Downloaded from orbit.dtu.dk on: Jan 23, 2019

DTU Library

Reference frame for Product Configuration

Ladeby, Klaes Rohde; Oddsson, Gudmundur Valur

Published in:
World Conference on Mass Customization, Personalization, and Co-Creation

Publication date:
2011

Link back to DTU Orbit

Citation (APA):
Ladeby, K. R., & Oddsson, G. V. (2011). Reference frame for Product Configuration. In World Conference on Mass Customization, Personalization, and Co-Creation

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Reference Frame for Product Configuration

Klaes Ladeby & Gudmundur Oddsson

Abstract

This paper presents a reference frame for configuration. The reference frame is established by review of existing literature, and consequently it is a theoretical frame of reference. The review of literature shows a deterioration of the understanding of configuration. Most recent literature reports on configuration systems in the shape of anecdotal reporting on the development of information systems that perhaps support the configuration task – perhaps not. Consequently, the definition of configuration has become ambiguous as different research groups defines configuration differently. This paper propose a reference frame for configuration that permits 1) a more precise understanding of a configuration system, 2) a understanding of how the configuration system relate to other systems, and 3) a definition of the basic concepts in configuration. The total configuration system, together with the definition of key concepts, comprises a strong frame of reference when working with analysing configuration systems.

1. Introduction

Product configuration systems are increasingly seen as an interesting option for firms who wish to pursue a strategy with high degree of product variance while retaining a low cost of specifying the product (Tseng & Piller, 2005). This scenario is often present in mass production companies who wish to pursue a mass customization strategy (Haug, Ladeby, & Edwards, 2009). Product configuration systems can also be seen as direct productivity drivers in engineering companies and productivity gains are documented. For instance, in 1999 FL.Smidth, a Danish engineering firm experienced an average drop in engineering hours from 2500 to 190 for producing a quote (Hvam, Malis, Hansen, & Riis, 2004; Hvam, 2006). Product configuration systems may also find use as stand alone product validation systems in parts of the engineering process thus also providing productivity gains (Larsen, Ladeby, & Gjøl, 2006). Finally, product configuration systems are seen as key enablers in reducing both the cost of eliciting customer wishes and the costs of controlling the high degree of variety in mass customisation companies (Moser, 2007).

The sad observation from the Danish research project 'Product Configuration – Economical, Technical and Organizational Issues' (PETO) has been that product configuration projects are systematically delayed. Indeed, all too often at least twice the estimated time is used (Edwards & Riis, 2004; Edwards & Ladeby, 2005). While product configuration systems indeed offer significant productivity potential, developing and implementing configuration systems are apparently difficult and prone to delays or failures.

Unfortunately, most published work about product configuration systems in general seems to refer only to an intuitive definition of the basic concepts of configuration, and therefore it is difficult to make meaningful comparisons, and not to say impossible to identify why some projects fail while others succeed (a few exceptions are (Stumptner, 1997; Fleishanderl, Friedrich, Haselbock, Schreiner, & Stumptner, 1998; Sabin & Weigel, 1998; Soininen, Tiihonen, Männistö, & Sulonen, 1998; Felfernig, Friedrich, Jannach, & Stumptner, 2004; Haug, 2007)).

Configuration is in some cases considered a part of the design science discussion - a special case of the general field of design activities. In Stumptner (1997) it is assumed that in configuration the design goals and requirements are fully specified, and subcomponents and functions are already known. This assumption is also supported by Sabin and Weigel (1998), who note: "…product configuration is informally a special case of design activity and consists of two key features: a) the artefact being configured is assembled from instances of a fixed set of well-defined component types, and b) components interact with each other in predefined ways."

The core of configuration is selecting and arranging combinations of existing parts that satisfy given specifications. No new component types can be created nor can the interfaces of the existing components be modified. The configured solution must provide a list of selected components, and describe the product structure and topology of the product (Sabin & Weigel, 1998).

Product configuration systems are a young area of research. The development of product configuration systems started with the research carried out on expert systems in the 1980s where the XCON system at Digital Equipment was the most influential one (see (McDermott, 1982; Barker, O'Connor, Bachant, & Soloway, 1989; McDermott, 1993) for a good description).

The reviewed literature shows a progression from the early development of expert systems to today's use of product configuration systems. It covers many subjects from application of new technology to science (isolated technical systems) to socio-technical systems and even why a user willingly spends 100% more on a configured product than a similar standard product. However, the definition of configuration has become defuse, as it is evident that the different research groups defines configuration differently.

Recent research on configuration of products, how to develop product configuration systems, and how configuration systems are applied in organizations was explored. Although the research shows an interesting interplay between the user and the configuration system, there is little user-focused research on this topic. What is more, knowledge on how to integrate configuration systems from an organizational point of view in existing sales systems is absent. Publications focus for instance on how organizations implement and use toolkits, not on how users interacting with them.

To become more specific in relation to how configuration systems can support the business of companies this paper will try to create a frame of reference, which defines key concepts and definitions related to configuration. Finally this paper identifies different kinds of product configuration systems and describes how the different systems differ. The remainder of the paper is structured as follows: Section 2 defines the basic concepts of configuration, while section 3 defines a product configuration system's relation to other systems by description of a reference frame. Finally, section 4 & 5 discuss and conclude on the findings of present paper.

2. Basic concepts

In order to create a frame of reference for configuration, it is essential to clarify the definitions of the most important concepts. As the review of existing literature has pointed out, we have several definitions and perceptions of configuration. The term is loosely used in many different contexts. Unfortunately, not only the term *configuration* carries an ambiguous meaning it is a more widespread problem. This section aims to clarify the definitions of the most important terms and concepts related to product configuration.

"Product configuration systems configure a product on the basis of a formalised product model." This sentence contains the four most central concepts of this paper: (i) A product i.e. the technical system being configured, (ii) configuring i.e. the configuration task, (iii) a product model, and (iv) a configuration system. We will start out by defining these four terms in section 2.1. through 2.4. In section 3. we will try to understand the broader context of a configuration system.

2.1. The product (or technical system)

The following is based on the theory of technical systems by Hubka and Eder (1988). Therefore, in this paper we define products as technical systems (TS): **Technical systems** are objects, products, things, machines, implements, technical objects, etc. which are made by humans to fulfil a specific need.

In other words, TS refer to all types of human artefacts. Humans have many varying needs, and for centuries philosophers and sociologists have discussed the nature of these needs, and how we prioritise them. The essential point is that, in general, people will tend to formulate their needs in terms of existing technologies (Hubka & Eder, 1988). In this way, an introduction of a novel product on the market can change our perception of our needs.

If the means to fulfil a need exist at the time, if the need can be realised, then a process of designing and manufacturing a product (a TS) can supply the means to fulfil that need. As TS have no intention and cannot fulfil needs of humans by themselves, it is necessary to describe the process where the TS are applied to fulfil a need. Hubka and Eder (1988) define processes in which TS are applied to fulfil a need technical processes. A technical process transforms an operant from an existing state to a desired state by the use of *operators*. Hubka and Eder (1988) describe three kinds of operators that have an effect on the transformation process: (i) Human systems, (ii) technical systems, and (iii) active environments. The *effect* posed on the transformation system can be described as *material*, *energy* or *information*, or any combination of those. A system of operators that transforms an operand through a technical process from an existing state to a desired state is called a *transformation system*.

Each transformation system has a well-defined purpose, which is to perform the intended transformation on the appropriate operands. Hubka and Eder (1988) divide the major elements of the total transformation system into a process (the operand which is being transformed), and the operators that drive and guide the process. The total transformation system can be divided into four subsystems. (i) A technical system (TS - the product), (ii) a human system (HuS – a human operator), (iii) the active environment (AEnv – the influence from the environment), and (iv) a technical process (TP - where an operand is transformed by effects from the three subsystems mentioned above). These four subsystems are depicted in Fig. 1.

The technical process transforms an operand from an existing state to a desired state (the attributes of the operand change). The TP can be described by using a variety of tools. In management science and operations management the methodology mostly frequently used to describe processes is called Business Process Modelling Notation (BPMN) (White, 2006).

TS can be described as being fully deterministic (Hubka & Eder, 1988). However, in many cases, the causes are so complex and with such a multitude of interactions that it is difficult to assign a cause to each consequence (this is particularly significant when humans or an active environment is involved).

The purpose of the TS is represented by the system of its output effects to the technical process. The actual abilities of the TS are referred to as functions. Functions represent an introvert view of how the effects of the TS are derived at.

So, the constituents of TS are fitted together so that a given input will lead to a given output in the shape of effects on the operand. In order to obtain a certain result (i.e. an output effect), various phenomena are linked together in an action chain (the TS-internal processes).

According to Hubka and Eder (1988) the TS can be described as three structures: a functional structure, an organ structure, and a component structure. The three TS structures (function, organ, and component) are different views or representations of the same TS at different levels of abstractions (Hubka & Eder, 1988).

2.2. Configuration task

There is a fine line between product configuration and design. Many refer to configuration as a subclass of the design activity. Traditionally, configuration has often been considered a part of the design science discussion, even a special case of the general field of design activities. In Stumptner (1997) it is assumed that in configuration the design goals and requirements are fully specified, and that the subcomponents and functions are already known. This belief is also supported by Sabin & Weigel (1998, pp. 42-43), who note: "…product configuration is informally a special case of design

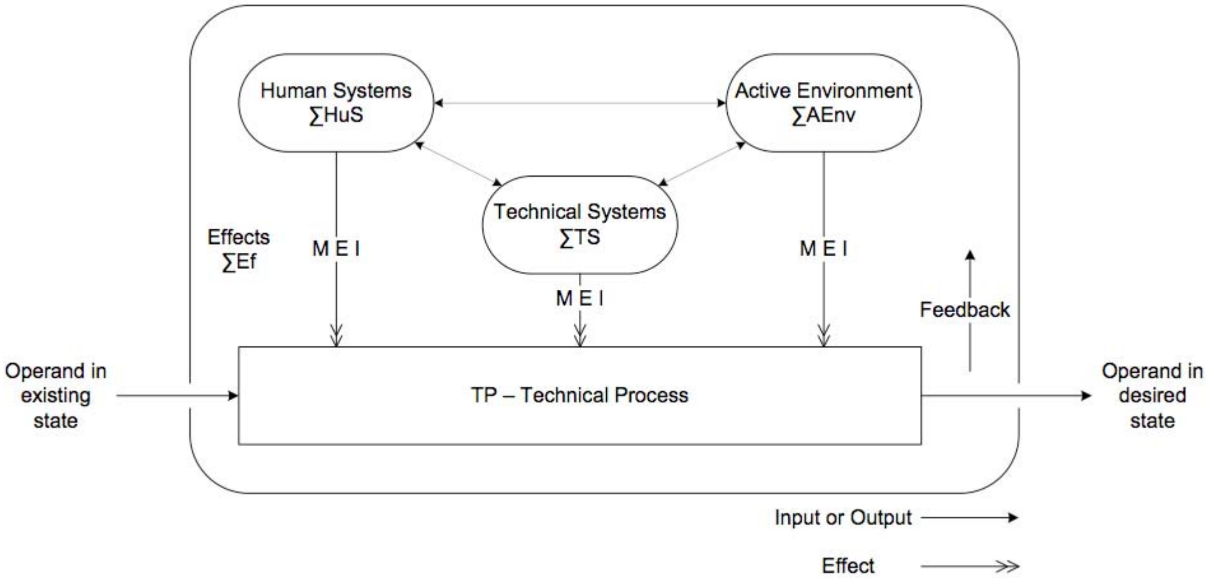

Fig. 1: Model of the transformation system (Hubka & Eder, 1988, pp.35)

activity and consists of two key features: a) the artefact being configured is assembled from instances of a fixed set of well-defined component types, and b) components interact with each other in predefined ways."

The most precise definition of configuration is given by Mittal & Frayman (1989), and even that definition is not unproblematic. Before going into the problems, let us revive the definition: According to Mittal & Frayman (1989) configuration is:

"**Given:** (A) a fixed, pre-defined set of components, where a component is described by a set of properties, ports for connecting it to other components, constraints at each port that describe the components that can be connected at that port, and other structural constraints (B) some description of the desired configuration; and (C) possibly some criteria for making optimal selections.

Build: One or more configurations that satisfy all the requirements, where a configuration is a set of components and a description of the connections between the components in the set, or, detect inconsistencies in the requirements." (Mittal & Frayman, 1989, p. 1396)

Thus the core of the configuration task is selecting and arranging combinations of pre-defined parts that satisfy given specifications. No new component types can be created nor can the interface of the existing components be modified. The configured solution must provide a list of selected components, describe the product structure and topology of the product. (Mittal & Frayman, 1989, p. 1396) This is also supported by Sabin & Weigel (1998).

Felfernig's group at Klagenfurt has a similar approach or definition to the configuration task. However, they add that attributes can have variable values.(Felfernig, Friedrich, & Jannach, 2000; Felfernig, 2007)

The group at Helsinki University of Technology and Science also provide a definition, which is in line with the definition of Mittal & Frayman in (Soininen et al., 1998, p. 357).

At the Technical University of Denmark, Anders Haug provides a definition of the configuration task, which uses the term *entity* instead of the term *component* in order to escape the world of physical products(Haug, 2007, p. 18).

The group around Forza and Salvador also complies with the definition of Mittal & Frayman (1989). However they note: "A pre-defined component is to be meant as either a standard component, or a standard component with variants or a parametric component, i.e. a component for which one or more attributes vary continuously" (Salvador & Forza, 2004, p. 275)

As Brown (1998) points out, there are some problems with the definition of configuration Mittal & Frayman (1989). The use of the term 'connect' throughout the definition indicates that the components in the configuration actually physically connect. Brown (1998) indicates that this is not always the case, and that a logical explanation of the use of 'connect' is that Mittal & Frayman have a background related to the configuration of computer equipment. Configurations are more often about relationships between components, where "touch" and "connect" are examples of relationships.

As with the term 'connect', there is also an issue with the use of the term 'ports'. While the term 'port' seems logical when configuring computer systems (i.e. a motherboard has ports for connecting with other kinds of hardware), it is more complicated to make a meaningful explanation of the term 'port' when configuring mechanical products. The use of the term is again connected with the idea of components which physically connects, and again this might no be valid for all configuration problems (Brown, 1998).

Another issue is the term 'pre-defined components'. It is not clear in the definition of Mittal and Frayman (i) at what level of abstraction the components have to be predefined, and (ii) whether all or just some components have to be used in the configuration. If the components allow additional refinement of attributes in terms of colour, shape, dimension, material, etc., there is a possibility of creating something new, which again conflicts with the first point of the definition of Mittal and Frayman. This problem is most outspoken for dimensional refinements. An abstraction of the shape of a component might be specialized into a square or a variety of different rectangles. This might affect the components' relationship with other components, changing how the components connects or touch other components. Hence there is a contradiction between point 1 and point 2 in the definition of Mittal & Frayman, as refining or concretising an abstract component can modify the component's allowed connections. (Brown, 1998)

This illustrates the point that configuration is on the edge of design. By allowing random refinement or the concretising of abstract components, we are on the edge of the class of problems we can describe as configuration. One must refine the definition of configuration.

To refine the understanding of configuration one can find inspiration in (Brown, 1998) and in (ten Teije, van Harmelen, Schreiber, & Wielinga, 1998).

From now on we will use the term ‘configuring’ when referring to the process itself, ‘configuration’ when referring to the output of the process, and ‘configurator’ or ‘configuration system’ when referring to the system that supports the configuration task.

The definition of Haug (2007) builds upon the definitions of Mittal & Frayman, Sabin & Weigel, and Soininen et al. However, it is necessary to point out that ‘configuration’ is equalled with the term ‘configuration task’. In the terminology of the present paper, a configuration is the output of the configuration task i.e. it is a description of a product that satisfies the given requirements. In this way, according to this paper, the definition of the configuration process is:

Configuration process: To combine predefined entities (physical or non-physical) and define their variable properties, while obeying constraints and legal interface combinations in a way that satisfies given requirements.

Logically, the definition of configuration can be derived from the definition of the configuration process. Taking the definition of Mittal and Frayman, Sabin and Weigel, and Soininen et al. into consideration it is necessary to point out that the component structure of the product has to be specified including any connections between components.

A **configuration** is the output of the configuration process, e.g. a description of the component structure of the product and any connections between the components in the set or a description of inconsistencies in the requirements.

2.3. Product model

The term *product model* has been commonly used in the literature related to product configuration systems, and a number of different definitions of a product model can be found, as illustrated below:

“…the product model is defined by a total set of characteristics, defining the transformation, function, organ and component structures of a machine system.” (Andreasen, 1994, p. 111)

“A model containing a description of the product’s functional and structural design” (Hvam, 1999, p. 79)

“A product model is an abstract representation or description, describing (a) the structure of P and (b) facts, object, concepts and properties that are relevant in any life cycle phase of P. P can be a single product or a family of products[1]. A product is a thing, a substance or a service produced by a natural or artificial process.” (Schwarze, 1996, p. 33)

“A Product model is usually intended to define various data generated through the product life cycle from specification through design to manufacture” (Shaw, Bloor, & de, 1989)

According to Krause et al. (1993) there are four kinds of product models. The kernel in **structure-oriented product models** is a description of the products’ structure. The structure of a product can be represented in many ways, e.g., bill-of-material types, UML, product family master plans. The purpose of **geometry-oriented product models** is representing the shape of the product, through wire frames, surface, solid, or hybrid models (often referred to as CAD). Since the data structures of geometrical models are designed to represent geometry it is often difficult to extend these with additional product information. The **feature-oriented product model** is an extension of the geometric-oriented product model. The feature-oriented product model represents often-used shape patterns as geometric items, called form features. **Knowledge-based product models** are often described as a model of the product based on artificial intelligence techniques such as object-oriented programming, rule-based reasoning, constraints, logical systems, etc

Hvam, Mortensen, and Riis (2008) provide a description, a framework, of relevant product models in a configuration context. The framework describes product models in terms of three different types of models: (i) Property models that describe the properties and the functions of the product, (ii) product structure models that describe the organs and parts of the product, and (iii) models of the product’s meeting with life cycle systems. The structure of the framework is inspired by the ‘theory of domains’ by Andreasen (1992; 1994; 1998).

In conclusion, there is an important thing to observe about the definition of product models. The product model in a configuration system must be a constitutional genetic model capable of instantiating many different configurations. For this reason the following definition of Schwarze’s is the most adequate: “A **product model** is an abstract representation or description, describing (a) the structure of P and (b) facts, object, concepts and properties that are relevant in any life cycle phase of P. P can be a single product or a family of products. A product is a thing, a substance or a service

[1] A product family is a set of products which differ only in a limited number of less important features (Schwarze, 1996, p. 33).

produced by a natural or artificial process." (Schwarze, 1996, p. 33)

2.4. Configuration systems

Having described the configuration task and the product knowledge formalised in a product model, the next step is to look at software tools – the product configuration system. In the literature the following three terms often are used in order to define such IT-systems: Configurator, product configuration system, and configuration system. These are often used interchangeably. As Haug (2007) notes, this would not represent a problem if there was consensus on using the three terms interchangeably. However, as the following quotation illustrates, by configuration system e.g. Forza and Salvador mean more than merely the software application: "Configuration system: The set of human and computing resources that contributes to accomplishing the configuration and modelling processes" (Forza & Salvador, 2007)

Therefore, it is important to strictly define what is meant by configuration system. In the present paper, the terms "configurator," "configuration system," and "product configuration system" will be used interchangeably, and all the terms refer to the software application. The social-technical system that Forza and Salvador refer to is in the present thesis referred to as a 'total configuration system' (see section 3 for further explanations).

Configuration systems are often described as belonging to expert systems or knowledge-based systems. Although it is argued that expert systems are a subset of the more general knowledge-based systems (Jackson, 1999; Hopgood, 2001), expert systems are typically defined as computer programs that represent and reason with knowledge of specialist matters with the purpose of solving problems or giving advice (Jackson, 1999). A knowledge-based system is defined more broadly as a computer system, which is programmed to imitate human problem-solving by means of artificial intelligence and with reference to a database of knowledge on a particular subject.

According to Hopgood (2001), the principal difference between a knowledge-based system and a conventional program lies in the structure: "In a conventional program, domain knowledge is intimately intertwined with software for controlling the application of that knowledge. In a knowledge-based system, the two roles are explicitly separated." (Hopgood, 2001, p. 2)

The goal of a configuration system is to build a specification in which a selection of components satisfy the needs of the configurer or to detect inconsistencies in the requirements given by the user. In this case, the definition of configuration system given by Haug (2007) is accurate: "A **product configurator** is a software-based expert system that supports the user in the creation of product specifications by restricting how predefined entities (physical or non-physical) and their properties (fixed or variable) may be combined" (Haug, 2007, p. 19)

As Haug (2007) notes, configuration systems should not be mistaken with systems that are capable of combining components without any restrictions.

3. Total configuration system

We define the total configuration system (TCS) as the configuration system including the context in which the configuration system operates. Forza & Salvador (2007) define a configuration system as the "set of human and computing resources" needed to "accomplish configuration and modelling processes". This definition correctly points out that the human systems influence the configuration task, and that human systems can choose to use a configuration system to support them, or they can choose not to be supported by a configuration system.

While Forza & Salvador (2007) describe two subsystems (the human system, and the computing system), the theory of technical systems delineates three important elements in a total transformation system: (i) A process, (ii) an operand which is being transformed, and (iii) the operators which drive and guide the process (Hubka & Eder, 1988). If one applies the same logic to a configuration system, the total configuration system also consists of an operand, operators, and a process.

These three key elements of the total configuration system will be described in the following three sections.

3.1. Understanding the operand of the TCS

In order to perceive product configuration systems as technical systems it is important to define the operand. The operand is a passive member of the TCS, and is the description of a product (TS). In the existing state it is a description of the product at a high abstraction level yet it still describes the needs of a customer. The operand in the desired state is a configuration of a product. Accordingly, the TCS is defined as the total system, which transforms the need of a customer to a specification of the product's component structure. In other words the TCS transforms a description of the requirements of a product (on a high abstraction level) to a description of the product that satisfies the requirements (which are given on a concrete level).

What characterises the description of a product in the existing state? One characteristic is that the customer is aware that he needs a product to perform, apply, or do something with. The product can be described using different levels of abstraction as previously described in section 2.1.

The Customer's knowledge of the TS or the product he wishes to configure/buy is not complete. The customer is only rarely able to describe the product completely on any one of the different abstraction levels. Usually (especially when talking about complex products) the customer has knowledge about the product on different abstraction levels. He might have complete knowledge of the purpose that the product has to fulfil, some knowledge about the function structure, no knowledge of the organ structure, and a perhaps a bit of knowledge about the component structure of the product.

The goal of the configuration process is to concretise the user's understanding of the product, that is to satisfy the configuration task. In other words, the configuration process combines predefined entities (physical or non-physical) and defines their variable properties while obeying constraints and legal interface combinations in a way that satisfies given requirements.

Hence on must focus on the conversion of a description of needs for a product into a specification of a product which fulfils these needs. What is changed by the total configuration process is the description of the product, and thus, the operand is defined as the description of the product. Consequently, the operand in the existing state is the description of the product before it is configured, and the operand in the desired state is the description after the product has been configured. This is the process which is typically called configuring, configuration process or configuration task, and it is the theme of section 3.3.

3.2. Understanding the operators of the TCS

Operators guide and drive the configuration process. One can identify the following classes of operators in the total configuration system (the references given in parenthesis are to the theory of the technical systems): (i) Users (human system), (ii) product configuration system (technical system), and (iii) organization (active environment)

In configuration literature all three classes of operators have traditionally been perceived as rational operating systems that act in a rational or at least a predictable way. Often this has been justified by referring to general system theories such as the one by Bertalanffy (1950; 1972). As it is not always sufficient to understand all three systems as rational systems, they will be described in the following three sections.

3.2.1. Understanding the user as an operator. The knowledge needed to solve the configuration task depends on the abstraction level of the user. If the user has extensive knowledge of the product, he might wish to configure the product on a fully structural level, selecting and configuring components. A user with less product knowledge might wish to configure the product on a more functional level, ensuring that the product gets the desired functionality thus paying no attention to the components used, as long the desired functionality is delivered.

The user interface affects how the product configuration system is accepted by the users. Andreasen (1994) describes the relation between man and machine as the man-machine interface, and notes that it is vital that the designer of the machine decides which tasks are technical and done by the machine, and which tasks are done by the operator. Likewise, the relation between the user and the product configuration system is carried out through the user interface, and according to Beyer & Holtzblatt (1998) it is important to consider how the work model of the system can be aligned to the work model of the user.

3.2.2. Understanding the product configuration system as operator. The basis of any product configuration system is knowledge; knowledge of the product that is configured, and knowledge about the process that the configuration system supports. The product knowledge needed to solve the configuration task depends on the abstraction level of the user and the abstraction level at which the customer presents his needs as input to the configuration process. If the user has a high degree of knowledge of the product, he might wish to configure the product by picking and configuring components. A user with less product knowledge might wish to configure the product on a more functional level, ensuring that the product gets the desired functionality without paying any attention to the components, which are used.

The foundation of the product configuration system is an abstract model of the product (a product model) that can transform user requirements into a concrete component structure of the product. The effect delivered to the configuration process can be divided into three different effects that a product configuration system exerts onto the configuration process: (i) Concretising knowledge about the product, (ii) abstracting knowledge about the product, and (iii) validating knowledge about the product

3.2.3. Understanding the organization as Operator. The active environment of the configuration process is comprised of actors, technical systems, and structures. The active environment is the part of the total environment, which has a direct relationship to the product configuration system, which is being analysed. The active environment is of course dependent upon how the total configuration system is defined. A product configuration system is implemented in an organization. So in order to understand how configuration systems are applied in companies in general it is important to be able to describe organizations and how they relate to product configuration systems. Since the XCON project it has been known that any technological development will result in an organizational change. Leonard-Barton (1991) concludes, that the greater the benefits to be realised from the introduction of a new technical system, the greater the amount of organizational change that should be anticipated. While this sounds like a daring conclusion, there is still a grain of truth in the conclusion. If you change and rationalize work (with or without the use of technology) it requires management of the mutual adaptation of technology to organization and organization to technology during design and development (Leonard-Barton, 1987b; Leonard-Barton, 1987a; Leonard-Barton, 1991).

In order to describe the organization in the total configuration system we will look at the different pulls that exist from different parts of the organization. The purpose of doing so is to identify the motivation for development of the product configuration system. Is the system developed to sustain the power of the technostructure in the organization or is it developed to reduce the power of the professionals by formalising their expert knowledge? Nevertheless the answer to this question a product configuration system is an instrument to support or diminish a given pull in the organization.

A configuration system might have a successful implementation phase and then slowly deteriorate because the experts are not willing to help maintain the system. There might be many explanations to this case but if one uses the structures in fives by Mintzberg (1980; 1993) the explanation would be: The strategic apex needs a reduction in the power of the experts in the organization, and it initiates a move towards a structure that depends more on standardisation. This motivates the company to build a product configuration system. Apparently, the system is a success, but behind the scene there is an even bigger pull to professionalise rather than standardise the work. This leads to the slow deterioration of the knowledge in the product configuration system, the knowledge in the system eventually becomes invalid, and the use of the system is dramatically reduced.

3.3. Understanding the configuration process of the TCS

An important lesson learned from the theory of technical systems is that the technical process cannot be designed. The only thing, which a designer can be totally in control of and design, is the technical system. Likewise the configuration process cannot be designed either. The only thing one can control, and which behaves in a deterministic way, is the product configuration system. If it is presumed that humans do not behave entirely rational but instead operate within the confines of a bounded rationality, it is difficult to argue that one must start a configuration project by designing the configuration process. Actually, this fact can be extended to the level of the organization, as the organization also acts with bounded rationality.

However, when you start a configuration project, you can organise the user interface of the product configuration system and design the system by using user-centred development techniques such as contextual design as presented by Beyer & Holtzblatt (1998). In that way, you can motivate the users to perform a sequence of tasks in a given order.

4. Discussion

The frame of reference together with the definition of key concepts is important for understanding the use and implementation of product configuration systems. The total configuration system describes the operators of the total configuration system as all the sums of the following operators: Users, Product configuration system, and Active environments. Active environments can be other IT-systems or environments such as an organization. Above the active environment is illustrated as the organization. All operators in the total configuration system have an effect on the configuration process. The illustration of the total configuration system is heavily inspired by the 'The Theory of Technical Systems' by Hubka and Eder (1988).

An interesting implication of the proposed reference frame is that it is not possible to look at the development of a product configuration system as being a merely a technical development project. The configuration process cannot be designed, it can only be supported. Starting a configuration project by modeling the as-if and to-be state of the configuration process does not per se lead to a successful configuration project.

A consequence of the model is that one must change three systems to change the configuration process if a product configuration system is applied in a company. Naturally you have to develop the configuration system itself. Next you have to develop your organization or at least carry out some kind of change management. Finally, you have to develop or train the users. An interesting observation from the theory of technical systems is that you cannot design how the human system applies the technical system. You can only design the technical system not the technical process. Of course you can do your best in guiding and training the human system but you cannot be sure that the human system acts as intended or trained. Consequently, it is supposed that the implementation of configuration systems requires as strong a focus on training of users, and change management as on the technical side of developing the product configuration system itself.

It is not enough merely to understand the configuration system in order to understand how a given product is configured. Other actors or operators than the configuration system affect the configuration process.

The work made in creating the frame of reference is purely theoretical. It is based upon papers and books from the configuration and design science community. The reference frame presents coherent definitions on key concepts related to configuration, and it offers an explanation of how the key concepts interrelate.

The purpose is to make the meaning of key terms clear, and to make it explicit how configuration systems and their connection to the configuration process and other systems could be perceived.

5. Conclusion

The present paper develops a frame of reference that highlights the organization’s role in the configuration process. The frame of reference contains definitions of the following key concepts of configuration related to configuration system: Product (technical system), configuration task, product model and configuration system. These key concepts are used to derive to the model of the total configuration system, which describes a configuration process that transforms a product specification to a concrete description of the product by the active involvement of users, configuration system(s) and active environments.

The purpose of this paper was to make the meaning of key concepts clear, and to make it explicit how configuration systems and their connection to the configuration process and other systems could be perceived. The proposed frame of reference helps in avoiding the pitfall of ambiguity when discussing how configuration systems are applied in industry.

6. References

Andreasen, M. M. (1992). *Designing on a 'Designer's Workbench' (DWB).* Paper presented at the Proceedings of the 9th WDK Workshop.

Andreasen, M. M. (1994). Modelling - The Language of the Designer. *Journal of Engineering Design*, *5*(2), 103-115.

Andreasen, M. M. (1998). *Conceptual Design Capture*. Paper presented at the Proceedings of EDC '98.

Barker, V. E., O'Connor, D. E., Bachant, J., & Soloway, E. (1989). Expert Systems for Configuration at Digital - Xcon and Beyond. *Communications of the Acm*, *32*(3), 298-317.

Bertalanffy, L., Von. (1950). An Outline of General System Theory. *British Journal for the Philosophy of Science*, *1*(2), 134-165.

Bertalanffy, L., Von. (1972). The History and Status of General Systems Theory. *Academy of Management Journal*, *15*(4), 407-426.

Beyer, H., & Holtzblatt, K. (1998). *Contextual Design - Defining Customer-Centered Systems.* Morgan Kaufmann Publishers.

Brown, D. C. (1998). Defining configuring. *(AI EDAM) Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, *12*(4), 301-305.

Edwards, K., & Ladeby, K. (2005). *Framework for Assessing Configuration Readiness*. Paper presented at the Proceedings of the 3rd Interdisciplinary World Congress on Mass Customization and Personalization.

Edwards, K., & Riis, J. (2004). *Expected and Realized Costs and Benefits when Implementing Product Configuration Systems*. Paper presented at the Proceedings from the International Design Conference, Design 2004.

Felfernig, A. (2007). Standardized Configuration Knowledge Representations as Technological Foundation for Mass Customization. *Engineering Management, IEEE Transactions on*, *54*(1), 41-56.

Felfernig, A., Friedrich, G., Jannach, D., & Stumptner, M. (2004). Consistency-based diagnosis of configuration knowledge bases. *Artificial Intelligence*, *152*(2), 213-234.

Felfernig, A., Friedrich, G. E., & Jannach, D. (2000). UML as domain specific language for the construction of knowledge-based configuration systems. *International*

Journal of Software Engineering and Knowledge Engineering, *10*(4), 449-469.

Fleishanderl, G., Friedrich, G. E., Haselbock, A., Schreiner, H., & Stumptner, M. (1998). Configuring large systems using generative constraint satisfaction. *IEEE Intelligent Systems*, *13*(4), 59-68.

Forza, C., & Salvador, F. (2007). *Product Information Management for Mass Customization.* New York: Palgrave Macmillan.

Haug, A., Ladeby, K., & Edwards, K. (2009). From Engineer-To-Order to Mass Customization. *Management Research News*, *32*(7), 633-644.

Haug, A. (2007). *Representation of Industrial Knowledge - as a Basis for Developing and Maintaining Product Configurators*.

Hopgood, A., A. (2001). *Intelligent systems for engineers and scientists* (2nd ed.). Boca Raton, Florida: CRC Press LLC.

Hubka, V., & Eder, W. E. (1988). *Theory of Technical Systems - A Total Concept Theory for Engineering Design* (2nd edition). Berlin: Springer-Verlag.

Hvam, L. (1999). A procedure for building product models. *Robotics and Computer-Integrated Manufacturing*, *15*(1), 77-87.

Hvam, L. (2006). Mass Customization of process plants. *International Journal of Mass Customisation*, *1*(4).

Hvam, L., Malis, M., Hansen, B., & Riis, J. (2004). Reengineering of the quotation process: application of knowledge based systems. *Business Process Management Journal*, *10*(2), 200-213.

Hvam, L., Mortensen, N. H., & Riis, J. (2008). *Product Customization.* Berlin Heidelberg: Springer-Verlag.

Jackson, P. (1999). *Introduction to expert systems* (3rd). Essex: Addison Wesley.

Krause, F. L., Kimura, F., Kjellberg, T., Lu, S. C.-Y., van, d. W.,A.C.H., Ating, L. et al. (1993). Product modelling. *CIRP Annals*, *42*(2), 695-706.

Larsen, B. D., Ladeby, K., & Gjøl, M. (2006). *Conceptualizing Pharmaceutical Plants*. Paper presented at the Proceedings of Virtual Concept 2006.

Leonard-Barton, D. (1987a). Implementing structured software methodologies: a case of innovation in process technology. *Interfaces*, *17*(3), 6-17.

Leonard-Barton, D. (1987b). The Case For Integrative Innovation: An Expert System at Digital. *Sloan Management Review*, *29*(1), 7-19.

Leonard-Barton, D. (1991). The role of process innovation and adaptation in attaining strategic technological capability. *International Journal of Technology Management*, *6*(3-4), 303-320.

McDermott, J. (1982). R1: A rule-based configurer of computer systems. *Artificial Intelligence*, *19*(1), 39-88.

McDermott, J. (1993). R1 ("XCON") at age 12: lessons from an elementary school achiever. *Artificial Intelligence*, *59*(1-2), 241-247.

Mintzberg, H. (1980). Structure in 5's: A Synthesis of the Research on Organization Design. *Management Science*, *26*(3), 322.

Mintzberg, H. (1993). *Structure in Fives - Designing Effective Organizations.* Upper Saddle River, N.J.: Prentice Hall.

Mittal, S., & Frayman, F. (1989). Towards a generic model of configuration tasks. *IJCAI-89 Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 1395-1401.

Moser, K. (2007). *Mass Customization Strategies.* Lulu Enterprises.

Sabin, D., & Weigel, R. (1998). Product configuration frameworks-a survey. *IEEE Intelligent Systems*, *13*(4), 42-49.

Salvador, F., & Forza, C. (2004). Configuring products to address the customization-responsiveness squeeze: A survey of management issues and opportunities. *International Journal of Production Economics*, *91*(3), 273-291.

Schwarze, S. (1996). *Configuration of Multiple-Variant Products - Application Orientation and Vagueness in Customer Requirements.* BWI Betriebswissinschaftliches Institut, vdf Hochschulverlag AG an der ETH ZÅrich BWI Betriebswissinschaftliches Institut.

Shaw, N. K., Bloor, M. S., & de, P.,A. (1989). Product data models. *Research in Engineering Design*, *1*(1), 43-50.

Soininen, T., Tiihonen, J., Männistö, T., & Sulonen, R. (1998). Towards a general ontology of configuration. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AIEDAM*, *12*(4), 357-372.

Stumptner, M. (1997). An overview of knowledge-based configuration. *Ai Communications*, *10*(2), 111-125.

ten Teije, A., van Harmelen, F., Schreiber, A. T., & Wielinga, B. J. (1998). Construction of problem-solving methods as parametric design. *International Journal of Human-Computer Studies*, *49*(4), 363-389.

Tseng, M., M., & Piller, F., T. (2005). *The Customer Centric Enterprise - Advances in Mass Customization and Personalization.* Berlin: Springer Verlag.

White, S., A. (2006). Introduction to BPMN.