A Tool for Comparing Configurable Products

Mikko Heiskala[1], **Andreas Anderson**[1], **Virpi Huhtinen**[2], **Juha Tiihonen**[1], and **Asko Martio**[1]

1) Helsinki University of Technology
P.O.B 9600, FI-02015 HUT, Finland
firstname.lastname@hut.fi

2) Kauppalehti
P.O.B 830, 00101 Helsinki, Finland
virpi.huhtinen@kauppalehti.fi

Abstract

Helping customers to select a suitable product is an increasingly important problem in e-commerce where personal advice is not available and many similar products are offered. Comparison of off-the-shelf products is supported in many web-based systems. Sales configuration helps customers in mass-customizing a configurable product for their needs. However, sales configuration alone does not support choosing between configurable products represented by separate configuration models. If the customer feels that the configured product individual is not satisfactory, the configuration process must be repeated for another configurable product. This requires time and effort and can be frustrating. In this paper we describe a web-based tool that helps customers in selecting a suitable product that can subsequently be configured to meet the requirements. Selection support is provided by combining search facilities, table-based comparison, and a configurator to check the consistency of requirements expressed as search criteria. The tool is based on a simple conceptual model that was developed to suit the table-based way of representing comparison data. To facilitate maintenance, the tool semi-automatically generates data for comparison from existing configuration models.

1 Introduction

Configurable products give customers the possibility to obtain product individuals adapted to their requirements. However, selecting a suitable configurable product from the set of available products can be difficult, especially for non-expert customers typical to e-commerce sites. A major contributor to the difficulty of selection is the complexity of such products caused by numerous and potentially incompatible possibilities for adaptation that possibly significantly affect the characteristics of the product individual. Unfortunately, configurators can usually actively support the configuration task only after a configuration model representing the desired product has been chosen. As a result, it may be impossible to reach a configuration satisfying the requirements with a product chosen without proper support. Another product must then be chosen as a basis for the configuration task. This kind of buying process that includes product selection through trial and error causes extra work and can be frustrating (Figure 1(a)). Frustration is elevated if configuration decisions are lost while changing between products or configuration models.

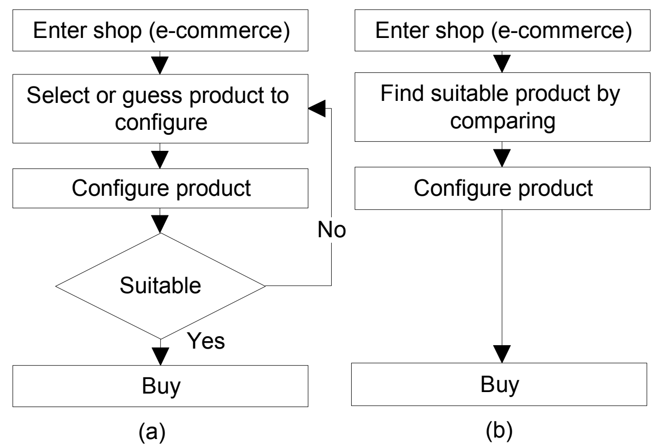

Figure 1. Configuration process without (a) and with a comparison tool (b)

The importance of supporting selection of configurable products and avoiding iteration caused by wrong product selections has been most directly addressed by Pargamin [2002]. There are many models and tools that support product selection of off-the-shelf products, analyzed, e.g. by Stolze [1999] and Ardissono et al. [2002], but support for initial selection of configurable products is largely missing.

The mechanisms for supporting product selection analytically can be divided into filtering, visualization and evaluation mechanisms [Stolze, 1999]. According to Stolze, filtering is provided by hierarchical browsing, product retrieval based on user defined constraints, and even interactive configuration. Selection support during the interactive configuration process has been proposed and implemented e.g. as described by Ardissono et al. [2002] in a form of a personalized adaptive configurator user interface and by Magro and Torasso [2001] in the form of catalog-based component selection supported by a configurator.

A common and practical visualization mechanism is to present products in side-by-side comparison tables. This is relatively widely applied in e-commerce of off-the-shelf products, for instance see [Zones, 2003; Dell, 2003; Garmin, 2003; Porsche, 2003]. Information enabling comparison can be conveyed in electronic product catalogs such as BMEcat

[Schmitz et al., 2001], commerce eXtensible Markup Language cXML [Ariba, 2002], or XML Common Business Library [Commerce One, 2002].

Evaluation mechanisms usually calculate a numerical measure of utility for each product that is consequently used to rank them [Stolze, 1999].

In this paper we describe a practical web-based tool called CCP (Comparison of Configurable Products) that extends analytical product selection to configurable products by applying filtering and visualization mechanisms adapted to this purpose. Our aim is that the tool helps customers to directly select a suitable product to configure, which leads to the buying process without iteration, illustrated in Figure 1(b). The tool is based on a simple conceptual model that enables table-based comparison and interactive search of configurable products, and semi-automatic generation of comparison information from configuration models represented in *PCML*, the Product Configuration Modeling Language of a configurator called WeCoTin described in [Tiihonen et al., 2003].

The rest of the paper is structured as follows: In Section 2 the conceptual model of the CCP tool is described and in Section 3 its implementation is outlined. Next, in Section 4 we discuss and compare our implementation and results with related work. Finally, in Section 5 we present topics for further work and our conclusions in Section 6.

2 Concepts

In this section we describe the conceptual model underlying the CCP tool, summarize PCML and show the conceptual mapping between PCML and the comparison concepts.

2.1 Comparison Concepts

Next we describe the conceptual model underlying the CCP tool. The conceptual model was developed to suit the table-based way of representing comparison data. The main concepts are *product*, *feature*, and *feature value*.

A *comparison model* defines a set of features and a set of products characterized by a set of feature values assigned to the features. A feature characterizes the products in a comparison model on an aspect thought to be interesting for customers. Each product defines a set of feature values for each feature and specifies via a *cardinality* the size of the set of feature value possible for a feature in a valid product individual. A feature value can represent a discrete value or a range in case of integers. Effectively a product is characterized with respect to each feature of a comparison model by a set of feature values and a cardinality. We use the term product in this context in sections 2 and 3.

Products of a comparison model can be divided into *product groups* based on some similarity or segmenting factor, e.g. membership in a marketing product line, basic technology or some major characteristic. For example, cars could be divided to product groups ‘Sedans’ and ‘Convertibles’.

2.2 Mapping of PCML to Comparison Concepts

In this section we briefly describe the main aspects of PCML and our conceptual mapping between PCML and

```
configuration model
Volvo_V40
component type V40
 part GearBox
  allowed types Auto-
   matic, Manual
  cardinality 1
property Motor
  value type string
   constrained by $
   in list('TD','T4')
```

```
configuration model
Volvo_V70
component type V70
 part Transmission
  allowed types Manual,
   Automatic, Manual_AWD,
   Automatic_AWD
  cardinality 1
 part Engine
  allowed types TD,T5,D5
  cardinality 1
```

Figure 2. Example PCML Configuration Models

Configuration Concept(s)	**Comparison Concept**
CONFIGURATION MODEL	*PRODUCT*
PART, PROPERTY	*FEATURE*
PART ALLOWED TYPE(S), POSSIBLE VALUE AND/OR RANGE OF VALUES OF A PROPERTY	*FEATURE VALUE*

Table 1. Mapping Configuration and Comparison Concepts

Volvos	**Volvo V40**	**Volvo V70**
Engine	TD, T4	TD, T5, D5
Transmission	Automatic, Manual	Automatic, Manual, Automatic AWD, Manual AWD

Table 2. Volvos - An Example Comparison Model

comparison concepts. This mapping enables semi-automatic generation of comparison data from PCML configuration models.

The main concepts of PCML are *component types*, their *properties, compositional structure* i.e. the decomposition of a configurable product to its parts, and *constraints*. Component types define the characteristics (such as parts) of their *individuals* that can appear in a configuration. A component type defines its parts through a set of *part definitions*. A part definition specifies a *part name*, a non-empty set of *possible part types* (*allowed types* for brevity), and a *cardinality*. A component type may define properties that characterize the type. A *property definition* includes a *property name*, and a *property value type* that defines the possible values of the property using e.g. a range or an enumeration of values. *Constraints* associated with component types define conditions that a correct configuration must satisfy. A *configuration model* consists of a set of component type definitions. The component type that acts as the root of the compositional structure is the *configuration type*. To support internationalization, a *display name* in the desired languages can be given to component types, parts, and properties and their possible values. Figure 2 shows extracts from two PCML configuration models used to model cars.

A PCML configuration model defines the possibilities for adapting a configurable product to customer requirements with parts and their allowed types, and with properties and their possible values. As a result, it is natural to map a configuration model to a product, properties and parts to features, and part allowed types and possible values of properties to feature values. The cardinality of a part mapped to a feature for a product is mapped as the cardinality of the feature value set. Possible values defined as a range are mapped to a feature value representing it. The conceptual mapping is summarized in Table 1.

An example comparison model, ‘Volvos’, is shown in Table 2. The ‘Volvos’ comparison model is a result of mapping information from the configuration models shown in Figure 2. The configuration model `Volvo_V40` is mapped to the product ‘Volvo V40’. Part `GearBox` is mapped to the feature ‘Transmission’ and the allowed types `Automatic` and `Manual` of `GearBox` as the features values of the product ‘Volvo V40’ for the feature ‘Transmission’. Property `Motor` is in turn mapped to the feature ‘Engine’ and `Motor`’s possible values `‘TD’` and `‘T4’` as the feature values. Property `Motor` in `Volvo_V40` and part `Engine` in `Volvo_V70` are used to model the possible engines the cars can have. They represent the same aspect of similar configurable products, although they happen to have different names. Therefore they are mapped to the same feature, in this case ‘Engine’, in the comparison model.

3 The CCP Tool

Next we give an overview of the CCP tool, its architecture, main components and functionality.

3.1 Overview and Architecture

The CCP tool consists of two parts: *Mapping Tool* and *Comparison Tool*. The high level architecture of the tool is shown in Figure 3.

Mapping Tool is a Java application used to maintain and edit comparison models. The comparison models are stored in a relational database. The tool can import configuration models from WeCoTin Configurators and semi-automatically transform them into comparison model products.

Comparison Tool provides functionality to search and compare configurable products using a web browser. The tool uses Java Servlet technology to dynamically generate HTML pages based on comparison models.

3.2 Mapping Tool

Mapping Tool is used to edit and maintain comparison models, and to keep them consistent with configuration models.

A comparison model is edited mainly by adding, removing, and editing features, products, feature values, cardinalities, and product groups. They can be given a display name in any number of languages to support internationalization. In addition to display names, a description text, and an URL can be specified to give a customer additional information

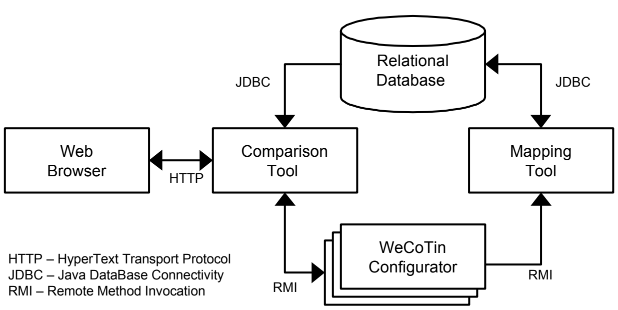

Figure 3. The high level architecture of the CCP tool.

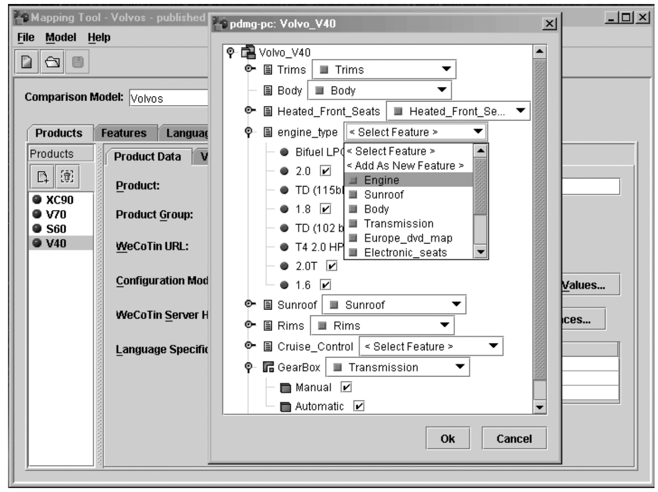

Figure 4. Mapping Tool

during comparison. A configurator URL can be defined for each product so that a customer can navigate to configure it.

User profiles are managed in Mapping Tool. They determine the features and products shown during comparison along with their display order. For example, internal sales might be given access to information that is hidden from customers and the display order be altered to reflect the differing interests of customers and sales personnel.

Mapping Tool maintains an unpublished and a published version of a comparison model. The published version is visible to customers and the unpublished version is intended for preparing major changes such as a product launch.

Practical maintainability of comparison models is further enhanced by the import facilities in Mapping Tool. The facilities enable semi-automatic mapping of configuration information to comparison models and automatic detection of differences between the already imported information and configuration models.

In the import facility, a part or property is mapped to an existing feature or imported as a new feature. Feature values are generated from a chosen set of possible property values or part allowed types. A screenshot of Mapping Tool is shown in Figure 4, the dialog opened in it shows how the mapping is done. The facility also imports internationalization information and cardinalities.

The comparison model being edited in Figure 4 consists of four products, Volvo cars XC90, S60, V40, and V70. The cars were modeled in PCML using information from the Volvo Car Configurator [2003] and then imported to a comparison model as products with Mapping Tool. The configuration models are rather simple and do not represent the actual cars in full.

3.3 Comparison Tool

Comparison Tool supports the selection of a suitable configurable product by providing product retrieval based on user defined constraints and table-based visualization.

There are two ways to choose products for comparison in Comparison Tool. First, the products can be selected directly from a list showing their display names and descriptions.

The list is divided to sections by product group. Second, the customer can use *product search*. With product search it is possible to select products satisfying constraints specified as a set of search criteria. Each search criterion defines a set of allowed feature values for a feature, specified as an enumeration of values or as a range.

Direct selection and product search can be combined: product search can be constrained to the set of currently selected products, or it can be used to extend the set. Direct selection is probably most suitable for expert customers and the product search for non-expert customers.

Product search goes through two phases before displaying the suitable products to the customer. First we perform an ordinary search that selects potential products that allow the specified feature values. This search uses the display names of the features and feature values. Second, our configurator verifies for each potential product the consistency of the search criteria by interpreting them as requirements. The configurator tries to construct a configuration that satisfies the requirements and complies with the configuration model representing the product. Only products that pass this test are shown to the customer. To provide this verification service, the configurator takes as input a set of PCML constraints derived from the search criteria. Each constraint restricts the allowed types of a part or the possible values of a property. Constraints representing the search criteria are transformed to a form suitable for the inference engine by negation: values for properties and part individuals of types that do not meet the criteria are denied from a configuration. Original properties, property values, parts and allowed types used in the constraints can be constructed from the search criteria because mapping information is stored when features values are imported from configuration models. The second search phase is not completely implemented yet.

The products matching to a search or selected from the product listing are shown to the customer as a comparison table. The comparison table consists of columns presenting the products and rows presenting the features. Each table cell displays the possible feature values of a particular product for a feature and cardinality if it is not exactly one. The top right column of the comparison table in Figure 5 displays the feature values of feature 'Engine' for Volvo V40 mapped in Figure 4. Links to additional information about products, features and feature values are also shown during comparison. The dialog in Figure 5 displays additional information for feature value "T6" of feature Engine of the product Volvo XC90. Comparison Tool also can highlight unique feature values or feature values that are available in all products.

The customer can temporarily hide products and features and change the display order in the comparison table. If a customer belongs to a user profile defined in Mapping Tool the products and features are shown in the order specified for the profile. The customer can select a product and start configuring it by following a link below the product name.

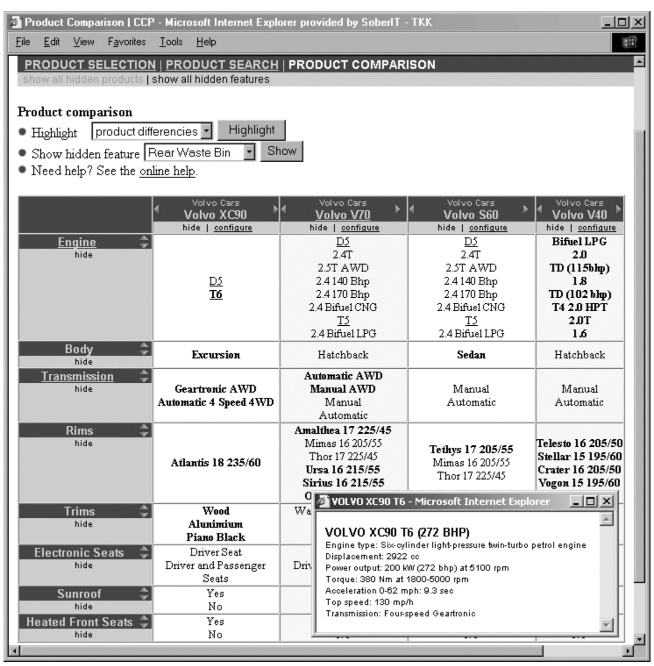

Figure 5. Comparison Tool

4 Discussion and Previous Work

There are many models and tools that support product selection of off-the-shelf products. In [Stolze, 1999] the mechanisms for supporting product selection are divided into three categories: information collection mechanisms, emotional and networking-based product selection, and analytical product selection support mechanisms. We concentrate on the analytical product selection mechanisms as both our own solution and the problem fall into that category. These analytical mechanisms can be further divided into filtering, visualization, and evaluation mechanisms [Stolze, 1999], summarized briefly in the introduction.

Although the difficulty of selecting a suitable configurable product has been recognized [O, 2002; Pargamin, 2002], we did not find any e-commerce systems providing support for the comparison of configurable products.

Filtering mechanisms for configurable products have been used for example by Magro and Torasso. Their sales support system [2000; 2001] assists in the selection and configuration of complex products by filtering out irrelevant products based on some initial customer requirements. After a customer has selected a product the system supports configuration by testing the user selections for consistency and by suggesting missing components. The system resembles ours by its product search capabilities and provides continuous configurator support, but it does not provide facilities for product comparison.

Product filtering on the basis of initial customer requirements, product comparison, and evaluation are all present in the PSC+ system described in [Stolze et al., 2000]. PSC+ resembles our Comparison Tool because it combines feature-based filtering and table-based visualization. PSC+ helps a customer to choose between almost complete configurations of insurance products that consist of pre-selected modules. The insurance products can be further configured only on deductible levels of some modules. In contrast, we compare configurable products at an abstraction level that allows flexible configuring of the product structure and other alternatives. PSC+ includes a scoring-based evaluation functionality lacking from the CCP tool.

Evaluation mechanisms are also used by Bichler et al. [2002] in a system for identifying the most appropriate configurations from a set of configurable offers based on the customer's preferences. The proposed configurations can be fetched from several vendors with configurable offers, but this approach requires company specific integration.

Visualization mechanisms are used by Emde et al. [1996]. They state that customers purchasing telecommunication systems do not usually know the products in advance and have incomplete and contradictory requirements. They approach the problem by supporting the comparison of alternative components of a single selection in an advanced table viewer during configuration. However, the utility is not used for comparing products before configuration.

Personalization is also used to assist in the selection of configurable products. Ardissono et al. identify in [2002] that the customers of electronic catalogs have differing knowledge about the products and services they are searching. They address the problem by customizing the interaction of the configuration process and representation of the configuration selections according to different customer profiles. Sakaguchi et al. present a shopping assistant agent in [1999] that personalizes the information shown to a customer while he chooses items for a product.

In the CCP tool products are represented in a similar way to several XML based e-catalog standards such as BMEcat [Schmitz et al., 2001], cXML [Ariba, 2002], OAGIS [OAGI, 2002], RosettaNet [RosettaNet, 2001] and xCBL [Commerce One, 2002]. However, we did not find any single system utilizing e-catalogs, supporting comparison with a configurator during comparison, and offering the facility to transform configuration models into catalog entries.

When creating configuration models for a set of similar products, a modeling compromise is typically made between creating several smaller and simpler configuration models and between creating a smaller number of larger and more complex configuration models. The variance and complexity of larger configuration models is difficult to handle with our simple table-based conceptual model. This is why our approach favors several smaller configuration models.

Characteristics common to all variants of a configurable product may be excluded from the configuration model, but it may be advisable to include them in a comparison model to provide information relevant for comparison.

CCP tool offers a familiar way to present information about configurable products, which often are quite complex and unfamiliar to customers. As such, it is useful for a single company selling configurable products. However, CCP tool could also be used by a third-party company to offer product comparison services, provided that it has access to the configuration models from separate suppliers.

By importing products from configuration models, existing product knowledge can be utilized efficiently and the comparison models can be kept up to date conveniently. The effort spent is in proportion to the number of parts and properties in configuration models, as for each property and part a decision has to be made about the corresponding feature.

Defining precise price and delivery time information for comparison is difficult because they depend on the selected configuration. However, basic price and delivery time information can be represented as features and feature values.

Presenting deep product structures of complex products flattened into a table may obscure some aspects of the product that could be important when comparing. This could be addressed with structured comparison models and combining hierarchical navigation with table-based comparison of feature values. Further, using a table to present configuration alternatives may be misleading for customers because they could mistake them to be a fixed presentation of the product. Nevertheless, in our view the solution is appropriate for relatively simple products, which are suitable for web-based commerce or when presenting only the main features of more complex products.

5 Future Work

There are several ways to improve the tool. Implementation of the second product search phase in Comparison Tool with the connection to the configurator should be completed. Importing complete and partial configurations from the configurator as products to the CCP tool would be desirable.

Even with the CCP tool there to help, it is possible that a customer may reach a dead end during configuration and realize he has selected an inappropriate configurable product. In such a situation it would be useful to be able to search products in Comparison Tool with the configuration decisions as search criteria.

Currently, the CCP tool imports configuration models expressed with PCML only. We expect that adding support for other configuration modeling languages having a similar conceptualization would be rather straightforward. Further, support for resource and connection oriented configuration conceptualizations should be added.

Because several e-catalog standards represent product data with corresponding concepts, we feel it would not be difficult to integrate the CCP tool with other e-commerce systems by transforming comparison model products to and from XML.

By adding some of the advanced visualization functionality of the FOCUS table viewer [Spenke et al., 1996], the usability of the comparison table could be enhanced.

Implementing similar advanced product scoring and evaluating functionality, as PSC+ of Stolze et al. [2000] does, would be difficult. We compare configurable products at an abstraction level that allows flexible configuration of the product structure and other alternatives. This flexibility significantly complicates meaningful scoring and evaluation of products. However, scoring between complete configurations saved in the configurator would be easier to do.

The CCP tool has not been in production use and its practicality has thus not been fully proven. The tool should be tested with real users, and more complex products with deeper structures.

6 Conclusions

Configurable products are often quite complex and unfamiliar to customers. Product selection support for such products is an increasingly important problem. We did not find any systems in previous work that provide good comparison support fully taking into account the nature of configurable products. In this paper we described a tool that extends analytical product selection of configurable products by applying filtering and visualization mechanisms to help customers in comparing configurable products. The tool is based on a simple conceptual model and it generates semi-automatically comparison data from configuration models and helps keeping comparison data consistent with configuration models. CCP tool offers companies selling configurable products a compact and illustrative way to present product information to their customers.

Acknowledgements

We gratefully acknowledge the financial support from Technology Development Centre of Finland (Tekes) for the work of the authors excluding V.H. We thank Lasse Anderson, Juha Havu, Matias Karvinen, and Anni Toikka for their help in implementing the CCP tool.

References

[Ardissono et al., 2002] L. Ardissono, A. Felfernig, G. Friedrich, A. Goy, D. Jannach, M. Meyer, G. Petrone, R. Schäffer, W. Schütz and M. Zanker. Customizing the Interaction with the User in On-Line Configuration Systems. In *ECAI 2002 Workshop on Configuration*, pp. 119-124, 2002

[Ariba, 2002] Ariba Inc., cXML User's Guide, Version 1.2.008, 2002, <http://xml.cxml.org/current/cXMLUsersGuide.pdf>

[Bichler et al., 2002] M. Bichler, J. R. Kalagnanam, H. S. Lee. RECO: Representation and Evaluation of Configurable Offers. *In Computational Modeling and Problem Solving in the Networked World: Interfaces in Computing and Optimization*, Kluwer Academic Publishers, 2002

[Commerce One, 2002] xCBL 3.5 Product Catalog Specification, 2002, <http://www.xcbl.org/xcbl35/documentation.html>

[Dell, 2003] Dell Computer Corporation, Comparison Chart, cited 6 March 2003, <http://www.dell.com/us/en/gen/topics/sbtopic_004_learningdt_whichdt.htm>

[Emde et al., 1996] W. Emde, C. Beilken, J. Börding, W. Orth, U. Petersen. J. Rahmer, M. Spenke, A. Voss and S. Wrobel. Configuration of Telecommunication Systems in KIKon. B. In *Configuration-Papers from the 1996 AAAI Fall Symposium*, pages 105-110, 1996. AAAI Press

[Garmin, 2003] Garmin Ltd., Marine Product Comparison, cited 6 March 2003, <http://www.garmin.com/marine/compare.jsp>

[Magro and Torasso, 2000] D. Magro and P. Torasso. Description and Configuration of Complex Technical Products in a Virtual Store. In *ECAI 2000 Workshop on Configuration*, pages 50-55, 2000

[Magro and Torasso, 2001] D. Magro and P. Torasso. Interactive Configuration Capability in a Sale Support System: Laziness and Focussing Mechanisms. In *IJCAI 2001 Workshop on Configuration*, pages 57-63, 2001

[O, 2002] Y. L. O. Configuration for Mass-Customisation and e-Business. In *ECAI 2002 Workshop on Configuration*, pages 11-16, 2002.

[OAGI, 2002] Open Applications Group, Inc., OAGIS - Open Applications Group Integration Specification, Release 8.0, 2002, <http://www.openapplications.org>

[Pargamin, 2002] B. Pargamin. Vehicle Sales Configuration: the Cluster Tree Approach. In *ECAI 2002 Workshop on Configuration*, pages 35-40, 2002

[Porsche, 2003] Porsche Cars North America, Inc., Porsche Car Configurator, cited 6 March 2003, <http://www2.us.porsche.com/isapi/english/usa/models/carconfigurator_us/startseite.asp>

[RosettaNet, 2001] RosettaNet, PIP2A1: Distribute New Product Information, Release 1.0, 2001, <http://www.rosettanet.org/>

[Sakaguchi et al., 1999] M. Sakaguchi, A. Sugiura, T. Kamba. A Shopping Assistant Agent for Web-shops. In *IEEE Workshop on Internet Applications*, 1999

[Schmitz et al., 2001] V. Schmitz, O. Kelkar, T. Pastoors, T. Renner, C. Hümpel. Specification BMEcat, Version 1.2, 2001, <http://www.bmecat.org/English/>

[Spenke et al., 1996] M. Spenke, C. Beilken, T. Berlage. FOCUS: The Interactive Table for Product Comparison and Selection. In *Proceedings of the ACM SIGGRAPH Symposium on User Interface Software and Technology*, pages 41-50, 1996

[Stolze, 1999] M. Stolze. Comparative Study of Analytical Product Selection Support Mechanisms. In *Proceedings INTERACT 99*, Edinburgh, pages 45-53, 1999

[Stolze et al., 2000] M. Stolze, S. Field, P. Kleijer. Combining Configuration and Evalution Mechanisms to Support the Selection of Modular Insurance Products. In *ECIS 2000*, Vienna, 2000

[Tiihonen et al., 2003] J. Tiihonen, T. Soininen, I. Niemelä, and R. Sulonen. A Practical Tool for Mass-Customising Configurable Products. To appear in *ICED 2003*, August 19-21, Stockholm, Sweden, 2003

[Volvo, 2003] Volvo Car Corporation, Volvo Car Configurator, cited 6 March 2003, <http://www.volvocars.co.uk/nbs20/default.asp>

[Zones, 2003] Zones Inc, cited 6 March 2003, <http://www.zones.com/cgi-bin/zones/site/guidedbrowse/index.html?cat_oid=-22001>