T is used for assembling wings, Human-win-A1 must assemble the wings		

Note: The unit of measure for cycle time is hours.

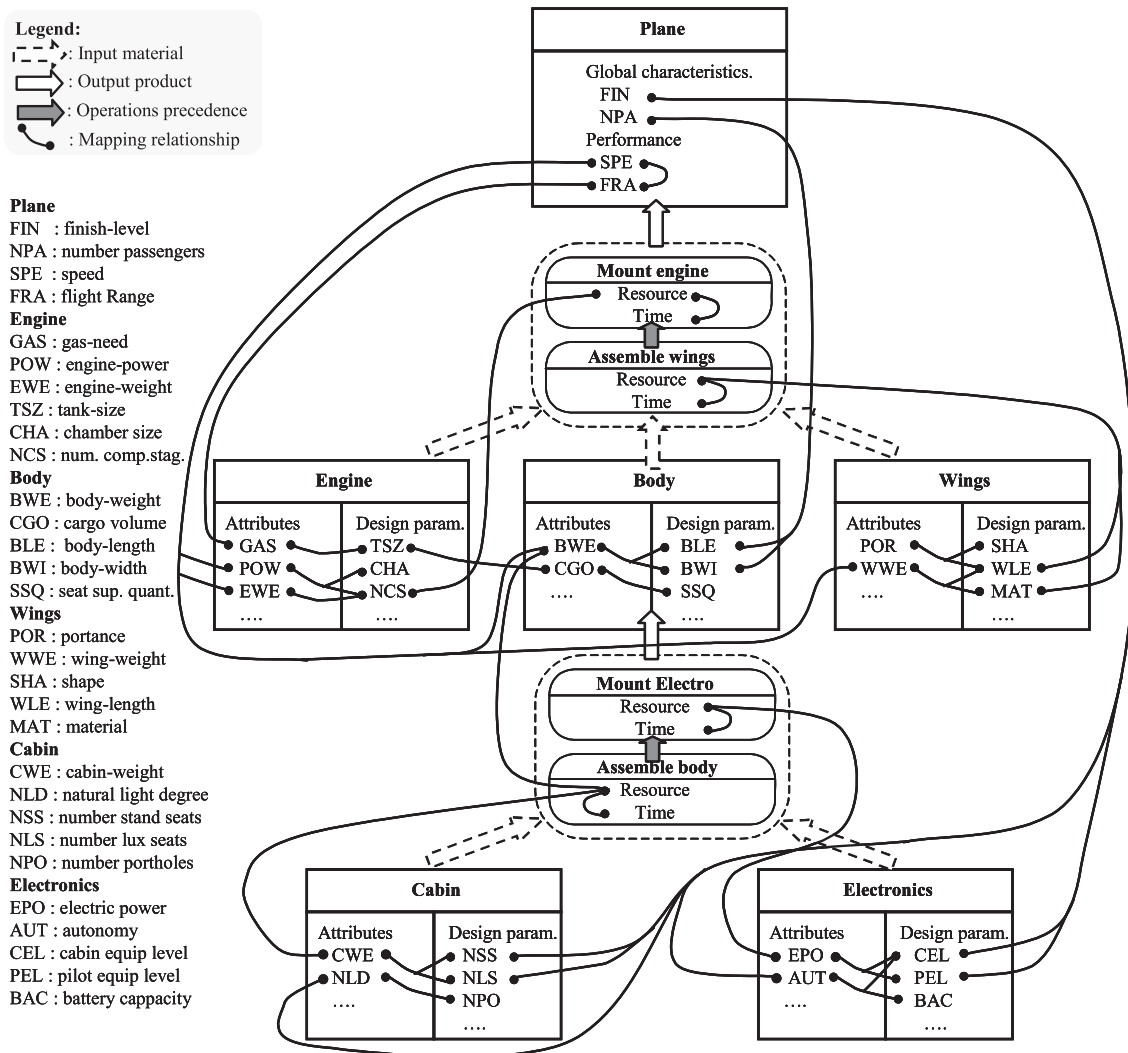


Figure 7. GBoFMO of the light passenger aircraft family.



and configuration evaluation. Recognising the importance of configuration models, we focused on the model underpinning SAP<sup>2</sup> configuration: the GBoFMO. As a starting point, in this study, we conceptualised GBoFMO and discussed it in detail from different perspectives: domains involved, views and view mapping, and concept implications. We also used a family of light passenger aircrafts to demonstrate the GBoFMO. In the near future, we plan to address configuration evaluation based on suitable programming models. In addition, more efforts from us and/or other researchers might be directed to these topics pertaining to SAP<sup>2</sup> configuration: SAP<sup>2</sup> configuration formulation, configuration process modelling, configuration evaluation, system architecture design, and prototype development. In the last topic (i.e. system architecture design and prototype development), one important issue to be addressed is how the system should be linked with legacy systems. This is because a good solution for this issue can help companies effectively design an SAP<sup>2</sup> configuration system by fully capitalising on the existing systems and databases.

### Acknowledgements

The authors would like to thank the French National Research Agency and the Aerospace Valley activity cluster for funding this study.

### References

- Aldanondo, M. and Vareilles, E., 2008. Configuration for mass customization: how to extend product configuration towards requirements and process configuration. *Journal of Intelligent Manufacturing*, 19 (5), 521–535.
- Ariano, M. and Dagnino, A., 1996. An intelligent order entry and dynamic bill of materials system for manufacturing customized furniture. *Computers and Electronic Engineering*, 22 (1), 45–60.
- Barker, V.E. and O'Connor, D.E., 1989. Expert systems for configuration at Digital: XCON and beyond. *Communications of the ACM*, 32 (3), 298–318.
- Bourke, R.W., 2000. Product configurators: key enablers for mass customization: an overview. *Midrange Enterprise*, August (2000).
- Chen, Z. and Wang, L., 2009. Adaptable product configuration system based on neural network. *International Journal of Production Research*, 47 (18), 5037–5066.
- Felfernig, A., Friedrich, G., and Jannach, D., 2001. Conceptual modeling for configuration of mass customizable products. *Artificial Intelligence in Engineering*, 15 (2), 165–176.
- Fleischanderl, G., et al., 1998. Configuring large systems using generative constraint satisfaction. *IEEE Intelligence Systems*, 13 (4), 59–68.
- Forza, C. and Salvador, F., 2002. Managing for variety in the order acquisition and fulfillment process: the contribution of product configuration systems. *International Journal of Production Economics*, 76 (1), 87–98.
- Hales, H.L., 1992. Automating and integrating the sales function: how to profit from complexity and customization. *Enterprise Integration Strategies*, 9 (11), 1–9.
- Hong, G., et al., 2008. Identification of the optimal product configuration and parameters based on individual customer requirements on performance and costs in one-of-a-kind production. *International Journal of Production Research*, 46 (12), 3297–3326.
- Jiang, Z., et al., 2011. Inventory-shortage driven optimization for product configuration variation. *International Journal of Production Research*, 49 (4), 1045–1060.
- Krebs, T., 2006. Evolution of configuration models – a focus on correctness. In: *Proceedings of the ECAI 2006 Workshop on Configuration*, 28–29 August, Riva del Garda, Italy, 31–36.
- Krebs, T., Hotz, L., Ranze, C., and Vehring, G., 2003. Towards evolving configuration models. In: *PuK2003 – Papers from the KI Workshop*, 123–134.
- Kopisch, M. and Gunter, A., 1992. Configuration of a passenger aircraft cabin based on conceptual hierarchy, constraints, and flexible control. In: *Proceedings of the IEA/AIE Conference*, Paderborn, 1992.
- Mailharro, D., 1998. A classification and constraint-based framework for configuration. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 12 (4), 383–397.
- Marcus, S., Stout, J., and McDermott, J., 1992. VT: an expert elevator design that uses knowledge-based backtracking. *Artificial Intelligence in Engineering Design*, 1, 317–55.
- Männistö, T. and Sulonen, R., (1999). Evolution of schema and individuals of configurable products. In: *Proceedings of ECDM'99 – Workshop on Evolution and Change in Data Management*, 12–23.
- McGuinness, D.L. and Wright, J.R., 1998. Conceptual modeling for configuration: a description logic-based approach. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 12 (4), 333–344.



- Mittal, S. and Frayman, F., 1989. Towards a generic model of configuration tasks. *In: Proceedings of the 11th International Joint Conference on Artificial Intelligence*, Detroit, USA.
- Ong, S.K., Lin, Q., and Nee, A.Y.C., 2006. Web-based configuration design system for product customization. *International Journal of Production Research*, 44 (2), 351–382.
- Pu, P. and Purvis, L., 1994. Formalizing case adaptation in a case-based design system. *In: Proceedings of the 3rd International Conference on Artificial Intelligence in Design (AID94)*, Lausanne, Switzerland.
- Ranze, K.C., et al., 2002. A structure-based configuration tool: Drive Solution Designer DSD. *In: Proceedings of 14th Conference of Innovative Applications of AI*.
- SAP. 1994. *Systeme R. Der SAP Konfigurator*. SAP AG. Mat.-Nr.:5000 5525.
- Slater, P.J.P., 1999. Pconfig: a web-based configuration tool for configure-to-order products. *Knowledge-Based Systems*, 12 (5), 223–230.
- Soininen, T., et al., 1998. Towards a general ontology of configuration. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 12 (4), 357–372.
- Song, Z. and Kusiak, A., 2009. Optimizing product configurations with a data-mining approach. *International Journal of Production Research*, 47 (7), 1733–1751.
- Stumptner, M., 1997. An overview of knowledge-based configuration. *AI Communications*, 10 (2), 111–125.
- Stumptner, M., Haselock, A., and Friedrich, G., 1994. COCOS – a tool for constraint-based, dynamic configuration. *In: Proceedings of the 10th IEEE Conference on AI Applications (CAIA)*, San Antonio, TX, 1994.
- Suh, N.P., 2001. *Axiomatic design – advances and applications*. New York: Oxford University Press.
- Sulonen, R., et al., 1998. Configurable products – lessons learned from the Finish industry. *In: Proceedings of the 2nd International Conference on Engineering Design and Automation*, Hawaii, USA.
- Tiihonen, J., et al., 1996. State-of-the-practice in product configuration—a survey of 10 cases in the Finnish industry. *Knowledge Intensive CAD*, 95–114.
- Trentin, A., Perin, E., and Forza, C., 2011. Overcoming the customization-responsiveness squeeze by using product configurators: Beyond anecdotal evidence. *Computers in Industry*, 62 (3), 260–268.
- Tseng, H.-E., Chang, C.-C., and Chang, S.-H., 2005. Applying case-based reasoning for product configuration in mass customization environments. *Expert Systems with Applications*, 29 (4), 913–925.
- Viswanathan, S. and Allada, V., 2001. Configuration analysis to support product redesign for end-of-life disassembly. *International Journal of Production Research*, 39 (8), 1733–1753.
- Wright, et al., 1993. A knowledge based configurator that supports sales, engineering, and manufacturing at AT&T network systems. *In: Proceedings of the 5th Conference on Innovative Applications of AI*, AAAI Press, 1993.
- Xie, H., Henderson, P., and Kernahan, M., 2005. Modelling and solving engineering product configuration problems by constraint satisfaction. *International Journal of Production Research*, 43 (20), 4455–4469.
- Yang, Y., et al., 2005. An internet-based product customization system for CIM. *Robotics and Computer-Integrated Manufacturing*, 21 (2), 109–118.
- Yu, B. and Skovgaard, H.J., 1998. A configuration tool to increase product competitiveness. *IEEE Intelligence Systems*, 13 (4), 34–41.
- Zhang, L., 2007. *Process platform-based production configuration for mass customization*. Thesis (PhD). Nanyang Technological University, Singapore.
- Zhang, L. and Rodrigues, B., 2009a. A tree unification approach to constructing generic processes. *IIE Transactions*, 41 (10), 916–929.
- Zhang, L. and Rodrigues, B., 2009b. Nested colored timed Petri nets for production configuration of product families. *International Journal of Production Research*, 48 (6), 1805–1833.
- Zhang, L. and Xu, Q., 2010. Solving production configuration based on constraint satisfaction. *In: Proceedings of the 2010 IEEE International Conferences on Industrial Engineering and Engineering Management (IEEM)*, 7–11 December 2010, Macau, China.