Specification and Configuration of Customized Complex Products

Hamideh Afsarmanesh and Mohammad Shafahi

University of Amsterdam, Informatics Institute, FCN Group,
Science Park 904, 1098 XG Amsterdam, The Netherlands
{h.afsarmanesh,m.shafahi}@uva.nl

Abstract. This paper addresses the design of an information system for specification of complex configured products, such as the solar power plants or large intelligent buildings, which by nature are designed, constructed, installed, operated, and maintained through virtual consortium of enterprises. In other words, typically a number of virtual organizations are involved during the entire life cycle of these products. The involved companies include the equipment suppliers, business service providers, and tailored software system developers, that form a consortium typically coordinated and managed by an EPC (Engineering, Procurement, and Construction) contractor or a so called project developer. Furthermore, remote access is required through the cloud to different elements of the complex product's specification, to effectively support these products. An approach and a set of mechanisms are introduced in the paper for effective cloud-based specification of such complex products. The addressed generic system supports different involved stakeholders with customization of the planned complex product, satisfying both their preferences and mandatory standard criteria. The specification system is being implemented as a generic pilot information system, supporting iterative specification of configured sub-products for the planned complex product. This information system also constitutes the base for the next step of this research, focused on semi-automating the process of sub-products cataloging and building an intelligent recommender for complex product configuration, in dialogue with the user.

Keywords: Complex Product Specification, product information systems, complex product configuration.

1 Introduction

Product configuration has been addressed in the areas of computer science and systems engineering and in research related to several domains, and particularly in manufacturing [1]. A number of commercial product configurator software systems exist today in the market, many of which categorize themselves as user-friendly tools for configuration of customized products. Every such system however, contains its own peculiar set of functionality and a proprietary design to support certain specific domain requirements. Furthermore, besides being costly and proprietary, most

L.M. Camarinha-Matos and R.J. Scherer (Eds.): PRO-VE 2013, IFIP AICT 408, pp. 81–90, 2013.

existing commercial systems are specialized in specific domains, e.g. manufacturing of customized products, such as cars, boats, bicycles, etc.

This research however aims differently, at the design and development of a *generic complex product information management system for the specification of service-enhanced configured complex products, exemplified by solar power plants and intelligent buildings.* There is a lack of standard definitions for such complex products and their components, as well as for the standard functionality and features that need to be supported for their configuration. Typically these products involve large number of stakeholders participating in different virtual organizations that are formed to support these complex products during their entire long life cycle of more than 20 years. For instance, their design and construction stage typically takes about one year, and their specification and customization is achieved incrementally involving a number of stakeholders. Furthermore, the participating stakeholders are typically dispersed geographically, sometime in different continents, and thus need to share their requirements and other information detailing the specification of these complex products through the cloud.

We have designed a generic information management system in support of our targeted service-enhanced complex products. Later on, in the conclusion section of this paper, we further reflect on the comparison between our developed features in the information system versus the main functionality supported by the existing commercial products.

One commonly referenced definition for the product configuration process is provided in [2]:

> "**Given:** (i) 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 (ii) some description of the desired configuration; and (iii) 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."

Considering the increasing rate of competition among the companies in the market, as well as the variety and number of involved stakeholders, the role of presenting the customer with a specification of the configured complex product that best meets his/her requirements and at the lowest possible cost, has become increasingly significant.

As the first main step in this configuration process, in order to make the process less complex for the involved stakeholders, there is a need to provide users with effective and iterative mechanisms and tools for specification of their product. As such, the complex product specification tool must also support the iterative and incremental nature of definition of complex product configuration.

To help stakeholders with co-configuring and/or ordering complex products, or their related sub-products, in this research we aim to design a generic reusable approach and mechanisms supporting stakeholders with effective customization of complex products, based on their preferences and/or mandatory criteria.

For the purpose of designing the complex product specification system, as the base we have mainly used the following: the requirements analysis of complex products, such as the PV (Photo Voltaic) power plants, as addressed in [3], the set of designed business scenarios addressed for this domain in [4], as well as the introduced information/knowledge provision services addressed in [5]. The important features aimed by the design of this complex product specification system are summarized below:

- Generality and Reusability – The designed meta-data (schema) for the database system, for storing complex product specification details, must be generic. The aimed generality/reusability is in the sense that it should be able to model any entity and concept belonging to any complex product, and not dedicated to supporting only one or two specific complex products.
- Usage Simplicity – While designed functionality for complex product specification system can be complex, designed interfaces for using these functionalities must hide system complexity and provide easy to use operations.

2 Complex Product and Its Atomic and Composite Sub-products

A complex product is composite and consists of a set of sub-products. A sub-product may represent a specific piece of equipment, a business service, or a software-based service, which is produced and/or provided in the market and is used for generating the complex product. Sub-products can also be composite, in the sense that they may consist of other sub-products. For example an intelligent building is a complex product, including a large number of equipment sub-products, such as an intelligent thermostat, while the intelligent thermostat itself consists of other equipment e.g. including a thermometer, etc.

The term Sub-product refers to all constituents of the complex product, being themselves atomic or composite. Furthermore, sub-products have a set of known characteristics, which we model through ***feature-kind*** in this paper, representing the "kind/type" of their characteristics*/features*. For example, a Pyrnometer can be characterized by its following feature-kinds. Each feature-kind below is also exemplified inside parenthesis by one specific feature-instance for this sub-product:

- Manufacturer (such as iPLON company)
- Minimum light sensitivity (such as 7µV/W/m2)
- Maximum light sensitivity (such as 14µV/W/m2)

Aggregation of a set of features identify/form, a *product* while the aggregation of a set of feature-kinds identify/form an ***object-class***. As such, each real physical sub-product is usually an instance of several object-classes, where each may characterize a different set of its feature-kinds.

3 Feature-Kind and Features

A feature-kind is a characteristic of a sub-product, and may have multiple scales. One specific value together with the ***scale*** for a feature-kind of a sub-product constitutes a tuple that we call a ***feature*** *of that sub-product.* The combination of all the features of each sub-product makes that sub-product unique.

For example the atomic sub-product lamp has the feature-kinds of: input voltage, amount of energy it consumes, amount of light it produces, and the color of its light beams. But the definition of *one specific lamp* is not unique until we indicate that it has an input voltage of 220 volts, consumes 100 watts of energy and provides 800 lumens of blue light. Therefore, while the latter definition of lamp may fully represent one real lamp, traditionally called an instance of the type lamp, the former definition of lamp only represents meta-data for type lamp in traditional database descriptions.

4 Separating Offer Information from Sub-product Information

An ***offer*** is either an agreement discussed between two stakeholders, or a special deal on a sub-product suggested by one stakeholder to the public (i.e. all potential buyers).

In the former case, during the creation phase of complex products, agreements on details of complex products are gradually reached between the project developer/EPC and the customer (or its consultant), constituting *offer under negotiations* between these two entities, related to the configuration of complex product.

Another kind of offer, the latter case, is for instance the specification of a special deal offered by a supplier/seller for a new sub-product. For example, an offer may include a 1MWp cabinet, to be delivered in Paris, on March 15th, with the price of 10000 Euros. Multiple offers could also be defined for one possible sub-product. For example, a sub-product could be offered by a seller with the price of 10 euros if it is delivered in Amsterdam, but 15 euros if delivered in Dusseldorf. Following gives a set of differences between the “sub-product” entity and the “offer” entity, when the offer consists of specification of a special deal. We therefore model these two entities differently in the complex product specification system:

- *Supplier/seller* of a sub-product may create an *offer*, while a *sub-product* itself is produced by its *equipment manufacturer and/or a service provider.*
- *Offer specification* contains information about the supplier/seller, while the *sub-product specification* contains information about manufacturer/provider.
- *Offer specification* contains sale conditions, geographical location for delivery, pricing, payment, etc. (as indicated in the example above), while *sub-product specification* indicates the production aspects and conditions.
- etc.

In our model, we classify both the manufacturers of equipment and the providers of services as the producing/providing ***organizations***.

Although the process of purchasing one sub-product is usually simple, the process of purchasing a complex product is much more complicated. It involves configuring the complex product, which in turn involves the configuration of all its sub-products. And finally it involves configuring an offer for the complex product, so that the customer can approve or reject it. In all the sub-steps for the process of purchasing a complex product the customer should be involved, so that the final result (the final specification of the complex product) meets all his/her requirements and that the final offer is acceptable by the customer.

5 Object-Classes

As addressed before, the feature-kinds of a *sub-product* together represent specific classification for that sub-product in the system. This is similarly true for the feature-kinds of the *offer* in the system. As such, we introduce ***object-class*** as a basic class defined to model the generic categorization of all sub-products, and offers, where each object-class (for a sub-product or offer) has a specific set of feature-kinds.

Furthermore, we can enforce a set of obligatory feature-kinds to sub-products and/or offers. For example, definition of an object-class "Electrical" (a class representing certain kind of sub-products and/or offers) can simplify the search for the sub-product in the category of "Electrical", while at the same time it can enforce the definition of an obligatory feature-kind, such as "voltage" to this class. In other words, any sub-product class that has the object-class "Electrical", must have the "voltage" feature-kind and therefore any instance of that sub-product class must have a feature (e.g. a value) for its voltage.

Figure 1 shows four different example object-classes that are defined for sub-product classes in the PV power plants, and represents a set of feature-kinds that are specified for each of these object-classes, e.g. sensor object-class has a feature-kind called accuracy. Furthermore, object-classes can be used to identify and filter what feature-kinds are relevant to which stakeholder. For example, the feature-kind "life span" might be interesting for an insurance company, but it might not be interesting for another stakeholder, e.g. the government regulator.

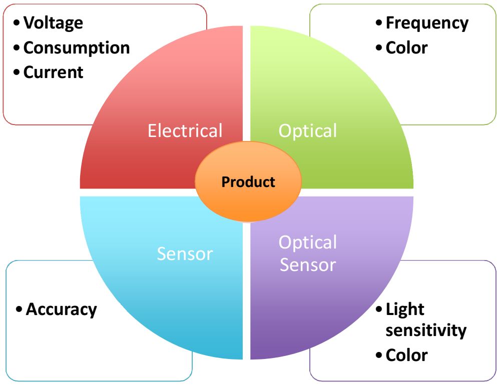


Fig. 1. Example sub-product related to four object-classes and their features

In addition to what is mentioned above, introduction of object-classes provides the following main benefits, both in product and in offer specifications:

- Flexibility to categorize the same product or offer as different kinds of products (e.g. electrical, optical, etc.) and/or different kinds of offers (e.g. Geo-based, etc.) based on a sub-set of feature-kinds of them.
- Guaranteeing that the user will provide the required (obligatory) input for each product and offer through binding feature-kinds to object-classes.
- Ensuring efficient and easy access to product and offer information for different stakeholders by providing views based on object-classes.

6 Relationships among the Entities and Concepts

In order to clearly specify our complex product specification system we model all needed entities and concepts involved in the system and define their interrelationships. Figure 2 provides a summary class diagram (representing only the class names and their inter-relationships). These classes model all the entities and concepts in the complex product specification system, while satisfying the requirement of being generic and reusable. It is important to note that in this diagram the class “Object-class” models classification for both sub-products and offers, while aggregating a set of feature-kinds related to their definition. Furthermore, the “Product” class represents any sub-product of a complex product, as well as ultimately the complex product itself.

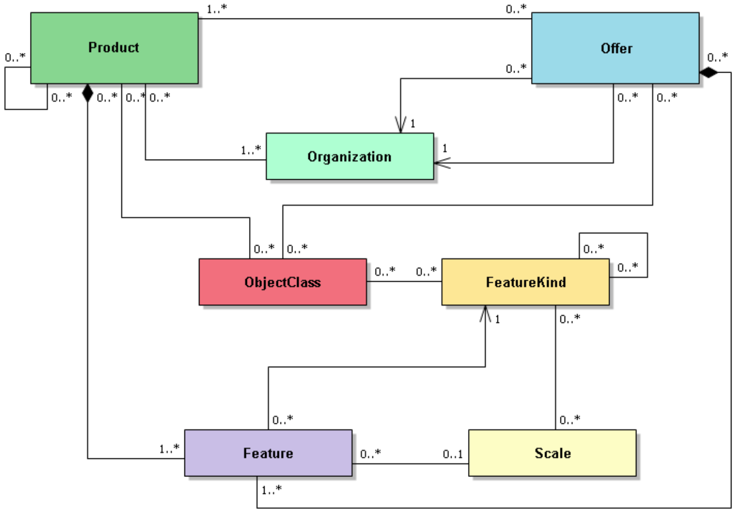


Fig. 2. Summary class diagram of modeled object in complex product specification system

7 Requirements Analysis and Use Cases

The main requirements considered for the design of the specification information system are partially *functional* and partially *non-functional* requirements. Non-functional requirements are those that are not directly related to the functions performed by the system. These include the security, scalability, availability, etc., and are primarily related to software provision constraints and quality requirements. Details of the non-functional requirements for our specification system are outside the scope of this paper.

Functional requirements on the other hand, define the necessary functions expected to be performed by the system, and also describe the input and outputs of the system.

One mostly used tool for documenting the functional requirements of a system is the use case definition. Below we briefly address use cases for the complex product specification system.

Each user of the specification system may interact with the system differently, mostly depending on under which class of stakeholders it falls. In general, the users that are classified as customer or consultant (i.e. customer's consultant) only interact with the complex product specification system in terms of viewing its data and/or approving the data (e.g. offers) made by other stakeholders related to the complex product.

An offer is at the heart of the complex product specification system. It allows the providers of a sub-product (Suppliers for equipment, or providers of the software-based services and business services), to suggest a specific sub-product to the public (including the EPC, project developer, or customer), which is related to the domain of the complex product. It is important to note that for the complex product specification system it is assumed that for viewing and evaluating offers, while frequently the EPC member will act alone (on behalf of the customer), it is also possible that the customer and the EPC members sit together for configuring the plan for the complex product, for viewing the offers related to the complex product, or at least to check certain criteria, e.g. range of price, delivery condition, etc. Then the offer maybe approved or rejected by the relevant stakeholders (e.g. the consultants or the customer).

Please note that in this paper, the EPC/Project developing firm is simply indicated as EPC. Furthermore, the function named as "manage", which is also introduced for different classes in many use cases of complex product specification system, refers to a set of data manipulation functionalities related to those classes. For instance "manage offers" addresses the functionality: *add, edit, view,* and *change status of* the offer class. Below the main use case for the complex product specification system is presented, addressing the functionality related to managing the offers.

The stakeholders that can create an offer include the suppliers, sellers, and the EPC.

Each supplier/seller is in charge of managing *(add, edit, view)* its own offers, which are the offers that are made by it, (i.e. offers related to sub-products of a complex product). Furthermore these offers are usually used by the EPC for making a more composite offers related to the complex product.

On the other hand the EPC may be in charge of managing the complete large offer(s) that is made for the entire complex product. In some cases, the EPC may create multiple offers related to big sub-products of the complex product. For example, the EPC may address the construction and the maintenance support of a complex product in 2 separate offers. On the other hand, the EPC can receive and then manage offers made by other stakeholders, although these offers can then only be viewed, approved, or rejected by it.

Finally the customer (with the advice of the EPC/project developer) and consultants (authorized by the customer) represent the ultimate consuming side of the offers that can only manage offers made by other stakeholders to them, in most cases by an EPC, although these offers can only be viewed, approved, or rejected.

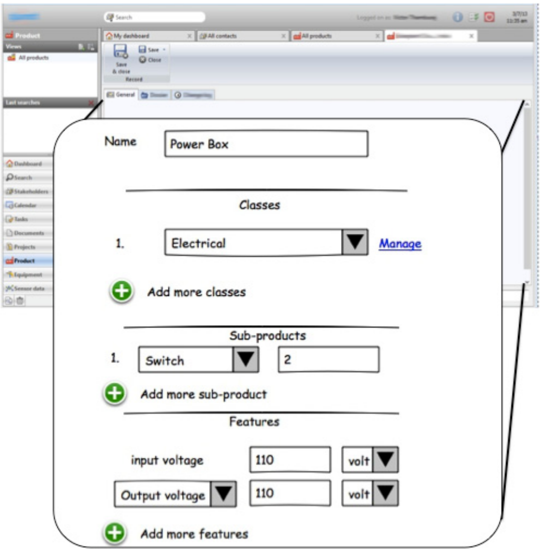
Please note that in principle, within a collaboration community/alliance established among the complex products' stakeholders, there are some stakeholders who act as **providers** (e.g. providers of equipment, manual services, software-based services, and composed services, etc.) and the others who act as **consumers** (e.g. including customers, as well as the intermediary organizations such as EPC, project developer, service integrator, etc.). Within this alliance, different kinds of offer may appear. One kind of offer is related to customization of a sub-product (large or small) to match certain specific requirements related to the complex product. Another kind of offer however, is related to the announcement / advertisement of a sub-product within the alliance. While the former kind of offer is made to a specific recipient, who would be the only stakeholder with the authorization to view and approve/reject that offer, the later kind of offer is made to "*public*", and therefore all members of the alliance can view and approve this kind of offer. Table 1 shows the Use Case of Managing offers.

Table 1. Managing Offers Use Case

Use Case Name	Manage Offers
Description	Supplier/Seller/EPC can review and manage its offers by choosing to add, edit, or view the offer details. EPC/Customer/Consultant can review and manage offers made to them by other stakeholders, by choosing to view, approve or reject the offer.
Actor	Supplier/Seller/EPC/Customer/Consultant
Precondition	Supplier/seller EPC successfully logs into the system
Normal Flow	*1.* System displays the offers that the Supplier/Seller/EPC has made 2. System displays the offers that other stakeholders have made to the EPC/Customer/Consultant or to the public. 3. The stakeholder selects an offer to view, edit, approve, reject or selects to add a new offer depending on its role 4. System redirects the user to the relevant use case.
Alternative Flows	

8 Implementation and User Interface

While the pilot implementation of the complex product specification system is the next step of this work, which will be implemented on top of the GloNet cloud platform [6], at this stage we have designed an abstract view for presenting the system functionality (e.g. data manipulation services) to the users, namely designed the main elements of the user interface.


Name: Power Box

Classes

1. Electrical

Sub-products

1. Switch 2

Features

input voltage 110 volt

Output voltage 110 volt

Fig. 3. Adding New Product window

In other words, clearly the user interface that will be implemented for the complex product specification system will be user-friendly and easy to understand and use by the stakeholders, while hiding the complexity of this system. But at the moment, the aim of the current interface design is to give a first feeling for how the planned functionality can be used through this interface. Figure 3 shows the position of this interface within the GloNet platform's user interface, and provides an abstract view of the window for adding a new product.

9 Conclusion

In order to support the customizable configuration of our targeted complex products, we have designed a complex product specification system. This system introduces generic constructs and functionality needed by different stakeholders, and assists them with their incremental and iterative complex product specification tasks. It further aims to properly authorize the stakeholders, depending on their roles and responsibilities, in relation to the product specification.

Our proposed complex product specification system is distinguished from other similar systems [7], due one or more of the following features:

1. Providing an information system that simultaneously supports the needed access and usage by two distinct types of stakeholders involved in our targeted service-enhanced complex products:
   a. those stakeholders that provide/supply sub-products (e.g. providing equipment and services catalogues), e.g. manufacturers introducing a new sub-product related to the complex products

   b. those stakeholders that participate in specification of the complex product itself

2. Supporting the collaboration and joint specification of one complex product by multiple involved stakeholders that can also be geographically dispersed
3. Providing a customized role-based view on the complex product for each of the involved stakeholders
4. Profiling both the users and the software systems that access the information system, thus keeping the history of actions related to the complex product

It is important to mention that item 1 mentioned above is the main distinguishing feature of our proposed approach against all other commercial configuration systems, while this feature is a necessity in support of the virtual organizations involved in our targeted class of service enhanced complex products.

Furthermore, the proposed complex product specification system provides the base for the next step of this research, focused on semi-automating the process of sub-products cataloguing, and building an intelligent recommender, in dialogue with the user, for the complex product configuration.

Acknowledgment. This work is funded in part by the European Commission through the GloNet project.

References

1. Aldanondo, M., Veron, M., Fargier, H.: Configuration in manufacturing industry requirements, problems and definitions. In: Systems, Man & Cybernetics (IEEE SMC 1999), Tokyo, Japan, vol. 6 (1999)
2. Mittal, S., Frayman, F.: Towards a generic model of configuration tasks. In: IJCAI, vol. 2 (1989)
3. Afsarmanesh, H., Thamburaj, V.: ICT requirements analysis for enterprise networks supporting solar power plants. In: Camarinha-Matos, L.M., Xu, L., Afsarmanesh, H. (eds.) PRO-VE 2012. IFIP AICT, vol. 380, pp. 149–157. Springer, Heidelberg (2012)
4. Camarinha-Matos, L.M., Afsarmanesh, H., Koelmel, B.: Collaborative Networks in Support of Service-Enhanced Products. In: Camarinha-Matos, L.M., Pereira-Klen, A., Afsarmanesh, H. (eds.) PRO-VE 2011. IFIP AICT, vol. 362, pp. 95–104. Springer, Heidelberg (2011)
5. Afsarmanesh, H.: Required Information/Knowledge Provision Services Specification. EU Project (285273) - Deliverable 2.1, Global enterprise network focusing on customer-centric collaboration (2012)
6. Bauer, M.: GloNet Platform Design Specification. EU Project (285273) - Deliverable 3.1, Global enterprise network focusing on customer-centric collaboration (2012)
7. Blecker, T., Abdelkafi, N., Kreuter, G.: Product Configuration Systems: State-of-the-art, Conceptualization, and Extensions. In: The Eight Maghrebian Conference on Software Engineering and Artificial Intelligence, Sousse, Tunisia (May 2004)