In: Proceedings of the International Conference on Economic, Technical and Organisational aspects of Product Configuration Systems (PETO 2004), June 28-29, 2004, Copenhagen, Denmark, pp. 97-107.

XML-BASED DATA EXCHANGE OF PRODUCT MODEL DATA IN E-PROCUREMENT AND E-SALES: THE CASE OF BMECAT 2.0

Volker Schmitz, Joerg Leukel

Department of Procurement, Logistics and Information Management
University of Duisburg-Essen (Campus Essen)
Universitaetsstrasse 9
45359 Essen, Germany
Email: {volker.schmitz, joerg.leukel }@uni-essen.de

Oliver Kelkar

Competence Center Electronic Business
Fraunhofer Institute for Industrial Engineering
Nobelstrasse 12
70569 Stuttgart, Germany
Email: oliver.kelkar@iao.fhg.de

While the exchange of product model data is standardized by the STEP standard, web-based e-procurement and e-sales show a different picture. The rise of XML has led to a variety of different, often from-the-scratch catalog standards. Heterogeneity is also reflected in diverse models for product data. This is especially true for complex product models as necessary for configuration. However, the capabilities of XML-based standards are limited regarding configuration requirements. Concerning this situation, we will present results of the BMEcat standardization. BMEcat is the leading standard for catalog data exchange in Europe. Whereas its present version was the result of a company-focused standardization process, the enhancements of BMEcat 2.0 are also based on comparative analysis and research activities. We will describe the process of adopting and integrating existing concepts from literature, and the introduction of new concepts regarding price definition based on formulas and repositories for industry-wide price parameters.

Significance: The paper presents concepts for product configuration and respective data models that will be introduced in BMEcat 2.0, the forthcoming version of the leading XML standard for catalog data exchange. The new capabilities contribute to the next phase of e-procurement that will also cover more complex products and dynamic pricing.

Keywords: E-Procurement, Electronic Data Interchange, Standardization, XML

(*Received* ; *Accepted*)

1. INTRODUCTION

While the exchange of product model data is, in general, standardized by the STEP standard, web-based e-procurement and e-sales show a different picture. The rise of XML as the common format for exchanging electronic product catalogs has led to a variety of different, often from-the-scratch exchange standards. For instance, cXML and xCBL are driven by major software companies, EAN.UCC and OAGIS are being developed by industry consortia, thus non.-formal standardization bodies. Numerous vertical initiatives like PIDX complement these standards. The heterogeneity caused by a number of different exchange formats also reflect in diverse capabilities for representing product data. This is especially true for complex product models as necessary for product configuration systems. Recent studies have shown that the capabilities of XML-based catalog standards are limited regarding configuration requirements [1]. Therefore, the current state of relevant standards is not sufficient to support the increasing requirements of buy-side procurement, electronic marketplaces, and supplier participation.

Taking the described situation in mind, this paper will present results of the BMEcat standardization initiative to overcome deficits in XML-based catalog data exchange. The non-proprietary BMEcat format is the leading standard for catalog data exchange in Europe [2]. While its present version 1.2 was the result of a company-focused standardization process, the enhancements of BMEcat 2.0 are also based on comparative analysis and research activities. In the respective field of configuration, we will describe the process of adopting and integrating existing concepts from literature (e.g., bills-of-material, constraints). In addition, we will introduce new concepts regarding price calculation based on formulas and repositories for industry-wide price parameters.

Widening the scope of web-based procurement and sales systems beyond MRO goods (maintenance, repair and operations) will help to step into the next phase of business-to business e-procurement. This will enable covering complex goods and targeting new product domains and branches of industry as well, thus the business value of e-procurement systems will be enhanced.

2. PAPER ORGANIZATION AND RELATED WORK

This paper aims at introducing basic concepts for product configuration and preliminary XML data models as they will be implemented in BMEcat 2.0. It supplements previous work on the capabilities of relevant XML-based catalog standards by providing a practical solution to important problems as they are seen from the view of web-based e-procurement and e-sales systems. Our paper is structured as follows:

First we will describe three complexity levels; these are fix, parameterizable and configurable products (Section 3). Then, we will look at current practice in handling product data in e-procurement and e-sales. On one hand, we have to consider sell-side systems that are loosely coupled with buy-side systems (Section 4.1). In this scenario, the exchange of product data is limited to the minimum; hence configuration processes take place on the sell-side only. On the other hand, if companies exchange product data for e-catalog applications, we have to consider relevant XML-based exchange formats. Hence we will refer to a comparative analysis that we have done within a European standardization project [3] (Section 4.2). Both, the levels of complexity and the identified needs for improved standards will serve as a foundation for introducing basic concepts to be implemented in BMEcat 2.0 (Section 5). Due to its preliminary status and the overall complexity of the BMEcat specification, we will outline basic concepts only. Finally, we will summarize our work and point out some implementation aspects (Section 6).

If we limit literature to research that deals with product modeling issues in interorganizational information systems expressly, we can form three main working areas. The first area deals with new modeling and configuration concepts that take interorganizational requirements into account (e.g., [4]). This work mainly aims at improving knowledge-based algorithms for configuration processes as the core of sell-side application systems (e.g., [5]). Recently, the semantic web and ontologies gave a new impulse (e.g., [6]). This topic is not in our focus, since we do not aim at developing new methodologies, but adopt classic configuration principles like bills-of-material and search for XML-based representations. Another area is built of work in the context of mass customization as a strategy that integrates construction, production and distribution management [7]. Research work on syntactical and semantic aspects of B2B product data exchange forms the third area. It is characterized by domain-specific issues, i.e. exchange protocols for catalog data and reference models for products, price and classification information (e.g., [8]). We address this area mainly.

In addition, the approach in our paper distinguishes from most research work in product data management, because it focuses on the distribution phase of the product life-cycle, not on the early phases (e.g., planning, engineering).

3. PRODUCT MODELS

Product models can be differentiated by their complexity. According to [9], we distinguish fix, parameterizable and configurable products.

3.1 Fix Products

The first complexity level is limited to the description of fix products which do not need to be configured. In a simple case the description of such a product is realized by a continuous text which contains all relevant information. In practice these descriptions are often used to transfer a lot of information in a proprietary structure. The problem is that these texts are not only specifying different product characteristics, but are also important for order processes; for instance, if they contain information about special prices, product availability or minimum order quantities. These specifications can not be interpreted by a catalog application.

The following example shows how product characteristics are often described: "10-60 Nm; 12.5=1/2; 392mm L; acc. DIN ISO 6789 (4.3.2 < 1 sec.); ± 4% Tolerance; right/left; Plastic Knob; Safe-Boxes avail." This description is understandable from an expert's point of view but it can not be assumed that this knowledge is equally distributed among all buying employees. In addition, it is difficult to compare the product with another one from the same product group, because texts have to be compared which might be structured in different ways.

Introducing a set of product attributes describing these products (here: i.e. max. torque, square wrench size, length, quality standard, tolerance) is not the solution to overcome heterogeneity in product description. To ensure a comparable product specification among all products of one product group, standardized sets of attributes have to be followed. These sets of attributes are provided by standardization organizations as part of or in addition to a standardized classification system (e.g., eCl@ss, EGAS, eOTD). A set of attributes defines for one product class which product attributes must be used to describe a product belonging to this class.

3.2 Parameterizable Products

The next level of complexity arises from the fact that single attributes are not sufficient for describing product variants. Product variants are a set of products which can be distinguished by a few attribute values especially when these values are selected from a predefined list. If product variants are represented by fix products, all possible combinations of attributes and values must be defined as individual products. In fact of the non-linear increasing number of possible combinations, a small amount of attributes already leads to a considerable number of products with an almost identical and therefore redundant description. Furthermore, the connection between the variants is lost for the user of a catalog application. The solution for this problem is to define in addition to static attributes so called variable attributes and allowed values for these attributes.

In order to identify a product, e.g. for the order process, it is necessary to select the base product and to fill all variable attributes with values. Eventually an order number is built by combining the product identification number of the base product with the coded values of all variant attributes. If product variants are described only by a selection of attribute values the order number can be built by concatenating the base product number with the attribute value codes.

The generation of a valid product order number is more difficult. In these cases the following principle is used. The selection or input of attribute values is determining further attributes – in addition to the derived order number. Thus a dependency between the specification of non-fix attributes and other elements of the product description exists. These dependencies have to be considered. But this is sometimes difficult or not possible at all. For example, if a product is identified by an EAN (European Article Number) then it is difficult to map a single product specification including non-fix attributes to a set of EAN, because each EAN is assigned in an independent manner (of course within the supplier's EAN domain), and thus follow no formula. Further examples for variant-dependent product data are figures, description texts, delivery time, availability, and especially the product price.

3.3 Configurable Products

So far we have discussed products only that were specified by discrete attributes; though in practice product configuration is not characterized by a close relation between a product feature and a single attribute but by the necessity to select from one or more components (device, assembly). These components are products in their own that can be described by the same set of data structures (price and order information, static and variable attributes, configuration). However, it is necessary to determine whether the component behaves like an ordinary product that can be ordered independently from a configuration process. The role that a product plays in configuration processes is described by semantic relationships between products (very similar to the bill-of-material concept).

Interdependencies exist between selected components or even between attributes and components similar to relationships between attributes (values of attributes respectively). These dependencies can be very complex; they require a flexible rule-based modeling concept (constraints). For example, selecting the rechargeable battery (component) for a cell phone (base product) determines the speech/stand-by time as well as the weight of the device (attributes).

Assigning values to variable attributes and selecting components in nearly all cases has an effect on the product price. Besides a completely independent price specification (defined price for each variant or configuration), often a flexible system of allowances and charges to the basic price is used. This information extends the bill-of-material.

4. PRODUCT CONFIGURATION IN E-PROCUREMENT AND E-SALES

Concerning e-procurement and e-sales, we have to consider current practice in product data exchange between suppliers and buyers. One alternative to buy-side systems containing all supplier product data is integrating both systems by transferring synchronous messages. Hence, the product model and configuration problem is solved by relying on proprietary, sell-side implementations.

4.1 Sell-Side Configuration

When coupling sell-side and buy-side systems, only a small part of all the product data is transferred via a product catalogs from the supplier to the buyer. This data forms the basis so that the products are findable by search and navigation mechanisms, and posses a meaningful description consisting of product name, basic attributes, keywords and so on. Additionally, the respective product, group or class has an URL (uniform resource locator) pointing to the sell-side system of the supplier. If the buyer selects such a product, group or class in his e-procurement system, a remote catalog access is started: The procurement process is carried forward to the remote sell-side system. In this system, the buyer can select products, specify variants or configurable products; in other words he fills a shopping cart in the remote systems. When he ends the remote session the shopping cart containing the necessary order information (product identification, unit, price, quantity, delivery time, and additional specification) is returned to the e-procurement system. There, the content of this shopping cart is merged with an existent or converted into a new shopping cart. The buy-side procurement process can be continued.

One advantage of the remote catalog access concept is that even complex configurations on the basis of expert systems and with direct integration with the supplier's ERP system can be realized, without the need of transferring all the product information within the catalog. A powerful product model is not needed. By this approach, a catalog-creating company can bypass the creation and update of extensive catalog data and prevent that valuable product knowledge is transferred to customers, or even competitors. Additionally, the connection to ERP systems enables the calculation of real-time availability and price information.

The application area of the remote catalog access concept is not limited to complex products and connecting sell-side systems. Especially large or constantly changing assortments of standardized goods are suitable. For example, it is not reasonable that a purchasing company builds up and maintains a catalog for books and magazines. In this case it would be advantageous to establish a remote catalog access to the sell-side system of a service provider who is a specialist for the whole assortment of books and magazines. Another scenario is to establish a remote access to marketplaces which offer a high number of supplier catalogs.

There are some disadvantages and limitations in using the remote catalog access. When switching to an external application the user acts in a new environment which differs from the original catalog application, in both handling and functionality. The integration of the remote sell-side application (or marketplace system) and the in-house purchasing organization is difficult and sumptuous. On one hand, established workflows, authorization and budget constraints are bypassed, on the other hand it can not be guaranteed that the product prices coming from the remote system are compliant with bilateral agreements between buyer and seller. Additionally, there is a danger that buyers will order products which are not approved because the buying company has no control over the assortment of goods in the remote system.

4.2 Capabilities of XML-based Standards

The standards for electronic product catalogs in e-procurement scenarios do not match the requirements on the representation of parameterizable or configurable products. Many catalog standards like RosettaNet, CIDX or EAN.UCC do not provide any support for complex products, but are designed for describing fix products only. Other standards like BMEcat 1.2, catXML, cXML, eCX, OAGIS or xCBL provide some concepts towards configuration, however, relevant requirements are not covered; thus parameterizable or configurable products can not be handled in a standardized way [3].

Table 1 gives an overview of the modeling concepts that are already covered by catalog standards (for details see [1]).

	BMEcat 1.2	**cXML 1.2.008**	**OAGIS 8.0**	**xCBL 4.0**
Variable Attributes				
Selection of discrete values (Variants)	yes	no	no	no
Cardinality	no	no	no	no
Default Values	no	no	no	no
Input Parameters	no	no	no	no
Relationships: derived Attributes	no	no	no	no
Relationships: derived Parameters	no	no	no	no
Relationships: Product Price	no	no	no	no
Relationships: Order Number	yes	no	no	no
Relationships: Restrictions	no	no	no	no
Selection of Components				
Optional Choice (Product References)	yes	no	yes	yes
Selection Types /Number of Types	yes/9	no	yes/1	yes/4
Selection: Mandatory	yes	no	no	yes
Cardinality	no	no	no	no
Default Values	no	no	no	no
Relationships: derived Attributes	no	no	no	no
Relationships: derived Parameters	no	no	no	no
Relationships: Product Price	no	no	no	no
Relationships: Order Number	no	no	no	no
Relationships: Restrictions	no	no	no	no
Remote Catalog Access	no	yes	no	no

Table 1. Comparison of Product Models in XML Catalog Standards

The empirical study shows that all standards allow a product description based on free-defined attributes. BMEcat is the only standard that supports the description of variants by defining how a variable attribute must be assigned with a value from a domain of discrete values. But the capabilities in BMEcat 1.2 are very limited, primarily because of missing constructs to declare dependencies between attribute values and prices. Furthermore it is not possible to limit the range of permitted variants or to exclude specific variants. Thus its practical usability is questionable. None of the standards contains data structures that represent (price) dependencies, constraints and rules between a basic product and its components; though these concepts are a prerequisite for modeling configurable products.

5. BMECAT 2.0

This Section presents a preliminary version of the BMEcat 2.0 product model. The product model was developed by a working group, consisting of all partners relevant to B2B e-commerce, such as buying companies, suppliers from industry, intermediaries (marketplaces and catalog service providers) as well as software companies specialized in e-procurement applications. All companies are located in Germany (e.g., Deutsche Telekom, cc-chemplorer, Heiler Software). This group compiled, evaluated and committed to a set of requirements. Supplemented by research studies and comparative analysis of competitive standards, the group's technology experts began to specify new and enhanced XML data structures by using XML Schema, the standard schema language for XML exchange formats.

All data models will be visualized by proprietary graphical representations, which are part of XMLSpy, a software tool for XML data modeling. The following table shows the main symbols and gives a short explanation.

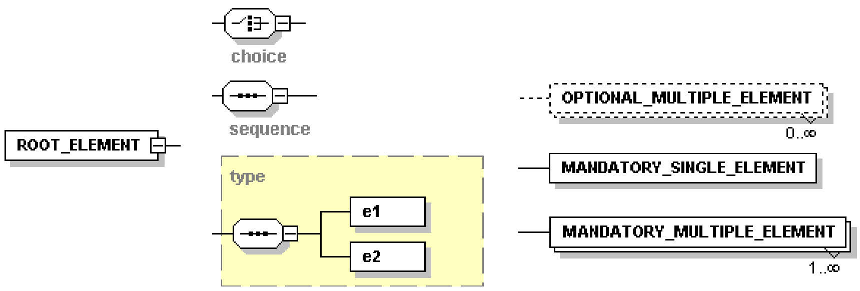

Table 2. Symbols used in XML Schema Diagrams

5.1 Basic Concepts

When modeling configuration, we must adhere to some concepts of product representation in electronic catalogs. These concepts are not specific to configuration, but can be reused and enhanced for configuration purposes. We will describe three basic concepts. These are standardized product classification, price calculation based on formulas, and uniform integration of all products into one catalog.

5.1.1 Standardized Product Classification

Contrary to sell-side systems or proprietary, often CD-ROM based product catalogs (and configurators), a standardized product classification is essential to e-procurement and marketplace systems. Standardized means that all product classes and associated sets of attributes are subject of a horizontal or vertical standard for product classification; hence supplier-specific class hierarchies and attributes are not sufficient. The reason is that catalogs of different suppliers have to be integrated into a multi-supplier catalog. This enables efficient product search, especially attribute-based search, and qualified product evaluation by comparing product properties expressed by attributes and their values. Meeting these requirements, buying companies can establish a common understanding of a product domain, its terminology and semantics.

Modeling electronic product catalogs calls for giving references to product classification systems, product classes and respective attributes whenever a product is categorized and described by attributes. This requirement is fulfilled by the element CLASSIFICATION_REFERENCE as part of product master data. In addition, it ensures that a product can be belong to more than one classification system, but may belong to one class of each system only. Due to the BMEcat 1.2 terminology, attributes are called features.

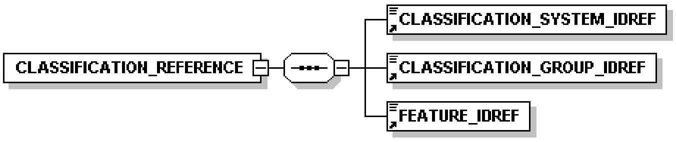

Figure 1. Data Model for Classifying Products.

5.1.2 Price Formulas

From the e-procurement point of view, product catalogs serve as an instrument for getting product information, choosing the right products and starting order processes. A prerequisite for such a decision and process is to get valid price information for each required product. Since prices of configurable products are dependent on selected components and attribute values, the underlying price model must be able to represent the relationships between configuration, attributes and the price. The BMEcat 1.2 price model is quite capable (see [5]), but not sufficient for configuration purposes. Therefore, we extended the price model and introduced the generic concepts of price formulas for representing price information and the described relationships. However, this model is not only used for configuration. It is also suitable for fix products, when the price is dependent on external price parameters. Prime examples are products, which consist of metals or include a relevant amount of metal. Since the metal stock market is quite volatile, the product prices are also dependent on the current market price.

The model for price formulas is subdivided in the definition of formulas (catalog header) and the application of formulas on the product level. Therefore, formulas need to be defined only once, and can be reused for different products of the same catalog.

The element FORMULAS as part of the catalog header builds a repository of formulas (see figure 2). FORMULA_ID and FORMULA_VERSION indentify each formula. The formulas are described by the elements FORMULA_NAME and FORMULA_DESCR; the element MIME serves for multimedia objects (files) relevant to formulas (e.g., drawings, specifications, standards). FORMULA_SOURCE gives a reference to the organization or company that defines and maintains the formula. This enables to build industry-wide repositories. In course of the BMEcat 2.0 development process, a couple of industry associations have already proposed their intention to standardize compliant formulas for their domains. These will be mandatory for all respective suppliers.

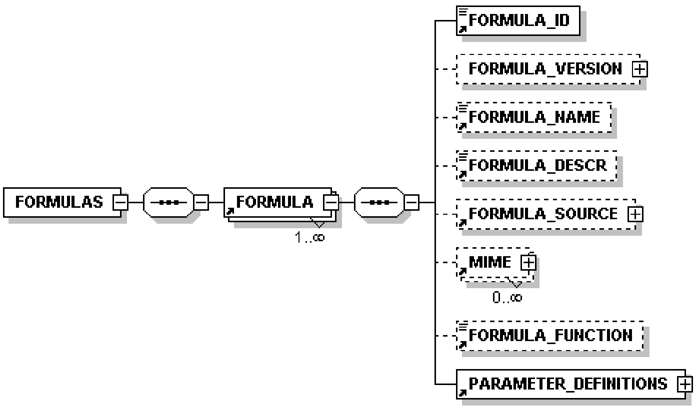

Figure 2. Data Model for Price Formulas.

The actual definition is subject of the element FORMULA_FUNCTION and a list of parameters that are part of the function (see figure 3). This fulfills two needs. First, it is possible to calculate the product price by using the formula, its parameters and respective parameter values. Second, the parameter can be described is such way, that the price calculation is meaningful, understandable, and traceable. In some branches of industry, defined parameters are already a common good that the purchasing decision depends mainly on the parameter values; calculating the exact price is not necessary.

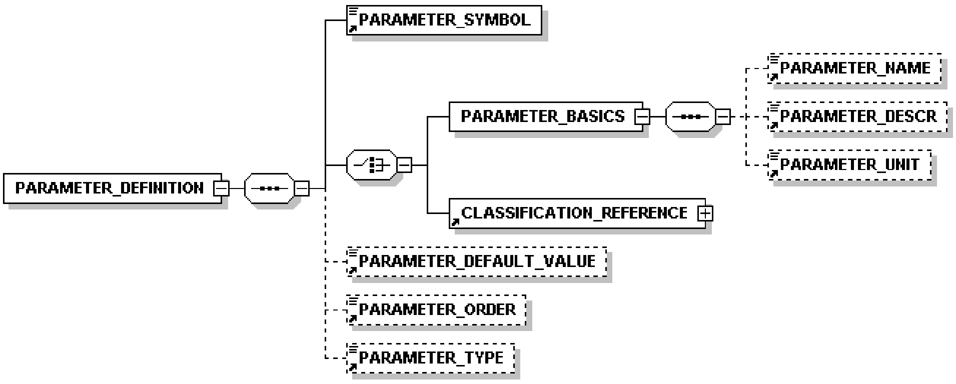

Figure 3. Data Model for Parameter Definition.

The syntax of FORMULA_FUNCTION is equal to functions or expressions in JavaScript, thus it is standardized, not proprietary. Speaking of PARAMETER_DEFINTION, we added CLASSIFICATION_REFERENCE, so that attributes of classification systems can serve as parameters, too (e.g., length, diameter, weight).

The element PARAMETER_DEFAULT_VALUE holds default values for parameters. In many cases, these values will be overwritten on the product level. Since formulas are applied during the run-time of a catalog different data sources can be used to get the actual values. We postulate three categories of parameters (PARAMETER_TYPE): The first one covers parameters which have a given, static value that is already attached during the build-time; these are not real parameters. The second category consists of parameters, whose values can be accessed from a web resource (e.g., URI, web service of a metal stock market). The third category groups parameters that get their values only by user interaction or requesting data from an internal software system.

Price formulas are applied on the product level by giving a reference to their identifiers (FORMULA_IDREF, see Figure 4). In many cases, parameter values will be overwritten or instantiated with product-specific values; this is the role of the PARAMETER element. PARAMETER_SYMBOLREF names the parameter, PARAMETER_VALUES sets the product-specific value. However, default values can be used directly, too.

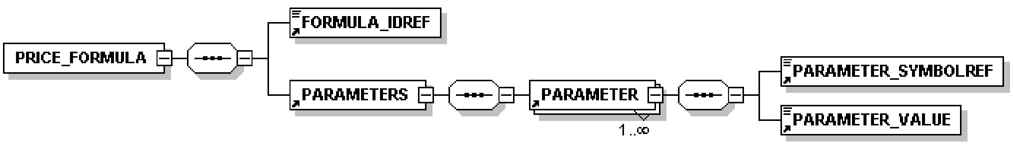

Figure 4. Data Model for Price Formulas on the Product Level.

5.1.3 Uniform Integration of all Products into one Catalog

One of the main goals of the BMEcat initiative is to transfer products of all kinds of complexity in one electronic product catalog. In today's e-procurement systems, due to different catalog exchange formats there are often separated electronic product catalogs for fix products and for products which are handled by remote catalog access. Therefore users of e-procurement systems have to know whether the product they are searching for is configurable or not in order to use the right catalog. Additionally the possibilities on describing the products differ very much between 'fix' and 'remote access' catalogs. Product catalog standards for fix products in most cases offer a wide range of support for the selection process like keywords, attributes, pictures and product classification whereas standards specifying the remote catalog access only provide very limited description possibilities [1].

The approach of BMEcat is to integrate products of all degrees of complexity in one catalog and providing the full set of description options. This allows using the same search and navigation principles like parametric search, keyword search and hierarchical navigation using a class tree. Another advantage is that the catalog creator is able to present his whole assortment of goods in a uniform way. Not until the start of the ordering process there have to be different procedures according the degree of complexity of the products. So it is possible to integrate all products which may vary from fix products, via simple configurations to complex products processed by means of the remote catalog access.

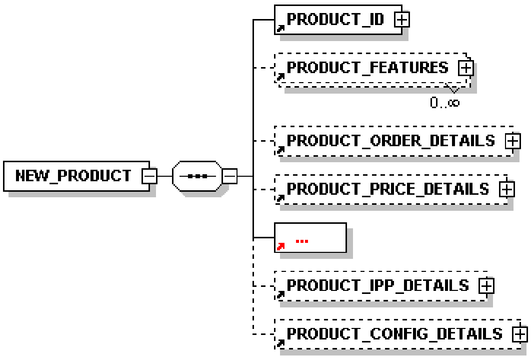

Figure 5. Data Model for Products.

5.2 Configurations

Configurations are modeled in BMEcat by the element PRODUCT_CONFIG_DETAILS. This element is used to represent parameterizable and configurable products. Only the product model, this means the structure of the product is specified here. There are no specifications how the configuration process has to be carried out, or how the GUI representation has to look like. These details are subject of the target software system. It was one of the main goals to build a product model which enables the catalog creators to extend the fix products with simple configurations in an easy way. To establish this, BMEcat forbear from some more complex modeling options like the definition of rule based settings of values. For products which are too complex to represent in BMEcat 2.0, there is still the alternative to use the remote catalog access.

5.2.1 Stepwise Configuration

The whole configuration process is divided up into single steps. With each of this steps one variant product characteristic is being defined. This product characteristic is specified by setting a value for a variant product feature (CF_FEATURE), or by choosing a sub component (CF_PART) from a list of alternatives. Whether more than one product characteristic is determined at one time (on one page) or step-by-step with only on product characteristic each step (one step on each page) depends on the implementation of the target system.

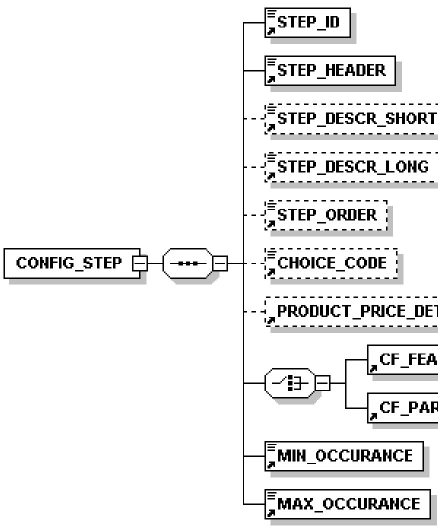

Figure 6. Data Model for Configuration Steps.

STEP_HEADER, _DESCR_SHORT und _DESCR_LONG are used for a textual description of the configuration step. With the element STEP_ORDER the sequence of the configuration steps can be specified.

The element CHOICE_CODE offers the possibility to specify a code which can be used at the end of the whole configuration process to generate an unambiguous order number.

To specify a basic price for one configuration step the element PRODUCT_PRICE_DETAILS is used. This basic price will be entered in the calculation of the total price of the fully configured product. In this context, the above introduced price formulas are used. Therefore it is possible to compute variant prices which are depending on the already completed configuration steps.

Due to the fact that total price and order number can also be computed by the configuration rules both fields are optional elements.

The structure of the product is represented by the elements MIN_ and MAX_OCCURENCE as well as CF_FEATURE and CF_PART. The set of cardinalities which are represented in the OCCURRENCE elements are defining how often a product characteristic can be and has to be specified by the user.

The element CF_FEATURE is used for the definition of product characteristics which can be specified by setting a value of a parameter or by selecting one or more values from a list. By means of the element CF_PART sub components, which can be full products itself, might be chosen from a list. Most of the time it is possible to specify the same product characteristic either as parameter or as sub component. Which of these two alternatives is chosen by the catalog creators depends on how detailed the description of the product characteristic has to be.

5.2.2 Parameterizable Product Attributes

To specify a parameterizable product characteristic either an attribute, which is already defined within a classification system, can be used or a new attribute can be defined by the element FEATURE_DEFINITION. Additionally, a file with a picture or a data sheet can be attached to the catalog using the element MIME to provide a more detailed description of the attribute and its values.

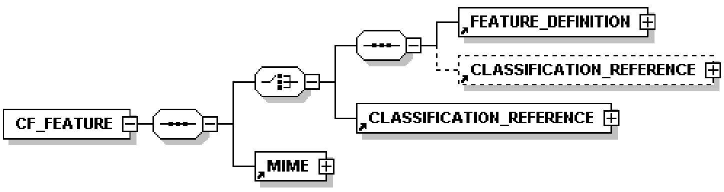

Figure 7. Data Model for Parameterizable Attributes.

The element FEATURE_DEFINITION allows making a detailed definition of all information relevant for the attribute, in particular the domain of the values. This element is used in BMEcat also for the definition of attributes in the context of classification systems and for fix products. The domain of the values can be a basic data type like float or string, but can also be an enumeration. For every value of this enumeration list an additional CHOICE_CODE and individual price information can be provided, which are also included in the final price calculation and generation of the order number.

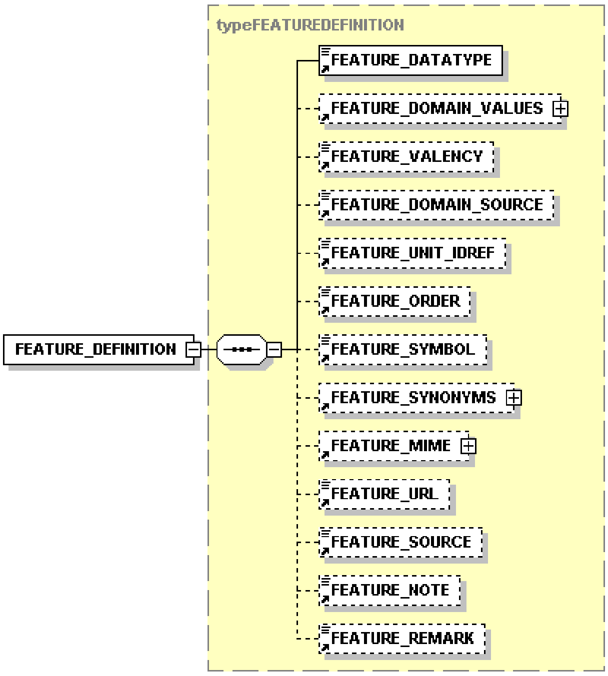

Figure 8. Data Model for Attributes.

5.2.3 Component-based Product Characteristics

Some product characteristics are specified by selecting components from a given list of parts (element PRODUCT_CHOICE). This information is stored in element CF_PART.

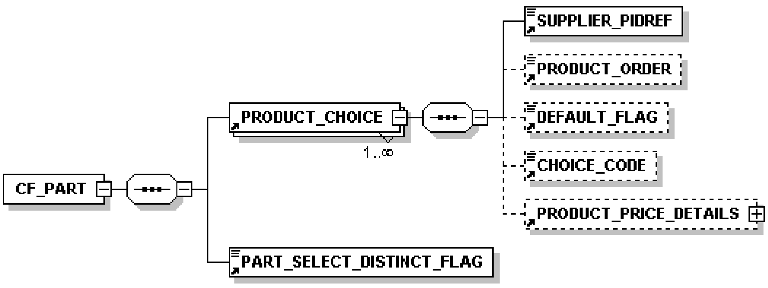

Figure 9. Data Model for Components.

Suppliers can build a list of alternative parts and decide whether different parts must be selected if it is allowed to select more than one part. Each component is referenced by using the supplier-specific product ID (SUPPLIER_PIDREF). The advantage is that such a component can be described by all data structures available for products, i.e. attributes, keywords, describtions, multimedia object etc. However, we have to consider that not all components can be ordered like ordinary products; this is marked by a flag. If a component is described like a complete, single product, this component may be configurable as well (recursion). This allows to build a hierarchy (bill-of-material) that has to be followed when configurating a product. The sequence of parts having the same father component can be controlled, too (PRODUCT_ORDER).

5.2.4 Configuration Rules

Assuring that each customized configuration results in a valid configuration requires a set of rules. These conceptualizations of configuration knowledge can be classified to (a) rule-based, (b) model-based and (c) case-based approaches [10]. Configuration rules fulfill two roles. First, they assure consistency. Second, they lead to modifications of product attribute values.

The BMEcat 2.0 product model can be classified into the model-based approaches, especially constraint-based approaches. Constraints among components restrict the ways components can be combined. The syntax is restricted to the minimum; it allows only the definition of valid and invalid combinations of product characteristics.

Concerning the second role, all attributes are subject of constraints, thus they can be modified. The specification is described in a rule-based manner leading to if-condition-then-modification statements. All product attributes can be referenced on the right side of these expressions (e.g., if component = ”wheel” then total_weight = total_weight + wheel.weight). This covers also order number and product price of the configuration.

The drawbacks of combining rule-based and constraint-based approaches are counter-balanced by advantages for catalog creators, who can easily describe configuration rules (due to rather the simple syntax).

6. CONCLUSIONS

In this paper we have presented a preliminary version of BMEcat 2.0 for complex products in e-procurement and e-sales. First, we distinguished different levels of complexity. Then we evaluated the current state of XML-based exchange format for product catalogs briefly. Eventually, we introduced basic concepts of BMEcat 2.0 and described their application and benefits to widening the scope of e-procurement on the buy-side.

While our approach is driven by industry requirements, we aimed at developing conceptual models that will last for a longer time. One technical obstacle can be the complexity of the exchange format itself, that might be slow down the proliferation. However, we believe that our model is quite balanced. Time will tell if software companies will come up with first implementations in 2004.

6. REFERENCES

1. Leukel, J., Schmitz, V. and Dorloff, F.-D. (2003). B2B E-Procurement Beyond MRO? Proceedings of the 6th International Conference on Electronic Commerce Research (ICECR-6), pp. 493-500.

2. Schmitz, V., Kelkar, O. and Pastoors, T. (2001). Specification BMEcat, Version 1.2. URL: http://www.bmecat.org.

3. CEN/ISSS (2004). WS/eCat Multilingual catalogue strategies for eCommerce and eBusiness. Preliminary Workshop Agreement, Brussels, Belgium.

4. Møller, J., H. Reif and Hulgaard, A. H. (2001). Product Configuration over the Internet. Proceedings of the 6th INFORMS Conference on Information Systems and Technology (CIST 2001).

5. Hadjhamou, K., Caillaud, E. and Lamothe, J. et al. (2001). Knowledge for product configuration. Proceedings of the International Conference on Engineering Design (ICED’2001), pp 131-138.

6. Fu, X., Li, S. and Chen, G. (2002). Semantic Web Based Distributed Configuration. Proceedings of the International Workshop on Grid and Cooperative Computing (GCC2002), pp. 389-401.

7. Rautenstrauch, C., Seelmann-Eggebert, R. and Turowski, K. (Editors)(2002). Moving into Mass Customization - Information Systems and Management Principles. Springer, Berlin et al., Germany.

8. Kelkar, O., Leukel, J. and Schmitz, V. (2002). Price Modeling in Standards for Electronic Product Catalogs Based on XML. Proceedings of the 11th International World Wide Web Conference (WWW2002), pp. 366-375.

9. Stahl, A., Bergmann, R. and Schmitt, S. (2000). A Customization Approach for Structured Products in Electronic Shops. Proceedings of the 13th Bled Electronic Commerce Conference, pp. 252-264.

10. Sabin, D., and Weigel, R. (1998). Product Configuration Frameworks – A Survey. IEEE intelligent systems, 13, pp. 42-49.