

B2B E-Procurement Beyond MRO?

Joerg Leukel, Volker Schmitz, Frank-Dieter Dorloff
Institute for Computer Science and Business Information Systems (ICB)
University of Duisburg-Essen at Essen
45117 Essen, Germany
{joerg.leukel | volker.schmitz | dorloff}@uni-essen.de

Abstract

Electronic product catalogs are the prime instrument in the information phase of procurement transactions. Likewise it is a vital interest of suppliers, thus the companies that create electronic catalogs, to be able to describe products in full detail, according to requirements of customers and in a manner that supports and influences procurement decisions. However, the main object of catalog-based transaction systems are standardized products of a limited specification and complexity. A common term is MRO goods (maintenance, repair and operations). Catalog data is exchanged between companies with the help of XML-based catalog standards. Each standard contains a more or less powerful product model that represents products through data structures and elements. Which types of products can be described depends on the range, complexity and restrictions of the respective product model. This paper aims at analyzing how these XML-based catalog standards model complex products and therefore answers the question if B2B e-procurement beyond MRO is possible.

1. Introduction

Electronic product catalogs are the prime instrument in the information phase of procurement transactions. Buying companies make high demand on the semantics and syntax of catalog data, in order to support buyers and other employees by their business functions and needs. Likewise it is a vital interest of suppliers, thus the companies that create electronic catalogs, to be able to describe their products in full detail, according to requirements of customers and in a manner that supports and influences procurement decisions [3]. The product description plays a major role in sales and procurement systems.

The main object of catalog-based transaction systems are standardized products of a limited specification and complexity. Among these products are primarily indirect goods that are not an immediate input factor for production

processes and can not be attributed to manufactured final goods. A common term is MRO goods (maintenance, repair and operations). Indirect goods are characterized by a limited specification, low single values and high order frequencies as well as at the same time a low share in the procurement budget. The described restrictions cause a limited area of application for e-procurement systems so far. By extending the capabilities of catalog applications concerning product complexity, product models and product data exchange, e-procurement system could re-shape their role as tools for buying direct, complex or strategic goods as well.

Catalog data is the foundation for product descriptions. In contrast to B2C, catalog usage in B2B is characterized by the fact that data of the catalog-creating enterprise is imported into an information system (target system) of the catalog-receiving enterprise. In addition, catalog data exchange is not limited to the relationship supplier-buyer. In many branches of industry catalog data is exchanged along the entire supply-chain, e.g. manufacturer – whole-sale – industry. On the other hand sell-side systems, typical e-shops that represent the products of only one supplier, lose their former importance [7]. Here we look at catalog data exchange as the transfer of catalog data into target systems. Meanwhile XML-based catalog formats and standards became generally accepted. Each of these standards contains a more or less powerful product model that represents products through data structures and elements. Which types of products can be described depends on the range, complexity and restrictions of the respective product model. The models vary from simple text-based descriptions to larger data models coming from enterprise resource planning (ERP) and product data management (PDM) systems.

2. Paper Organization and Related Work

This paper aims at analyzing how XML-based catalog standards model complex products. The empirical analysis can help answering the question to what extent B2B data exchange standards fulfill requirements from B2B e-procurement. To do so, our paper is structured as follows: First we will describe the attribute-oriented modeling of less complex products (Section 3.1). Then we will look at more complex products by distinguishing parameterizable from configurable products (Sections 3.2 & 3.3). Eventually we will draw the attention to a specific B2B e-procurement concept that deals with high complex products and the need for a close integration of e-procurement systems with sell-side systems. This concept is called PunchOut and was introduced by one of the relevant XML catalog standards (Section 4). The different degrees of complexity will serve as the foundation for our analysis of the four most relevant industrial

XML catalog standards. These standards will be examined, which concepts they implement and to what extent they are able to fulfill the outlined requirements (Section 4). Finally, we will evaluate the current state of these standards and formulate some future trends and concerns.

If we limit relevant literature to research and development that expressly addresses product modeling issues in interorganizational information systems, then we see three main working areas. The first area deals with new modeling and configuration concepts that take interorganizational requirements into account (e.g. [12]). This work mainly tries to improve knowledge-based algorithms for configuration processes as the core of sell-side application systems (e.g. [8]). Recently the driving semantic web and ontologies gave a new impulse [6] [11]. Another important topic is personalization (e.g. [1]). The second area is built of work in the context of mass customization as a strategy that integrates construction, production and distribution management [14]. Mass customization calls for user-friendly order and configuration platforms often based on electronic product catalogs [18]. Research work on syntactical and semantic aspects of B2B product data exchange forms the third area. It is characterized by domain-specific issues, for example exchange protocols for catalog data (e.g. [10]), reference data models for products, price and classification information (e.g. [9]), and conversion of XML-based business data (e.g. [17]).

The main contribution of this paper comes from an in-depth analysis of product models in XML-based standards for product data exchange. This analysis is based on a differentiation of product complexity levels from an e-procurement point of view. It structures products into three categories in accordance with [16].

3. Complexity of 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 a fix product is realized by a continuous text which contains all relevant information. In practice these description texts are often used to transfer a lot of information in a proprietary structure. This is problematic if these texts are not only specifying different product characteristics but are also important for the order process (if they contain information about e.g. special prices, limitation of 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 meaningful from an expert's point of view but it can not be assumed that this knowledge is equally distributed among all buying employees. Furthermore 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. Facilitation can be established by individually accessible attributes. To ensure a comparable product specification among all products of one product group, standardized sets of attributes are introduced. These sets of attributes can be specified by a company on its own or can be provided by a standardization organization within a standardized classification system like eCl@ss, EGAS or ETIM [5]. A set of attributes defines for one product class which product attributes must be provided to describe a product from this class. Beside an unambiguous name and a semantic description often allowed domains of values or units are specified. The usability of such a domain of values goes along with its precision in restricting the values. Looking at fix products all attributes are invariable filled with values from the domain of values.

According to the classification system eCl@ss 4.1 the above mentioned torque wrench must be specified as follows (see Table 1). It must be considered that the set of attributes defined in eCl@ss is not sufficient to include all important product characteristics. For instance the lower limit of the adjustment range (here: 10 Nm) and the accuracy (here: 4%) are missing.

Attribute	Value	Unit
Product class	torque wrench	-
Product class number	21-04-02-22	-
Torque, max.	120	Nm
Quality features, certificate	DIN ISO 6789	-
Length	392	mm
Square wrench size	12.5	mm

Table 1: Specification of a torque wrench according to eCl@ss (Cutout).

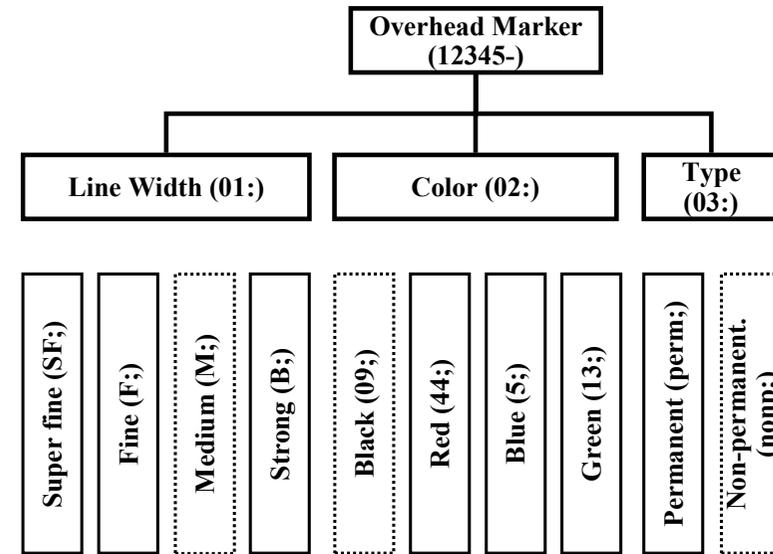
3.2 Parameterizable Products

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 (e.g. a folder, which can be ordered in different colors). If product variants are represented through fix products all possible combinations of attributes and values must be defined as an individual product. Because 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 additionally to static attributes so called variable attributes and assign allowed values to them.

An example for product variants is a set of similar overhead markers which differ in color. In this case the attributes "line width" and "color type" (permanent/non-permanent) are identical whereas the attribute "color" is variable with the values "red", "blue", "green" and "black". In this example only one product has to be specified instead of four ones. This improves the usability of the catalog since the products can be found independently from their color when performing a product search. The size of the search result is reduced and the user is free to choose the desired color.

Variant products might be built with several variable attributes. Looking at the example the overall number of products can be reduced when additionally to the attribute "color" the attributes "line width" with values "super fine", "fine", "medium" and "strong" and the attribute "color type" with values "permanent" and "non-permanent" are specified as variable attributes. The proportion between the use of variable attributes and the use of static attributes which lead to individual products is determined by the catalog creating company. In this way it is possible to highlight different aspects from a marketing point of view and to adapt the catalog to specific industries or customer requests.

Is a product defined by more than one variable attribute the problem might occur that there are some theoretical possible combinations of attributes and values which are not permitted. In this case there has to be a supplemental mechanism, which either excludes illegal combinations or explicitly defines the permitted ones. Looking at the already explained example again it is imaginable that the superfine overhead markers can only be ordered in black.



Product Code: 12345-01:M;02:09;03:nonp;

Figure 1: Example for the creation of variant products

The concept of product variants allows representing certain complex products. But furthermore there are products which can not be described through discrete, predefined values so called parameterizable attributes. Examples are measures like length e.g. for ordering individually tailored cable (e.g. 3 pieces of cable with a length of 4.5 m each) or product-related textual parameters which have to be transferred within the order (e.g. engraving on a pen, text on a business card). Because these values can not be specified in advance but have to be entered at the time of the product selection, the product model has to be extended. The input of the attribute values has to comply with a predefined domain of values and a data format which can vary in detail (field length, pattern, intervals, precision). These specifications must be transferred as product data within the product catalog to enable the target system to generate input forms and ensure that only allowed values are being entered. It is important that the input values are well-defined not only for order processing but especially because the values might be used in formulas, for instance for calculating dynamic prices.

To unambiguously 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. Afterwards an order number is built through combining the product identification number of the base product with the coded values of all variant attributes. If product variants are described only through a selection of attribute values the order number can be built by concatenating the base product number with the attribute value codes.

If parameterizable attributes come to application 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 – in addition to the derived order number – further attributes. The conclusion is that dependencies between the specification of non-fix attributes and other elements of the product description exist. These dependencies have to be considered and if possible modeled in the product data. 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 hard to map a single product specification including non-fix attributes to a set of EAN, because all EANs are assigned in an independent, freely 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. To model these dependencies it is suitable to combine conditional rules with formulas that calculate values of attributes. For instance: <IF Attribute "Color" = Value "red" THEN Attribute "Figure" = "red_pen.jpg">. The formula <Attribute "area" = Attribute "Length" * Attribute "Width"> calculates the values of one attribute by multiplying the values of two other attributes. Rules and formulas are a powerful instrument to assign values to attributes, set default values and define constraints for attribute values.

3.3 Configurable Products

So far we have discussed products only that were specified through 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 acts 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).

A first type of relationship expresses that another product can be (but need not) ordered in addition to the basic product. Those relationships are required for accessories, spare parts and alternative products. For example: In addition to a laptop computer you can order a laptop bag, but you can also order the laptop without a bag.

The situation is different when you have to choose from a list of components. In this case the laptop is ready for order if a device for the empty CD drive cartridge is selected. Self-evident is that some products have to be specified through more than one components; a bill of material describes the structure and the relationship between products and components. Multi-level bills of material are built if a component is from the type configurable product, too.

Similar to relationships between attributes (values of attributes respectively) interdependencies exist between selected components or even between attributes and components. 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 have in nearly all cases an effect on the price of a product. Beside a totally independent price specification (defined price for each variant or configuration) often a flexible system of allowances and charges in addition to the basic price is applied in practice. This information extends the bill of material.

3.4 External Products

In the so called PunchOut model only a part of all the describing product data is transferred via a catalog document from the supplier to the target system. This data forms the basis so that the products are findable through search and navigation mechanisms, and possess a meaningful description consisting of product name, static attributes, keywords and so on. Additionally, the respective product, product group or product class comes with an URL (unique 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, he can start a PunchOut process: 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, 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.

Figure 2 shows the message transfer between a buy-side and a sell-side system as it is implemented in the e-business software of Ariba: Prerequisite is the initial transfer of a catalog containing PunchOut information (message 0: Catalog). A PunchOut session begins with the buyer who requests to setup a remote procurement process (message 1: PO SetupRequest). The supplier answers this request with a confirmation notice (or refusal; message 2: PO SetupResponse). One reason for this setup interaction is to allow the supplier to modify the URL that was sent in his product catalog. Eventually the sell-side system is called via the current URL (message 3: PO Redirect), there the buyer can select or configure products. The PunchOut session ends by a buyer action; the complete shopping cart will be transferred in an XML message to the e-procurement system (message 4: PO OrderMessage).

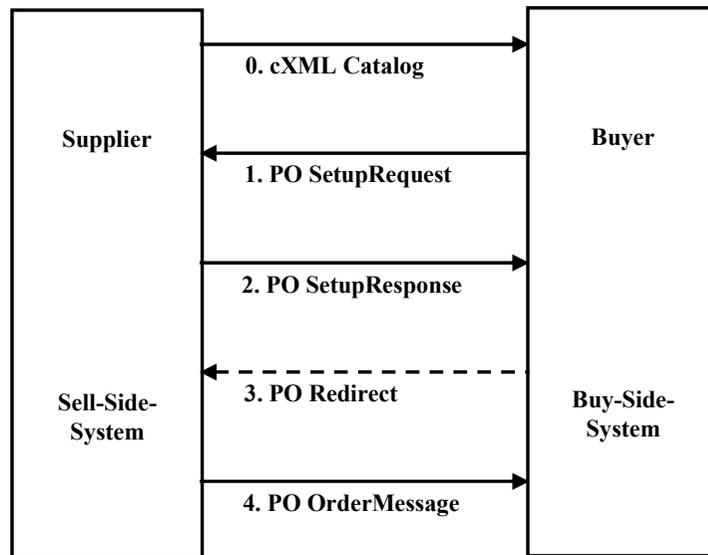


Figure 2: Ariba PunchOut Model

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

The application area of the PunchOut model is not limited to complex products and connecting supplier-side systems. Especially wide 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 PunchOut process with 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 PunchOut process to a marketplace (instead of to a sell-side system) which offers a larger number of supplier catalogs.

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

4. Product Models in XML-based Catalog Standards

On the basis of the developed requirements on product models for electronic catalogs we can now examine selected XML-based catalog standards, ask what requirements they already fulfill and determine, which areas and concepts have the smallest support so far. The selection is limited to the most important, horizontal standards. The selection covers the following four standards:

- BMEcat is a genuine catalog standard [15].

- cXML is the standard data exchange format used by the e-procurement solutions of Ariba. The focus here is on the supply of formats for catalog-based order processes [2].
- OAGIS contains over 200 XML transactions for business documents called business object documents (BODs). It will be integrated into the ebXML framework covering the document level [13].
- xCBL (XML Common Business Library) is an extensive collection of XML business documents developed by CommerceOne [4].

Table 2 gives an overview about the modeling concepts for complex products that are already covered by the catalog standards.

	BMEcat	cXML	OAGIS	xCBL
Static Attributes	yes	yes	yes	yes
Variable Attributes				
Selection of discrete values (Variants)	yes	no	no	no
Cardinality	no	no	no	no
Input Parameters	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	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
External Products (PunchOut)	no	yes	no	no

Table 2: Comparison of Product Models in XML Catalog Standards

Our 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 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.

Selecting components is supported by BMEcat, OAGIS and xCBL in the shape of references to accessory parts. BMEcat offers additional types for specifying references (e.g. necessary additional products, accessories, spare parts). The concept of selection of components serves in these three standards solely for finding products respectively building links to relevant products. 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.

Only the cXML standard offers room to define external products. Its own concept of PunchOut catalogs is a widespread mechanism in practice to integrate sell-side catalog systems into e-procurement applications on the buy-side.

5. Conclusions

This paper shows that despite the need for suitable models for configurable products by suppliers and users of catalog-based procurement systems there are no capable standards for electronic product catalogs which satisfy the specified requirements, thus B2B e-procurement beyond MRO is hardly possible.

A pragmatic and pretty simple to realize approach is the PunchOut concept. As stated before there are some disadvantages which have to be accepted concerning the integration of the external application and the in-house procurement policy and the underlying organizational model. Another restraint is that the PunchOut process is now only supported by cXML and that its messages types are not capable of describing static products in an appropriate way.

To use the advantages of catalog-based procurement for configurable products to a full extent it is necessary to enhance the modeling concepts for fix products with the requirements on describing complex configurable products. This includes in particular the integration of the PunchOut concept in "classic" electronic product catalogs to combine products in one catalog which are

handled within e-procurement system on the one hand together with products covered by PunchOut processes on the other hand.

The paper discusses a taxonomy of requirements for models for configurable products. These requirements are used to analyze four state-of-the-art catalog standards, to examine to what extent they meet these requirements. The results of this paper make some suggestions for the standardization of electronic product catalogs and will contribute especially to the development of the BMEcat standard.

References

- [1] Ardissono, L. and A. Felfernig, A. and G. Friedrich et al., "Personalizing on-line configuration of products and services", in: Proceedings of the 15th European Conference on Artificial Intelligence (ECAI 2002), Lyon, France, 225-229 (2002).
- [2] Ariba, "cXML 1.2.008", URL: <http://www.cxm1.org>, 2002.
- [3] Baron, J. P. and M. J. Shaw and A. D. Bailey, "Web-based E-catalog systems in B2B Procurement", *Communications of the ACM*, vol. 43, no. 5, pp. 93-100 (2000).
- [4] CommerceOne, "xCBL 4.0", URL: <http://www.xcbl.org>, 2003.
- [5] Fensel, D. and Y. Ding and B. Omelayenko et al., "Product Data Integration in B2B E-commerce", *IEEE Intelligent Systems*, vol. 16, no. 4, 54-59 (2001).
- [6] Fu, X and S. Li and G. Chen, "Semantic Web Based Distributed Configuration", in: Proceedings of the International Workshop on Grid and Cooperative Computing (GCC2002), Hainan, China, 389-401 (2002).
- [7] Ginsburg, M. and J. Gebauer and A. Segev, "Multi-Vendor Electronic Catalogs to Support Procurement: Current Practice and Future Directions", in: Proceedings of the 12th International Bled Electronic Commerce Conference, Bled, Slovenia, 331-345 (1999).
- [8] Hadjhamou, K. and E. Caillaud and J. Lamothe et al., "Knowledge for product configuration", in: Proceedings of the International Conference on Engineering Design (ICED'2001), Glasgow, Scotland, 131-138 (2001).
- [9] Kelkar, O. and J. Leukel and V. Schmitz, "Price Modeling in Standards for Electronic Product Catalogs Based on XML", in: Proceedings of the 11th International World Wide Web Conference (WWW2002), Honolulu, HI, USA, 366-375 (2002).
- [10] Leukel, J. and V. Schmitz and F.-D. Dorloff, "Coordination and Exchange of XML Catalog Data in B2B", in: Proceedings of the 5th International Conference on Electronic Commerce Research (ICECR-5), Montreal, Canada, 2002.
- [11] Mohan, K. and B. Ramesh, "Ontology-based Support for Variability Management in Product and Service Families", in: Proceedings of the 36th Hawaii International Conference on System Sciences (HICSS-36), Big Island, HI, USA (2003).
- [12] Møller, J. and H. Reif and A. H. Hulgaard, "Product Configuration over the Internet", in: Proceedings of the 6th INFORMS Conference on Information Systems and Technology (CIST 2001), Miami Beach, FL, USA (2001).
- [13] Open Applications Group, "Open Applications Group Integration Specification", Release 8.0. URL: <http://www.openapplications.org>, 2002.
- [14] Rautenstrauch, C. and R. Seelmann-Eggebert and K. Turowski (eds.), "Moving into Mass Customization - Information Systems and Management Principles", Springer, Berlin Heidelberg New York, 2002.
- [15] Schmitz, V. and O. Kelkar and T. Pastoors, "Specification BMEcat", Version 1.2. URL: <http://www.bmecat.org>, 2001.
- [16] Stahl, A. and R. Bergmann and S. Schmitt, "A Customization Approach for Structured Products in Electronic Shops", in: Proceedings of the 13th Bled Electronic Commerce Conference, Bled, Slovenia, 252-264 (2000).
- [17] Wüstner, E. and T. Hotzel and P. Buxmann, "Converting Business Documents: A Classification of Problems and Solutions using XML/XSLT", in: Proceedings of the 4th IEEE International Workshop on Advanced Issues of E-Commerce and Web-based Information Systems (WECWIS 2002), Newport Beach, CA, USA, 61-68 (2002).
- [18] Ying-Lie, O., "Configuration for mass-customisation and e-business", in: Proceedings of the 15th European Conference on Artificial Intelligence (ECAI 2002), Lyon, France, 11-16 (2002).

Author Information

Joerg Leukel is a research assistant at the University of Duisburg-Essen at Essen (Germany) since 1999. His main working areas are catalog data management, B2B e-commerce and web content management systems. He holds a Master's Degree in business information systems from the University of Duisburg-Essen at Essen.

Volker Schmitz is a research assistant at the University of Duisburg-Essen at Essen (Germany) since 1996. His main working areas are standardization in B2B, electronic product catalogs and web content management systems. He chairs the eBusiness Standardization Committee (eBSC) and is a team member of the CEN/ISSS eCAT Workshop. Mr. Schmitz holds a Master's Degree in computer science from the RWTH Aachen.

Frank-Dieter Dorloff is professor of procurement, business logistics and information management at the University of Duisburg-Essen at Essen (Germany). His main working areas are B2B e-commerce, e-government and supply chain management. He holds both a Master's Degree and a PhD in industrial engineering from the RWTH Aachen.