INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 03, STOCKHOLM, AUGUST 19-21, 2003

SIMPLIFYING CONFIGURATION THROUGH CUSTOMER ORIENTED PRODUCT MODELS

Thomas Leckner, Martin Lacher

Abstract

Mass customization research explores the design and production of individualized products at mass product costs. Model-based configuration systems allow the customer to configure an individualized product. With increasing product and configuration complexity, ways of simplifying configuration for the customer are required. This paper discusses some principles of model-based configuration and introduces the concept of customer-personalized masks for product models. The concept of *masks for product models* provides a basis for personalized, simplified views on a product model, thus simplifying configuration for each individual customer. In this paper, we illustrate the concept of product model masks with a simplified product model for customizable products. Moreover, we introduce two principal ways for the creation of individualized product model masks from customer profile information.

Keywords: mass customization, configuration modelling, product model, personalization

1 Introduction

In recent years, there has been a trend towards individualized products. So far, customers had to adapt their expectations to the actual qualities of available products. Nowadays, customers increasingly demand products, that are tailored to their individual needs and preferences. This forces manufactures to take care of the individual customer's product expectations. Even for products of daily use, more and more customers prefer products that fit their individual preferences best. Therefore manufacturers try to find ways, to adapt their products to each individual customer, while keeping production cost at mass product level. The production of completely individualized products at mass product costs is called *mass customization* [1, 2].

Aiming at mass customization, manufacturers face some obvious problems. One fundamental challenge is the integration of the customer into the design and development phase of a product. Also the customer has to become aware of the properties and functionalities of the product she can adapt to her individual needs. A first approach to overcome these difficulties are online configurator tools, which are offered, for instance, by the large automakers. In some respects, with the configurator tools the customer gets the possibility to virtually assemble a car that is adapted to her individual needs and preferences.

Admittedly the customer's possibilities are fairly restricted within an online configurator tool, because she can merely influence limited attributes of only selected parts of the product. And even here the customer usually can only choose from a discrete number of options. In our opinion such common configuration-tools offer merely an enhanced representation of a product-catalogue, because all possible products have been predefined by the manufacturer, even if the number of possibilities can become quite large.

For complete individualization of products, configurator tools have to provide more design possibilities for the customer. This means that, at least theoretically, even the smallest subcomponents of a product should be customizable by the customer. In addition, we do not want to restrict the customer's choice for an attribute to only a discrete number of options. The customer should have the opportunity, to also select an attribute value within the boundaries of an interval. The boundaries and restrictions of the product and its attributes should be defined by a professional product designer and stored in a product model. Thus, our approach for supporting the customer is a model-based configurator tool. One of the main problems with configuration is, that the customer, by changing attributes and parts of the product, may create products, which cannot be produced, which break existing laws or defy laws of nature. To prevent the customer from doing so, additional dependencies, constraints, rules and restrictions have to be stored in the product model. (see for example [3]).

Usually, such complex product models are only used by professional product developers as part of the design phase of the product family. A special aspect of our contribution is to make product models available for customers for configuration. In the consequence, we have to find ways to simplify existing product models, because customers usually do not have enough product design knowledge to configure complex products. Thus our aim is to realize a model-based configurator tool, which enables the customers to take over a part of a professional product designer's work. Therefore product designer's knowledge has to be represented in the system itself and the customer can play the creative part.

Today's practice is far away from the pure customer-driven product development. Also, we expect, that most customers will get overstrained by designing a product completely on their own, even if they are supported by some kind of «intelligent» product model. We have different ideas of how to meet this challenge. E.g. the customer can be supported by automatically generated personalized recommendations for parts and attributes of the product. Also, we want to support collaborative product-development, where customers can interact with other customers or even with professional product-designers to help them during the design process. However, the product model will be a crucial part of the system in all scenarios. The reason for the importance of the product model is that all knowledge about the product and the restrictions within which the customer can manipulate the product is stored in the product model. To support the customer during the configuration process, we focus on those parts of the model, which are relevant to the customer. A lot of information, which is essential in traditional product models, is normally not necessary for the customer. In the following, we are going to explain and discuss our approach of creating and providing a customer orientated product model.

In Section 2 we are going to explain our approach to a simplified product model. This model will be used to illustrate the concept of *masks for product models*, which is described in more detail in Section 3. In Section 4 different ways of how to generate masks for product models will be discussed. Finally, Section 5 summarizes our contributions and presents our ideas for future work.

2 Customer orientated product modelling

2.1 Requirements

The idea of model-based product-development is not new. There is a multitude of software systems, that are using the model-based representation of a product's geometry, structure, variant and configuration in today's practice.

An example are CAD systems, which often follow the parametric approach, which allows a configuration of the product on the basis of geometric elements on a very detailed level [4]. With a graphical user interface CAD systems help in creating detailed models of single parts or a special instance of the product model. But normally they are hardly able to manage the whole product family with all its variants and configurations.

Also the existing types of product models are normally too complex for the customer-driven configuration and design of products. In our opinion a simplified view on the product model should be provided to the customer. The common solution are configurator systems [5], which can handle the complexity of a product's variety, i.e. its predefined and accordingly «configurable» variants. Usually such configurator tools are used in e-commerce applications, to give a first way of support for the idea of mass customization. But this approach faces some serious drawbacks: (a) Most configuration models have to be created manually, i.e. for each change in the product model there is manual adaptation necessary in the configuration model. (b) Configuration models are primarily based on an existing product spectrum and do not support the development process. Therefore configuration models are often simplified too far, that customers can only choose one special instance of a predefined set of variants. (c) The same configuration model is used for the product configuration of all customers, independently from their individual interests, preferences or faculties.

Especially the last aspect does not fit to the paradigms of mass customization in our point of view. For example an engineer gets exactly the same model for his configuration task, like a lawyer. But the engineer has much more knowledge about product design and most probably he would like to customize the product to his individual needs to a much broader degree, than the lawyer, who has only limited knowledge about product engineering and who is therefore mainly interested is simply adapting the product’s shape, colour or maybe functionality. Because of this our approach is a concept, which can offer a different access and a different view onto the product model to different customers. This approach is realized by the usage of masks for product models.

In Section 3 we will describe our concept of masks for product model in more details. But before we want to introduce a simplified product model, which we will use as an example in the further course of this paper, to explain the concept of masks. Nevertheless we think, that the concept of masks for product models can be applied to any type of product model.

2.2 Simplified product model

In the following we will give an introduction to some ideas of a simplified product model. We suppose, that the product model is created by a professional product designer as an abstract and extensible model of a product family. The product model also contains knowledge about interdependencies or physical conditions of the different parts of the product. The product model can be created by using the *Product Model Editor* (see Figure 1) and defines the restrictions for the product configuration.

On the other side the customer can create her own instance of a product, by using the *Product Editor* tool (see Figure 1). The customer can create her own instance of the product, the so-called product configuration, which serves her preferences and needs and which is based on and valid within the product model, defined by the product model designer. An overview for a possible system architecture is given in Figure 1: One type of user for the system is the Product Designer, who can create product models by using the Product Model Editor and then store these models in a database by using the Product Model Server. The other type of user is the Customer, who interacts with the Product Editor tool (i.e. the online configurator) to create a product configuration. The validation of the product configuration with the corresponding product model is done by the Configuration Manager. After the configuration process the customer can store the product configuration in a database, by using the Product Configuration Server. Another important part of the system is the Customer Profile Server, which handles the data-exchange and enables access to information about customers. Customer Profiles are necessary for different parts of our system and also for the creation of masks for product models (see Section 4). The Filtering Manager is used to create product recommendations for customers, and will not be discussed in more detail in this paper.

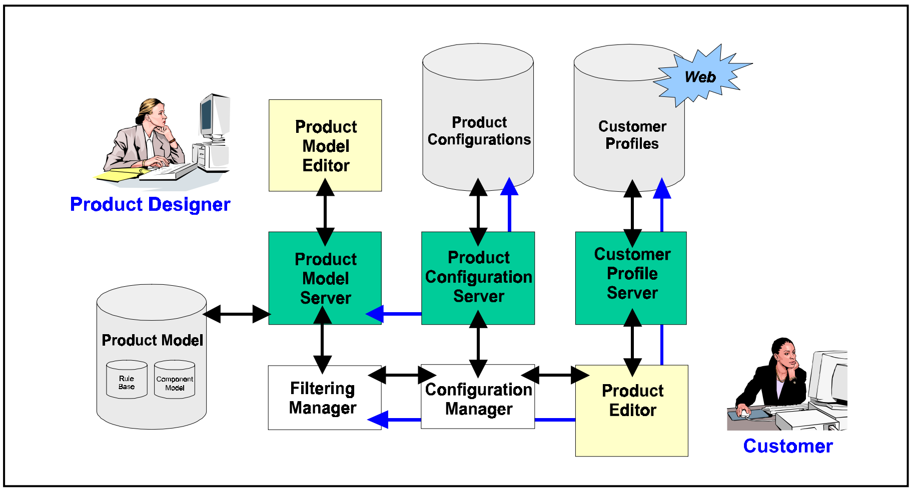

Figure 1: System architecture for the maintenance of product models

Concerning the product model, we assume that every physical product consists of a set of components, the connected structure of which can be described by a component-tree. Each component can in turn consist of a set a of components and/or a set of attributes. Attributes represent degrees of freedom (see Section 2.3) and can thus be based on various data types.

For a better understanding of product modelling the model itself can be divided into four principal parts: (1) The *product meta model* defines the conceptual building blocks, used to model a product spectrum. E.g. the product meta model describes, how components can be connected in the component tree or how dependencies between components can be modelled as certain type of rules or logical constraints. (2) The *product spectrum model* implicitly defines all possible configurations of a product. This is what one might intuitively call a product model. (3) The *component model* describes how different components of a product are assembled by means of the component-tree. Here one also can define constraint sets, describing what rules must be satisfied for a valid component(connection). (4) Finally the *attribute model* defines the types and restrictions of attributes.

Furthermore, we allow alternative components, which enables us to model different versions of actually one and the same component model. An example for alternatives are different types of engines for the product model of a car: A petrol engine will have a different component model than a diesel engine, but the two of them can substitute each other.

2.3 Degrees of freedom

Normally the term «degree of freedom» is a synonym for the measure of variability which merely expresses the number of options available within a variable or space. In a system with N states the degree of freedom is N [6]. In the following we use the term «degree of freedom» to describe those parts of the product model, which can be modified by the customer. Every degree of freedom can have certain restrictions, e.g. an interval attribute, which can only take values within the interval boundaries. In the following we will describe the available degrees of freedom of our simplified product model in more detail:

We distinguish degrees of freedom of a product (a) concerning to the component tree and (b) concerning to the attributes of the components. The first degree of freedom concerning to the component tree are *alternative component models*. Here the customer can choose exactly one from a set of exclusive alternative components, during the configuration of the product.

The second degree of freedom concerning to the component tree are *optional component models*. Those components are not obligatory for a product. The customer can select those optional components so to speak as an add-on to the product. An example for an optional component model in the product model of car is a CD-player. It is possible to integrate such an equipment to a car, but obviously is not obligatory.

Another degree of freedom are the values of the components' attributes, which can be selected by the customer. Here we just want to have a look at two different kinds of attribute models: In the *enumerated set model* the customer can choose exactly one value from a predefined set of possible values. For example the size of the fuel-tank of a car could be available only in three different variants: 45 l, 55 l and 65 l.

Another type of attribute model are *numeric interval models*, where the customer can choose one value within the interval boundaries. For example the varnish of a car might be any colour within the numeric RGB colours ranges.

Also other types of degrees of freedom might be useful for the customer-driven configuration of products. But as we have stated before, we just want to use these restricted degrees of freedom as a simplified product model to illustrate and be able to discuss the concept of masks for product models.

3 Masks for product models

3.1 Principles of masks in product models

For the configuration of her individualized product the customer is restricted to the possibilities provided to her by the degrees of freedom of the product model. The degrees of freedom have been predefined by a professional model-designer. These degrees of freedom are equal for every customer and the customer has to make a clear decision for each undecided degree of freedom, before she can finish the configuration process.

The main idea of masks for a product model is to be able to flexibly adapt the product's restrictions to the individual preferences and needs of a specific user. By the usage of masks the system can provide different views onto the product model to different customers. Thus, degrees of freedom available and visualized for a certain customer might be quite different to the degrees of freedom of the same product model available and visualized for another customer. An intuitive example for the usage of masks on product models can be given for the numeric interval attribute. For example the possible values for the power of an engine may be constrained to [40...300 kW] in the product model. However, the restriction for a specific user may result in possible values within the interval [110...200 kW]. The restriction for another user my result in the interval [80…130 kW].

We call the restricted product model as *mask for a product model*. In Section 4 we will explain how to generate such a mask. In any case the mask for a product model is a subset of the original model. This subset is created by using the customer's profile. Therefore a mask is a personalized view onto the degrees of freedom of a product model.

The concept of masks also allows the implementation of special recommender systems, helping the user to overcome the complexity of the configuration and design process.

If a customer demands product features, which cannot be realized within the degrees of freedom of the current mask, the model will be gradually enlarged back to the original product model, at least for the specific degree of freedom. If the demanded feature also cannot be realized within the restrictions of the original product model, the model can be adapted manually by a product designer if wanted or necessary.

3.2 Examples of masks in the simplified product model

In the following we will show, how each of the available degrees of freedom can be restricted by the usage of masks for product models.

First of all we have a look at the degrees of freedom concerning to the component-tree: For *alternative component models* the number of alternatives can be restricted to a smaller number. In the extreme the mask can break down the degree of freedom, when the available number of alternatives is reduced to simply 1. Then the customer has no choice left and the degree of freedom does no longer exist.

Nearly the same can happen for *optional component models*. By the use of a mask the degree of freedom of an optional component model can disappear, either to permanently select or permanently deselect a specific component model.

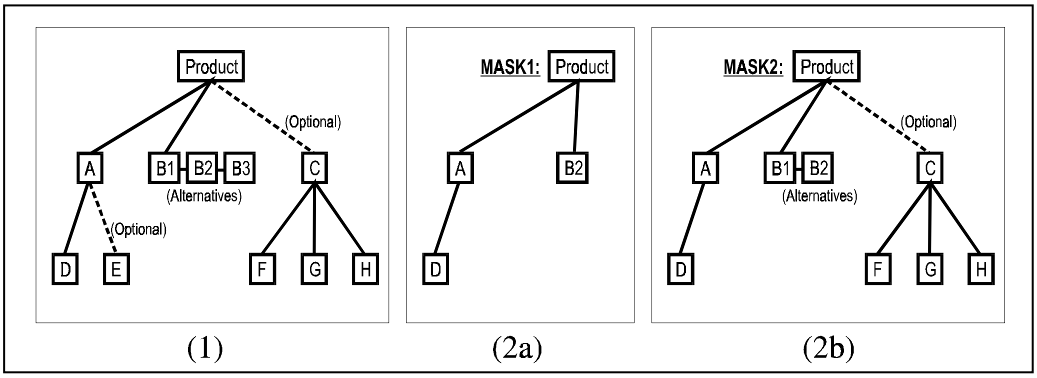

Figure 2: Masks for alternative and optional components

In Figure 2 the component-tree of a product model is illustrated and two different masks for the product model are shown. In picture (1) the whole product model is visualized. The picture (2a) shows MASK1, a very simple mask of the product model. The degrees of freedom are much less here than in the original product model, especially because every component model will also carry a set of attribute models, representing further degrees of freedom. Finally picture (2b) shows MASK2, another valid mask for the product model in picture (1). Here only one alternative component model and one optional component model are gone, compared to the original product model in picture (1).

Masks for product models can also influence attribute-models: For the *enumerated set model* the consequences of using masks are nearly the same, than for alternative component models. The number of possible attribute values will be restricted by the mask and in the extreme the degree of freedom will get lost, when the attribute is set fix to a specific value.

For *numeric intervals model* it gets a little bit more complicated. Numeric intervals are constrained by an upper and a lower boundary. In a first step a mask will simply adapt these boundaries to define a subinterval. This idea is illustrated as MASK1 in Figure 3.

But also it is possible to split the original interval into two or even more independent intervals, each of them being a subinterval of the original. The mask of such a numeric interval model will be the union of all the independent subintervals. This kind of mask of a numeric subinterval is illustrated as MASK2 in Figure 3.

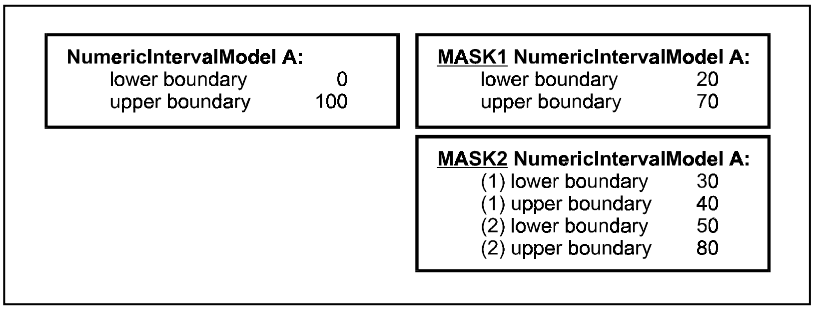

Figure 3: Masks for alternative and optional components

To realize this kind of mask we either need a new type of attribute model, or we have to use the logical operator OR to connect the independent subintervals. We prefer the second way, because in our point of view it is not practicable, that masks can create new types of degrees of freedom. In the consequence a mask is longer only a restricted product model, but can consist of a set of models, connected by logical operators. We will not discuss this problem in more detail in the further course of this paper.

4 Generation of masks for product models

4.1 Customer profile information

The usage of masks can be seen as a possible way of personalization of product models, because the view on the degrees of freedom differs from customer to customer. Up to now we do not know another approach for personalization in product modelling.

Generally personalization is about selecting or filtering objects for an individual by using information about the individual [7, 8]. So before we explain how to generate masks, we want to give an overview to the necessary and useful information in customer profiles. In our application area customer profiles consist of: (a) Basic and demographic attributes, (b) information about specific product interests, (c) information about general interests, (d) information about relationships to other customers, (e) information about the buying history and usage/interaction behaviour with the Product Editor, and (f) ratings of products, product components and certain attributes. Also the problem of customer profile acquisition is one of our main research tasks, however this topic will not be discussed in this paper.

Instead we suppose, that the system can identify customers and can somehow collect or import all necessary information about customers. That information will be used to create personalized masks for product models. In the following we will explain some possibilities for the generation of masks for product models.

4.2 Stereotyped generation of masks

In a first approach predefined rules are used to generate a personalized mask for the product model, due to the information in the customer profile [9]. These predefined rules can be compared to stereotype-building in marketing. For instance, the professional model designer on the one hand can define, that a very detailed product model will be provided to the stereotype ‘engineer’, while a very restricted product model will be provided to the stereotype ‘lawyer’. On the other hand the enterprise can also define the criteria to assign a customer to a certain stereotype, depending to the information in her customer profile. Especially demographic attributes and information about the customer’s interests are used to assign a customer to a stereotype (see Figure 4).

```
switch (stereotype(customer)) {
     // assign stereotype to product model
     case "engineer": ProductModel=MASK1; break;
     case "lawyer": ProductModel=MASK2; break2;
     ...
}
```

```
public String stereotype(UserProfile customer) {
     String job=customer.basic.business.jobtitle;

     // definition of stereotype "engineer"
if ((job=="engineer") or (job=="machanist")
        or (job=="electrician") or (job=="designer")
        or (job=="natural scientist")
        or (customer.interest.computers>70))
     return "engineer";

     // definition of stereotype "lawyer"
if ((job=="lawyer") or (job=="magistrate")
        or (job=="judge") or (job=="tax inspector")
        or (customer.interest.computers<30))
     return "laywer";
     ...
}
```

Figure 4: Example for assigning stereotypes

On the left part of Figure 4 we have assigned masks for product models in a technical notation to the two stereotypes ‘engineer’ and ‘lawyer'. In right part of Figure 4 rules are defined, by which a customer can be assigned to a predefined stereotype depending on her profile information. Here only information about her job title and her interest in computers is used.

This approach can be compared to rule-based filtering in recommender systems [9]. But the result here is just a restriction of the degrees of freedom of the product model and not a product configuration with exact choices for each degree of freedom.

4.3 Generation of masks by dynamical customer clustering

In a second approach we want to introduce a concept to create masks without the necessary effort of the manufacturer. Indeed the idea of dynamically generation of masks is, that the manufacturer does neither have to define stereotypes nor rules to assign a customer to a certain stereotype. Instead the system can create customer clusters automatically. Therefore, demographic data and information about interests in the customer’s profile will be compared to the profiles of other customers, on the one hand. The result of the analysis is a set of similar customers. On the other hand the information about relationships to other customers can directly be used to get similar customers.

Given the set of similar customers the information about previously bought products and the information about the usage behaviour can be computed to restrict the product model to the most probable preferences and needs of a specific customer.

This approach can be compared to collaborative filtering [10]. Same to Section 4.2 the result of the procedure is not a product recommendation with exact choices for the values of the degrees of freedom, but a restriction of the degrees of freedom of the product model.

5 Conclusion and future work

In this paper, we have presented first concepts from a project, the aim of which is the creation of a system for supporting customers during the configuration of individualized products. We have presented a new approach for simplifying product configuration through customer orientated product models.

The main contribution of this paper is the concept of masks for product models, to personalize product modelling and provide individualized views onto the product model for different customers. Moreover, we presented two different principal approaches towards product model mask generation: via static customer stereotypes and via dynamic customer clusters. The proposed approach is the first to integrate model based product configuration with user profile based personalization techniques. In addition we integrated the basic approaches into a system design that includes other important components like customer profile management, product data management and product data filtering. In our work, we have focused on how to support the customer in the difficult design process.

The concepts and ideas we have presented in this paper have partly been realized in a prototype. We have implemented a product model editor and we are currently realizing an online configurator tool (product editor), enriched by personalized recommender systems, which allows customers to configure and design individualized products in a simplified fashion. The product model and the concept of masks for product models we proposed will be an important backbone of our configuration application.

6 Acknowledgements

This work is part of a larger research effort towards local production of individualized products funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) in the program SFB582 – “Marktnahe Produktion individualisierter Güter”. See http://www.sfb582.de/ for more information about the whole research effort.

References

[1] Piller F., “Mass Customization“. Ein wettbewerbsstrategisches Konzept im Informationszeitalter“, Gabler Deutscher Universitäts-Verlag, Wiesbaden 2001

[2] Pine J., “Mass Customization: The New Frontier in Business Competition”, Boston: Harvard Business School Press, 1993

[3] Soininen T. and Niemela I. And Tiihonen J. and Sulonen R., “Representing Configuration Knowledge with Weight Constraint Rules”, Proceedings of. Spring Symp. On Anwer Set Programming: Toward Efficient and Scalable Knowledge, March 2001

[4] Amft, M., „Phasenübergreifende bidirektionale Integration von Gestaltung und Berechnung“, TU München, Diss. 2002

[5] Rogoll T. and Piller F., Konfigurationssysteme für Mass Customization und Variantenproduktion Strategie, „Erfolgsfaktoren und Technologie von Systemen zur Kundenintegration“. München: ThinkConsult 2002.

[6] Krippendorf K., A Dictionary of Cybernetics, Part of: Web Dictionary of Cybernetics and Systems, at URL: http://pespmc1.vub.ac.be/ASC/indexASC.html

[7] Schubert P. and Koch M., “The Power of Personalization: Customer Collaboration and Virtual Communities”, in: Proceedings of the Eighth Americas Conference on Information Systems (AMCIS), Dallas USA 2002.

[8] Levy A. and Weld D., “Intelligent Internet Systems”, Artificial Intelligence, Vol. 118, number 1-2, 2000, pp.1-14.

[9] Adomavicius G. and Tuzhilin A., “Expert-Driven Validation of Rule-Based User Models in Personalization Applications”, Data Mining and Knowledge Discovery, Vol. 5, number 1/2, 2001, pp.33-58.

[10] Ungar L. and Foster D., „Clustering Methods for Collaborative Filtering“, Proceedings of Workshop on Recommendation Systems, AAAI Press, Menlo Park California, 1998

[11] Helander M. and Jiao J., “Electronic Product Development (ePD) for Mass Customization”, Proceedings of the 6th Intl. Conf. on Work With Display Units – World Wide Work (WWDU2002), H. Luczak, A. E. Cakir, G. Cakir (Hrsg.), Berchtesgaden 2002, Germany, pp. 17 – 19.

Corresponding author:
Thomas Leckner
Technische Universität München
Institut für Informatik
Boltzmannstr. 3
D-85748 Garching
Germany
Tel: Int +49 89 289 18650
Fax: Int +49 89 289 18657
E-mail: leckner@in.tum.de
URL: http://www11.in.tum.de