Integrated Computer-Aided Engineering 10 (2003) 163–175
IOS Press

Business models to offer customized output in electronic commerce

Christian Scheer[a], Torben Hansen[b] and Peter Loos[a]
[a]*Information Systems and Management, Johannes Gutenberg University Mainz, D-55099 Mainz, Germany*
Tel.: +49 89 382 33802; Fax: +49 89 382 7033802; E-mail: {cscheer, loos}@wiwi.uni-mainz.de
[b]*Information Systems and Management, Chemnitz University of Technology, D-09107 Chemnitz, Germany*
E-mail: torben.hansen@s1998.tu-chemnitz.de

Abstract. Seller-driven business models (e.g. online bookstores) have been successfully implemented and concretized in Electronic Commerce both in practice and science in the last years. In contrast to this we can depict that more customer-driven business models are implemented in the beginning. One major problem of customizable products and services in Electronic Commerce can be found in the adaptation of the human advisory activity which is inevitable in the traditional sale. For this reason we depict the customization in the customer's view and the corresponding business models in electronic markets. Main focus will be on the improvement of the communication interface between customer and seller in order to better specify the output, especially for customer-driven output. At this point we suggest an IT-enabled consulting component which creates predictions for the customer's specification by using association rules.

1. Methodology

First of all we classify customer-oriented output in customer's view using two parameters and derive classes of seller-driven output, customer-oriented output and customer-driven output (Section 2). On this basis we discuss corresponding business models in Electronic Commerce (Section 3). We will find out that further research is needed concerning integration of customer and customer-driven business model. Hence we suggest an extended configuration process (Section 4). In order to realize an IT-enabled consultation interface within the configuration process we discuss existing concepts to create predictions within the specification (Section 5) und suggest an algorithm using association rules (Sections 6 and 7).

2. Classification of customer orientation

Many of the so called seller markets are changing to buyer markets. This leads to an enhancement of customer-centered activities on production-oriented markets. As a result there is a demand for everyday products and services (in the following abbreviated with the term output) as well as for individualized benefits on consumer goods and supplies.

Customer orientation in the seller's view involves all interactions (both physical and informal) between the seller and the customer which offer customer value. "The key to success is to maximize value to the customer and successfully implement the changes that make maximum customer value a reality for any organization in any industry" [1]. To create customer value the seller can influence different interfaces related to the customer. Relevant interfaces can be depicted by the so called seven C's of customer relationship management:

– Customer Care (high-quality and comprehensive service)
– Convenience (comfort and simplicity in offer)
– Content (persuading and high-quality contents in addressing the customer)
– Cash (suitable cost and performance relation)
– Component Integration (integration of IT-enabled interfaces in the customer interaction and in back-end systems of the seller)

ISSN 1069-2509/03/$8.00

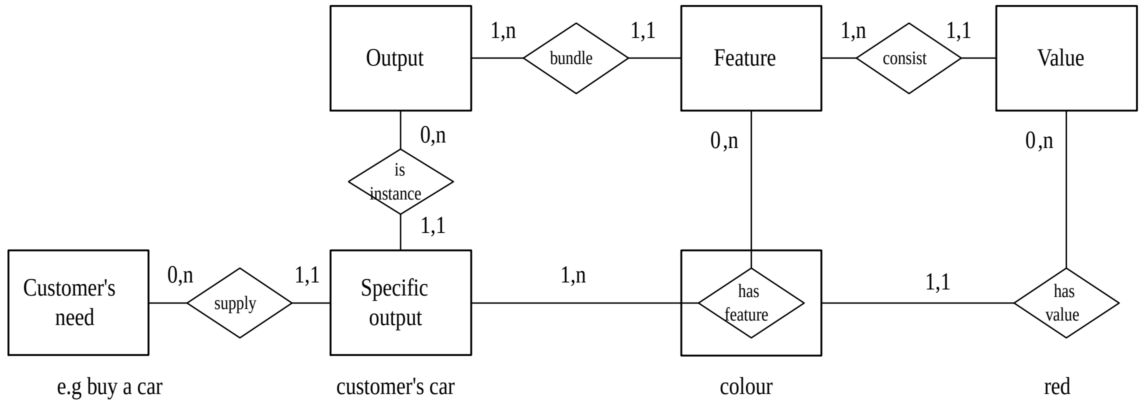

Fig. 1. Relation between customer's need, output, feature and value [24].

– Customization (adaptation of the offer to individual needs)
– Communication (simple and fast communication interface)

High effectualness can thereby be found between customization of output and customer value. Apart from other factors, the customization results in the individualization of output and directly satisfies the customer's need.

The customization is done in the seller's value chain where output can be customized in research and development, production, marketing and sales. The degree of individualization depends on the date where the customer's specification affects the value chain activities (so called freeze point). In this context, late consideration of the customer's view is called soft customization, the earlier one hard customization [20].

In order to classify the spectrum of customized products and services it is necessary to define appropriate parameters [9]. Reichwald and Dietel describe customer orientation issues focusing on production. They differentiate the complexity and the variability of tasks in the production program [23]. Pine et al. use the alteration rate of products and processes to distinguish between standardized and customized products [21]. However, the success in customer orientation will be granted in adaptation of customer's needs to output. Therefore it is important to measure the personalization from the view of the customers [18]. At this point we need parameters which describe the customer's felt adaptation.

A suitable parameter is the degree of individuality. It describes the orientation of output to a customer's individual need according to his or her personal situation. The individuality arises with the individual content or value of output. However, the relation between the individuality of output and customer's need depicts only a single feature of output because different features can have different levels of individuality. Furthermore, a customer would like to look on various features in order to find a personalized product or service [10,11]. The features describe all parts (e.g. product attributes, price, colour) of output which make a difference to a customer. In this context an additional parameter has to be established: the degree of complexity. Complexity depicts output from a multi-layered basis. It describes the variety of different features of output. Figure 1 shows the relations between the customer's need, output, feature and value.

As a result we note that a customer's focus can be described with the felt individuality and complexity of output. Furthermore it is possible to break down the parameters by a granular gradation. First of all we want to look at the parameter individuality. It depicts the number of allocatable values of a feature:

– No individuality: the value of a feature is fixed and can not be changed (e.g. one unchangeable colour of a car).
– Limited individuality: the value of a feature can be chosen from a pre-defined selection which offers more than one value (e.g. 5 colours are selectable).
– High individuality: the product is unique, there are no restrictions for the specification of the value (e.g. self allocatable colour).

The complexity describes the number of allocatable features of an output. A break down of the parameter complexity will look like this:

– No complexity: no feature can be chosen (e.g. interior, engine and colour of a car are not assignable).

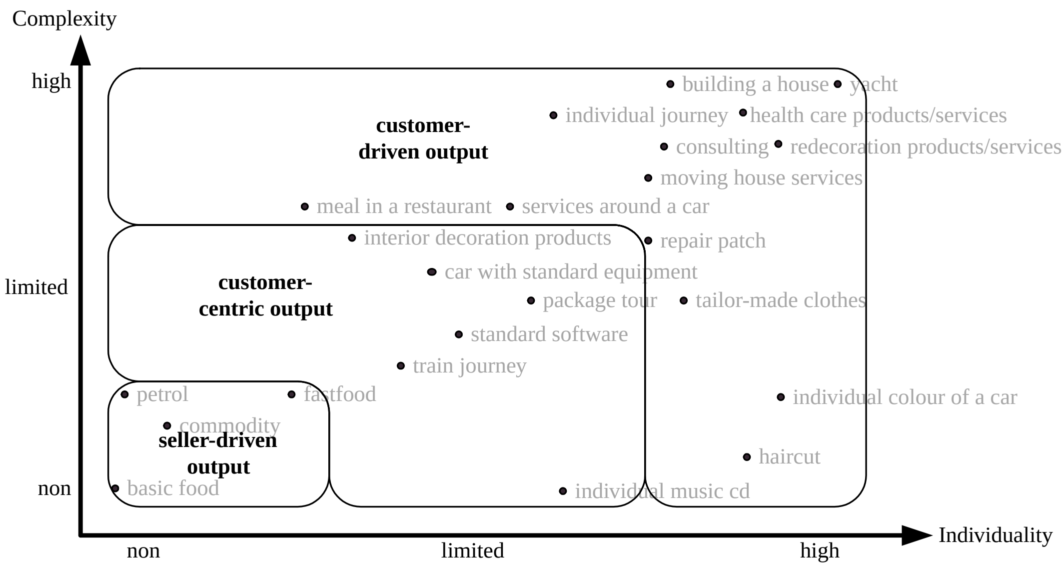

Fig. 2. Classification of customized output from a customer's perspective [24].

– Limited complexity: the features can be chosen from a pre-defined selection which offers at least one feature (e.g. mutual dependent specification of colour and interior).
– High complexity: there are no restrictions for the design of features. The customer can determine the features (e.g. the construction of the car can be designed).

We are now able to transfer the parameters and the granular gradation into a matrix (cf. Fig. 2).

Furthermore we differentiate between three classes of output to classify the different degrees of individuality and complexity. These classes focus on the releasing moment of manufacturing which can be customer-driven and/or seller-driven:

– Seller-driven output: it is manufactured and standardized independently from individual customer's need. The production process is seller-driven.
– Customer-centric output: it offers a number of pre-defined options. The customer can customize the output within these options. The production process is both seller and customer-driven.
– Customer-driven output: it allows the customer an individual design of the output. The production process is customer-driven.

See Fig. 2 for a compiled classification of personalized output from a customer's perspective. We are now able to measure the felt adaptation in three classes of output by the parameters individuality and complexity.

3. Business models to offer customized output in electronic commerce

The requirements of individualized output have their sources in customer-driven markets and the corresponding business models. Specifically business models for Electronic Commerce apply modern information technology to shape the seller's process and the interface to the customer's process (cf. Fig. 3).

Business model and its interface to the customer's process are different depending on the degree of personalized output. The reason for this can be found in the temporal consideration of customer's specification which determines both seller's process and customer's process. On the basis of the characteristics of personalized output (cf. Section 2) we will therefore discuss mentioned differences in the following.

The *seller-driven output* can be completely controlled by the seller and is manufactured independently from the customer's needs. The seller's processes and the organizational structures can be designed in a seller-driven environment. The model of mass production realizes the seller-driven output. It leads to standardized output concerning design and distribution [9]. Mass production pursues the principle of Henry Ford: "You can have any color car you want as long as it's black" [20]. The production of variants can also be used to realize seller-driven output with limited personalization. Here the customer gets products or services in different variations of features which are set by the manufacturer and cover average individual needs. Each

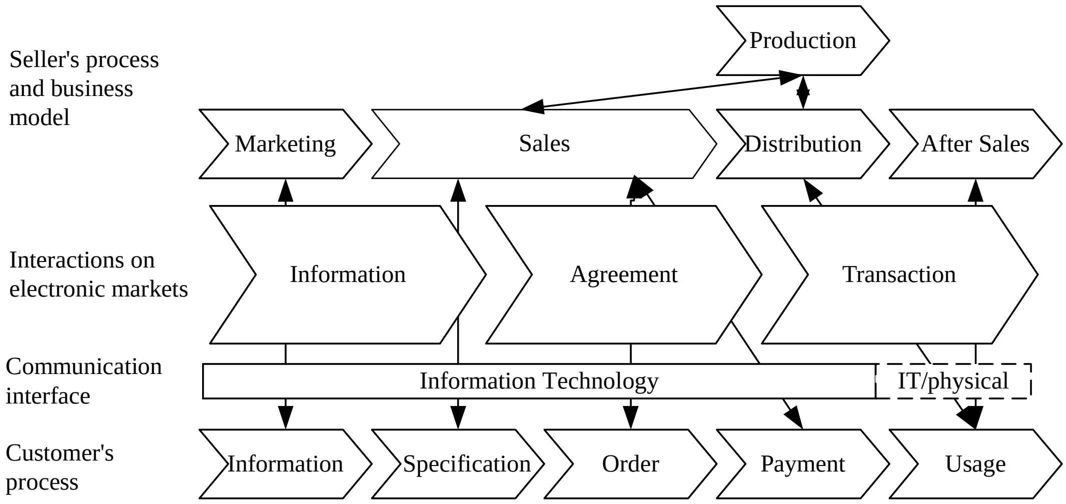

Fig. 3. Customer process and seller process and the communication interface between both processes in Electronic Commerce.

variation is made for a small group of customers. This can lead to a high number of variants which won't fit exactly the customer's needs [19].

In the customer's process the customer selects a completely seller-driven output. This can be explained by the example of an used car purchase. Here the customer can not individually manipulate the output, apart from the negotiation of the price. The configuration space is zero since neither features or values can be assigned. Therefore there is no need for adaptation. In Electronic Commerce the seller-driven output is specified using catalogs in which all available products and/or services and further information are categorized.

A *customer-centric output* will be realized in a process which is customer and seller oriented. At the beginning of the value chain the business processes and the organizational structure are driven by manufacturer's interests. This changes at the order penetration point, also called freeze point. At this point the seller integrates the customer's specification with the production process. In general, the specifications of the customer are integrated as late as possible. "-Value chain customization begins with the downstream activities, closest to the marketplace, and may then spread upstream. Standardization, in contrast, begins upstream, with fundamental design, and then progressively embraces fabrication, assembly, and distribution" [9] Starting at the order penetration point, the output will be adapted within a range of pre-defined options (i.e. values and features) to fit customer's needs. Another way of customer orientation is to extend the standardized product or service with additional value-adding services [20]. The concept of mass customization can be used to implement the customer-centric manufacturing of output "with enough variety and customization that nearly everyone finds exactly what they want" [20]. Finally, mass customization offers the customer a number of pre-defined values. They can be used to define the also pre-defined features of the output [19]. Individuality can also be created with additional services, a specific degree of delivery service and a kind of product image. Decisively the customer chooses the options which are relevant for his satisfaction. The resulting complexity for the manufacturer can be reduced by the mass production of modular output, by new concepts of production, usage of information technology, supply networks and additional points of order penetration [19,20].

Specifying the customer-oriented output the customer can influence the output concerning its features and values. The customer evaluates and selects offered options in order to specify his custom output. Using the example of a car configuration we can depict the customer's process. The customer specifies different features (e.g. the colour) within pre-defined values (e.g. colour red, green, blue) and can thereby configure his individual car. The integration of the customer process can be realized by configurators which offer all available options and record the customer's decision.

A customer-driven output will be realized with the degree of individuality and/or complexity determined by the customer. The organizational structure must be designed order specific to combine required resources and functions. The trigger of all activities is the customer's order [24].

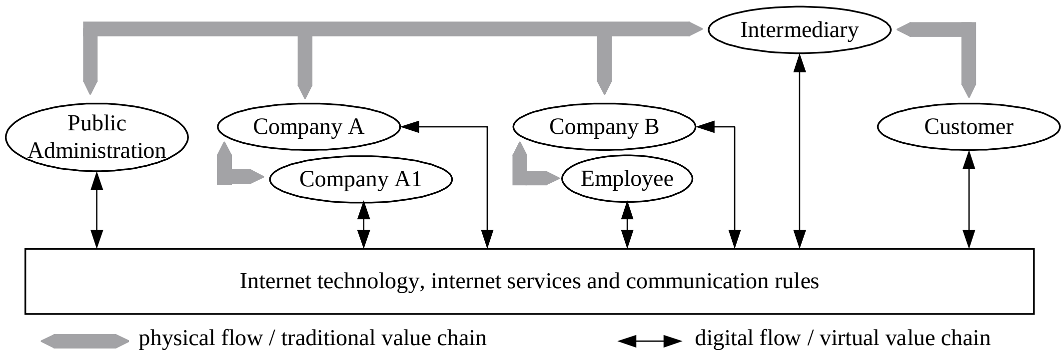

Fig. 4. Internet business model to create the customer-driven output [24].

Implementation of customer-driven output has been discussed in literature in different concepts. One category of concepts suggests internet-based business models in which different suppliers are aggregated in order to co-produce custom output [16,22,24]. The concept of Scheer and Loos for example describes an order-specific cooperation of suppliers which is initialized and coordinated by an intermediary (cf. Fig. 4). The intermediary records the customer's specification and splits his requirements in different orders. This is necessary because of the high degree of individuality and/or complexity of the output which can not be realized by one supplier. The suppliers are mostly integrated by information technology in their value chain and supply chain relationships. They produce parts of the output (concerning the order) within their resources and core competencies. All parts are integrated at the end of the value chain by the intermediary and offered as one single output to the customer [24].

The customer has more influence to specify the output in contrast to the process of customer-centric output. To realize high individuality and/or high complexity in the output the seller offers some leeway in the specification which is not restricted to pre-defined features and values. At the beginning of the specification process the customer can configure the output within the pre-defined options of a customer-centric output. If the customer is not content within the offered range of features and values, the process will be extended. The extended configuration process provides the functionality to compose an individual model of output by adding and deleting features from a repository, adding individual features and creating individual values.

In summary we can state that seller-driven output and customer-oriented output are successfully transferred to business models in Electronic Commerce. The theoretical base is available which principally consists of knowledge in doing traditional commerce and the enrichment in doing business with information technology. Questions can be found in the implementation of the customer-oriented output in Electronic Commerce. At this point we see a range of research fields starting at the organization of virtual supply chain and value chain relationships up to the specification interfaces for output. A serious question in our point of view concerns this specification interface in Electronic Commerce.

4. Specification of customer-oriented output in electronic commerce

Customer's integration has a special significance in Electronic Commerce as human interaction needs to be reproduced comparable to traditional brick and mortar sales using information technology. In case of seller-driven and customer-centric output, this will not impose a problem, as the options are set fixed by the seller and customer's choices are constricted.

In this context the seller can use his knowledge about output, features and values to directly control the retrieval and specification process of the customer in advance. Examples can be found at ordering processes within online bookstores or computer configurations at online-distributors.

To offer customer-driven output, an additional focus on customer integration is required that can be achieved by individual determination of features and/or values themselves. At this point, the specification process cannot be controlled entirely by the seller anymore. The customer needs to be offered more possibilities to influence design within his or her model of output. In literature several tools to realize the customer's design in the specification (using information technology) have been proposed [6,8,13–15,17,25]. They depict

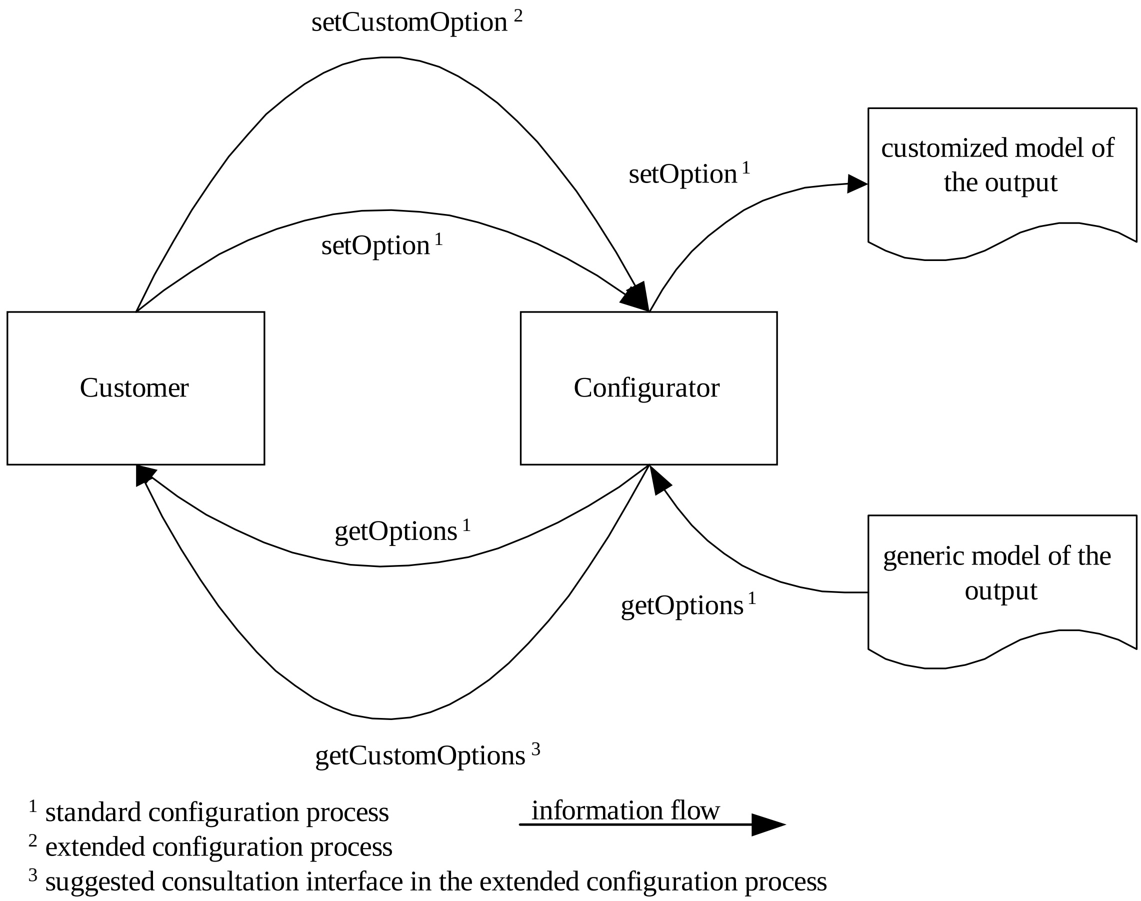

Fig. 5. Extended configuration process of customer-driven output.

an extended configuration process. On a closer look from customer's perspective, it seems, that customers are offered powerful tools to specify the model of output. On the other hand we ascertain that there is no assisting guidance to precede or to go along with the design process. In a logical sequence there should be a consultation process which offers probable options before entering the design process. This can avoid all those specification cases where the customer wants to quit the process because he does not find what he wants in a complex configuration process. To face this problem, we propose an consultation interface within the extended configuration process.

Figure 5 clarifies the cohesions. Within the traditional configuration process for customer-centric output, customers are first offered a set of pre-defined options (features and/or values) (getOptions) and are then asked for their choice (setOption). Main task of the configurator is to create a customized model of output with reference to options in the generic model. The specification process for customer-driven output differs in such a way (see Section 3), that additionally tools are offered, which allow for more influence on the design process (setCustomOption). From our point of view, there needs to be an additional instance for advice (getCustomOptions) to ensure, that the customer is able to complete the configuration successfully. The task of the configurator here is to create a customized model with the possibility of additional leeway in specification with reference to constraints in the generic model.

The consultation interface in the extended configuration process for specification of customer driven output shall therefore be described in the following. Existing approaches in the field of configurators only offer additionally static informational resources in order to assist the customer in the complete specification process. It would be more promising though, if the helpdesk of the information system would be modeled on the archetype of the human interaction partner, so that the system gets enabled to give actual individual user support in each step of the configuration process.

5. Common approaches for consultation interface in configuration tasks

In order to realize individual user support within the extended configuration process, we will present a consultation interface, which creates a proposition for any desired option value based on the individual configuration goals and past configurations. The objective here is to complete the configuration process in terms of the customer and to keep the customer in the process.

At first it is fundamental to understand customer's goals of configuration. Therefore a measure needs to be defined characterizing the preferences of the user. On the base of the measure, preferences can be analysed and used for user individual support. Basically four approaches are to be considered within this context (cf. Fig. 6).

The horizontal dimension in Fig. 6 describes how to collect information about the customer. A customer model concerning configuration goals, preferences and resources can either be built by gathering information via direct inquiries – the user provides information directly by himself – or via passive observance based on behavior analysis. The knowledge about the customer then needs to be matched with know-how from past configurations (so called experience model) and the product model in order to be able to propose likely option values. Know-how about past configurations can either be in the form of the specific past configuration runs themselves (*memory-based experience*) or in the form of an abstract model (*model-based experience*) based on the collected data. As a result, there exist a variety of approaches, each with different attributes and therefore also different suitability for use with configurations for customer-driven output.

Customer active data collection: A registration process is used to query the customer about his preferences and goals, so that the collected information can afterwards be used as basis for individual support.

– Model-based experience (Method 1): The prediction is based on the assignment of the customer to a pre-defined user class. The classification depends on information provided by the customer [27]. The problem of initially not having any information to apply is avoided, as the predictions are already set before the actual configuration starts.
– Memory-based experience (Method 2): The provided information is compared to information about other users which were collected in the past. The profile matching the configuration goal of the current customer best is searched. In a second step, the corresponding configuration run containing all the option values is used for predictions. Therefore it is necessary to associate completed configurations with the information provided at the registration process.

Customer passive data collection: User's behaviors is assessed based on information provided by the user at the time he or she requests additional help. Hence it causes problems if the user asks for support at the very beginning of the configuration process, since few information about his or her configuration goals are available.

– Model-based experience (Method 3): A model is created based on past configuration runs and is compared to the behavior of the user, who requested a prediction. Different approaches based on different data structures can be applied: a weighted tree structure could be used [12] as well as a model consisting of strict association rules [5].
– Memory-based experience (Method 4): The behavior of the customer is compared to individual past configuration runs of other users. Predictions are based on those option values of the past configuration run matching best the information provided by the user so far [7].

None of the presented approaches can solve the problems imposed by configurations for customer-driven output by it one: If active information retrieval is chosen, the option values are already specified before the actual configuration even starts. Thereby the prediction engine works completely independent from decisions within the configuration process. With passive information gathering it is not possible to create any likely predictions in the first steps of the process, since no information about the user is known yet. The four possible methods can be estimated as follows:

– *Method 1:* The probability of creating a likely prediction which matches user's configuration goals is very little if clustering methods are used because of the large number of possible option values in cases of customer-driven Output. As all information was supplied by the customer himself in advance, there is apparently the need to draw conclusions for every single option value based on that higher level information provided within the registration. This again cannot be possible, as the concept of customer driven output asks for a high degree of user involvement. The gathered infor-

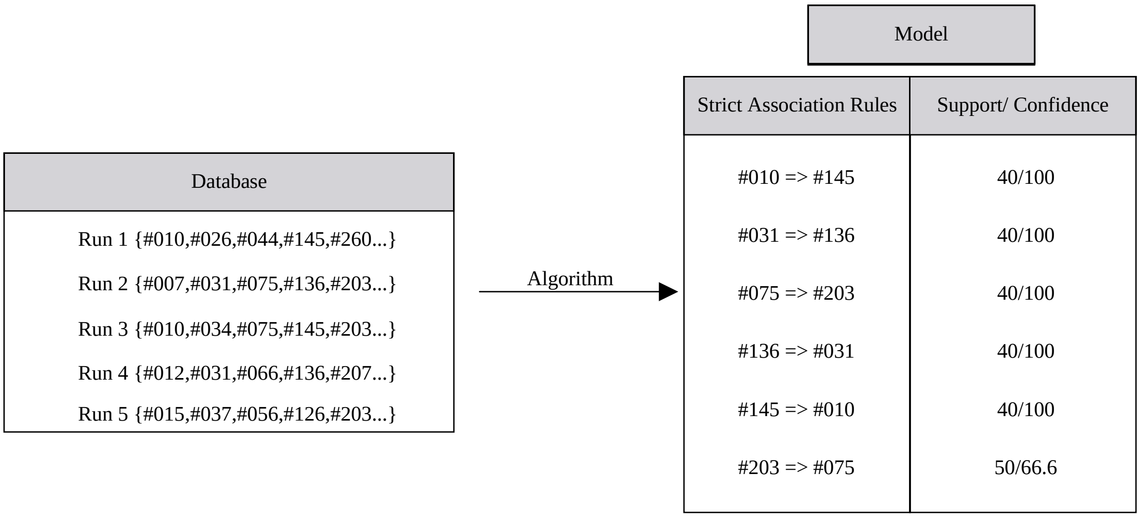

Fig. 6. Simplified depiction of model creation.

mation based on general information necessarily cannot provide that much insight into the preferences regarding the choice between certain option values.

– *Method 2:* Comparable to the previously regarded approach, this method is also better used for configurations with a lesser degree of user involvement. Here, detailed option values need to be derived from general information, which in practice cannot succeed especially if the amount of option values is large. Additionally it is difficult to set priorities among past configuration runs, whose queries fully match the provided information, but came to different configuration results.
– *Method 3:* Likely predictions can be created comparing the provided information within the process so far to a model based on past configuration runs. Four conditions have to be met: (1) The database that the model is based on needs to have a certain minimal size, (2) the user needs to have completed the first steps within the configuration process in order to have provided some information, (3) the user needs to act in a way which at least partially resembles the behavior that users in past configuration runs have shown and (4) a suitable algorithm, which is able to extract cohesions among option values, has to be used in order to create the model [26].
– *Method 4:* Theoretically the comparison with past individual configuration runs is a good approach to create likely predictions. In practice though pure memory based collaborative filtering and the corresponding methods of optimization [30] by reducing the size of the database are not adequate. The necessary search time in larger databases will serve as an additional reason of process abruption itself when using the configurator. Methods aiming at reducing the database size then again cannot guarantee the quality of predictions when used with highly user individual configuration settings.

Because of the mentioned pros and cons of each approach we want to combine several methods in order to create a support mechanism which is able to predict accurately and dynamically regardless of the point in time, when the user requests a prediction in the configuration step.

6. Model-based consultation interface using association rules in the extended configuration process

Our approach consists of a model (see model-based experience in Section 5) which is based on transferring the procedure to determine association rules to the product configuration process. Additionally it uses clustering methods to solve the resulting problems. By doing so, our model can predict far more precisely on a smaller database than it would be possible using linear correlations to similar configuration runs. Predictions are based on the cohesions extracted from all configurations, not just the exactly corresponding ones. Especially in configuration settings offering a large selection of options it is still possible to make an accurate prediction even though there has not been a single previous configuration run, which shows a strong correlation to current user's configuration goals.

Association rules have the purpose of identifying formerly unknown strong relations between objects within a database [2]. First, we want to equate transactions with completed past configuration runs, which are stored in a database. This enables to adapt the support/confidence framework for this purpose. The database shall be composed of completed configuration runs which each consist of several parameter values of the options.

Thresholds are introduced to distinguish between relevant and irrelevant relationships of objects in the database. Therefore we need values to help us select among all existing relations:

- *Support-Thresholds* describe the absolute frequency with which two object values appear together within one stored configuration run measured above all configuration runs in the database.
- *Confidence-Thresholds* measure the absolute frequency of one object appearing in configuration runs in which also the other object appears. It is possible, that objects exist, that appear in about every relation. Therefore several relations including this one object get selected when exclusively considering the support threshold, although there is only very little informational value within those relations. As a result there is the need to include a second filtering method, which also takes into consideration how relevant the information is within the given context to filter out relations that were only picked because of general commonness of at least one of the objects.

We need to consider the necessity of strictly constraining the results as we are only interested in the one best matching association rule for the specific prediction task, and not a considerate amount of relations such as in basket case analysis. Also the query for prediction is highly specified – the option value range is known–, which improves the efficiency of the search, as none of the rules that do not contain any value within the option value range need to be considered.

The creation of the model based on past configuration runs which are now stored within the database is the initial step. An algorithm is applied to the data to find common object relations. In the following the Apriori Algorithm shall be used for that purpose, other derivations such as Apriori TID or distributed algorithms such as the Count Distribution (CD), can be seen analogous [3,4].

The Apriori Algorithm [4] is an iterative procedure, which generates single-dimensional, single-level, boolean association rules. Apriori aims at filtering object relations that satisfy the pre-defined constraints in form of threshold values with the help of iterations of subsets. The algorithm starts out with the examination of the commonness of itemsets with length 1, and then continues based on the results item sets with more elements.

The algorithm can be characterized as follows:

begin
  $M1 =\{$ *itemsets with length 1* $\}$
  for $(k = 2; Mk - 1! = \{\}; k++)$ *do*
   // *Calculation of support values*
  begin
          $Ck =$ *apriori-gen(Mk-1);*
   //*Candidate Generation*
   for all transaction t *in* D
   begin
    $Ct =$ *subset(Ck,t) ;*
    // *Candidates contained in* t
     for all Candidates c *in* Ct *do*
      c.supportcount= c.supportcount $+1$*;*
   end
   $Mk = \{c$ *in Ck I c.supportcount* $>=$
  threshold_support$\}$
  end
end

The absolute frequency of each item in the database with length 1 is calculated within the first iteration. Based on the object quantities with length K, potentially common itemsets, so called candidates, of length $K + 1$ are generated and usually get stored in hash tables for algorithm optimization purposes. Then the frequency of each candidate is calculated and compared with the thresholds: If the value is below the threshold, the candidate is sorted out. In order to constraint the amount of candidates, information generated in the previous iterations is included in the candidate generation process: the apriori attribute is used. This refers to a monotony characteristic, that every subset of a common itemset, which is not empty, has to be common, too.

In reverse, patterns of a defined length, whose support is below the threshold, don't have to be taken into further consideration, as no itemsets can be generated, which possibly could be common. Every $K - 1$ subset of a common itemset with length K therefore has to be common. The algorithm terminates when no further common itemsets can be generated anymore.

Table 1
Data collection in customer model and utilization for prediction

How to get the experience model?	How to get the customer's model?	
	Customer active data collection	Customer passive data collection
Model-based experience	Method 1	Method 2
Memory-based experience	Method 3	Method 4

The overall procedure of generating strict association rules is as follows:

- At first candidates are generated by combining object combinations with length $K - 1$, as long as they differ only in one element.
- In a second step all candidates are checked against the confidence threshold, as not all candidates necessarily need to be strict association rules, though all strict association rules can be found among the candidates in reverse. All candidates, whose confidence value transcends the threshold are accepted as strict association rules in the form $A \Rightarrow B$ and are included in the model.

The result of this procedure is an model of experience, which consists of strict association rules taken into account all past configuration runs within the database (cf. Fig. 7).

7. Consultation process in detail

Whenever the customer needs additional help in form of a prediction, a pattern consisting of three steps based on our previously developed model starts:

1) Based on the query of the user, all association rules within the model are picked, in which exactly one object is within the range of the option values to be predicted.
2) Each relation then needs to satisfy certain requirements: The object in the picked relation, which is not within the range of the option values a) may not be zero and b) has to be previously picked by the customer in the configuration process.
3) At this point, there are precisely three possibilities:

   a) Exactly one relation is found. In this case no further evaluation is necessary and the result can be used as a basis for prediction.
   b) Several or even contradicting relations are found. So it has be decided by a selection criterion such as the confidence value, which relation is most likely to appeal to the configuration goals and preferences of the customer. As a result the relation with the highest confidence value is chosen.
   c) No relation can be found. All relations have been filtered out in the previous steps, which either was caused by overly strict definitions of the thresholds when creating the model or a small-sized database. The foremost mentioned problem is less a technical than a strategical problem: The quality of the predictions is dependent on the amount of strict association rules in the model. The system administrator therefore can lower the thresholds to enforce prediction generation, while the quality analogously gets reduced. The limit concerning technical realization is a certain maximal capacity of the database, which still allows for prompt response times.

Still we face difficulties to make predictions when the customer requests support at the very beginning of the configuration procedure, when no or little information about his goals were previously gathered.

As a solution we propose the usage of *clustering techniques* in combination with the previously developed model in order to be able to immediately generate predictions. Precondition for this is the existence of a registration process in which the customer reveals his top-level preferences, demographic information and vague design ideas about the attributes he wants his product to have. Based on this information the system is able to classify the user into a pre-defined category, whereas the amount of categories is dependant from the desired degree of detailing.

For every user group there is not only one option value pre-defined rather than a ranking which option values correspond more closely to the specific user category.

This is done in order to combine our previously developed model with the clustering technique to further improve our prediction quality.

We introduce a correlation coefficient which measures the degree of compliance between the value chosen via the association framework and the in the regis-

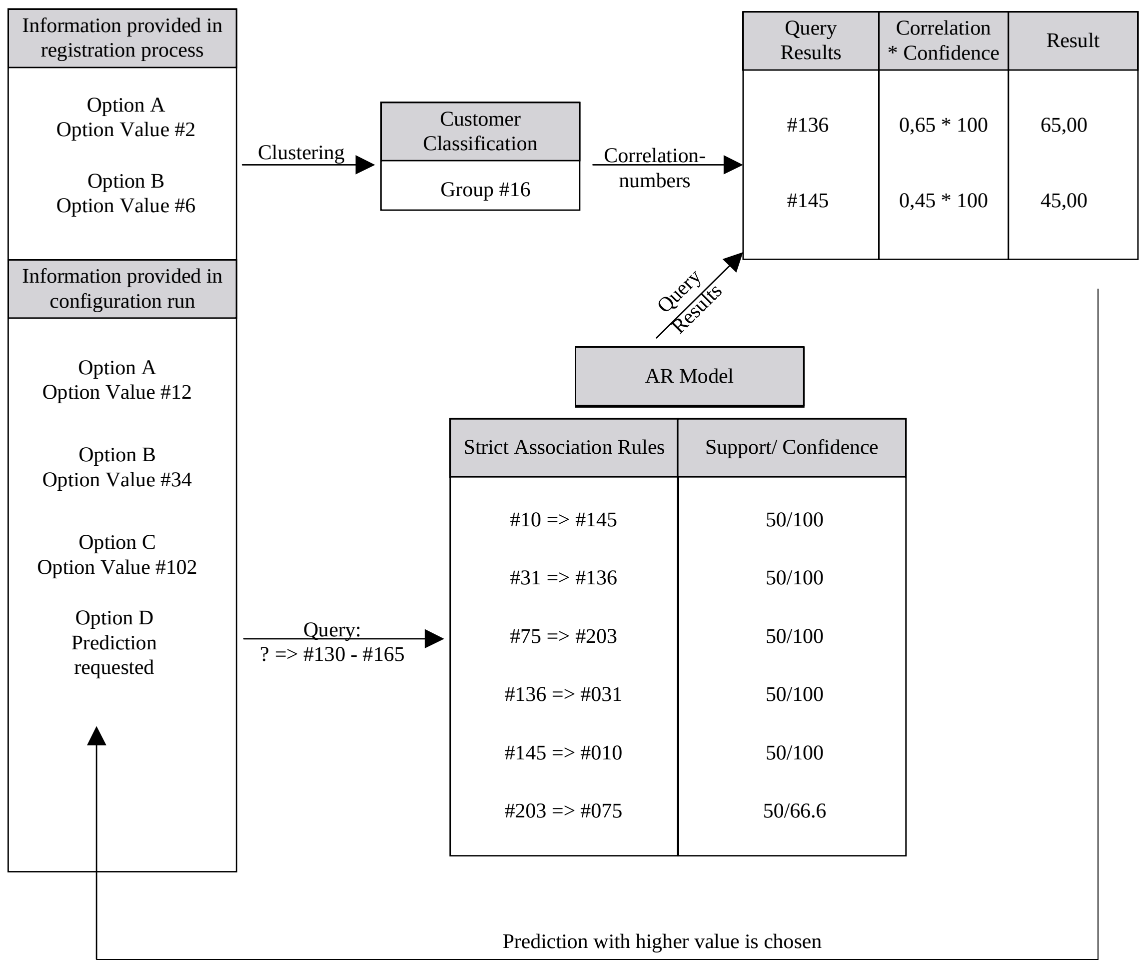

Fig. 7. Model displaying the integration of AR und clustering techniques.

tration process mentioned configuration goal. The final selection therefore is no longer dependent exclusively on the confidence value of the relation, but on the product of correlation number and confidence value; if no matching association rule could be found, the prediction is based solely on the advice out of the clustering routine. This newly calculated product is a composite that tells about how likely the via association rule generated value goes along with the user stated configuration goals. This value therefore unites primary and secondary information retrieval methods. By this it is possible, to intercept wrong predictions which otherwise will appear especially in unusual configurations. The explicitly stated goals of the user featuring top level information gets combined with detailed collaborative based information about the most likely option values, to form a value, which combines best of both worlds:

All possible prediction values derived via the association rule mechanism get multiplicated with a number, stating how likely this prediction value is considering the user stated goals. The higher the total number, the more likely is that the value is an accurate prediction.

When a prediction is requested, a query is issued. The model is searched for appropriate association rules, which include one of the previously selected option values on the left side of the association rule as well as an option value within the specified range on the right side of the rule. If the query was succesful, each of the results get matched with a correlation number manually stored in the system for each customer group describing how likely the results will match customer's configuration goals (cf. Fig. 8).

As stated before, customer-driven output does not require the existence of any pre-defined options, so that it might be impossible to state an exact correlation number for the values of some options in advance. In these cases, a classification based on basic attributes can help to identify cohesions among the option values. Possible methods for realization include pattern matching for design recognition tasks or RGB-ratios for characterization of color [28,29].

The next step in our research activity involves the implementation of the suggested model in the configuration process. This will help us to approve mentioned components and their coactions in the consultation interface.

8. Summary

Starting from the basics of user customizable products and services in Electronic Commerce and related business models, the paper describes the extended configuration process for specification of customer-driven output. In addition to the approaches in literature, a new support component is suggested, that offers individual advice within the specification process. This component includes a prediction mechanism, which is able to generate likely predictions based on previous configuration runs. Thereby we use the strength of object value relations as an indicator for the relevance of an option value within that context. We combine this information with the information provided by the user himself in order to unify the explicitly stated configuration goal with our prediction. As a result, we are able to make likely predictions which are less dependant on both the size of the database and the point in time, when the prediction is requested by the customer in the configuration step.

References

[1] C.B. Adair and B.A. Murray, Breakthrough Process Redesign: New Pathways to Customer Value, AMACOM, New York, 1994.

[2] R. Agrawal, T. Imielinski and A. Swami, Mining Association Rules between Sets of Items in Large Databases, in: *Proceedings of the International Conference on Management of Data*, Washington D.C., 1993.

[3] R. Agrawal and J. Shafer, Parallel Mining of Association Rules, *IEEE Transactions On Knowledge And Data Engineering* **8**(6) (1996), 962–969.

[4] R. Agrawal and R. Srikant, Fast Algorithms for Mining Association Rules, in: *Proceedings of the 20th International Conference on Very Large Data Bases* (*VLDB*), Kaufmann, M., 1994.

[5] Y. Chien and E. George, A Bayesian Model for Collaborative Filtering, in: http://uncertainty99.microsoft.com/chien.htm (10.11.2002).

[6] F. Feldkamp, M. Heinrich and K.D. Meyer-Gramann, SyDeR - System design for reusability, Artificial Intelligence for Engineering Design, *Analysis and Manufacturing* **12**(4) (1998), 373–382.

[7] H. Inakoshi, S. Okamato, Y. Ohta and N. Yugami, Effective Decision Support for Product Configuration by Using CBR, in: *The Fourth International Conference on Case-Based Reasoning*, Vancouver/ Canada, 2001.

[8] U. John, Konfiguration und Rekonfiguration mittels Constraint-basierter Modellierung, Ph.D. Dissertation, Akademische Verlagsgesellschaft Aka, Berlin, 2002.

[9] J. Lampel and H. Mintzberg, Customizing Customization, *Sloan Management Review* **38**(1) (1996), 21–30.

[10] K.J. Lancaster, Consumer Demand: A new Approach, Columbia University Press, New York, 1971.

[11] K.J. Lancaster, Modern Consumer Theory, Edward Elgar, Hants, Vermont, 1991.

[12] W. Lin, S. Alvarez and C. Ruiz, Efficient Adaptive-Support Association Rule Mining for Recommender Systems, in: http://citeseer.nj.nec.com/483133.html (22.08.2002).

[13] T. Männistö, H. Peltonen and R. Sulonen, View to Product Configuration Knowledge Modelling and Evolution, in: *AAAI Fall Symposium Series Configuration* (*Technical Report FS-96-03*), Cambrigde, Massachusetts, 1996.

[14] D.L. McGuinness and J.R. Wright, Conceptual Modelling for Configuration: A Description Logic-based Approach, Artificial Intelligence for Engineering Design, *Analysis and Manufacturing* **12**(4) (1998), 333–344.

[15] S. Mesihovic and J. Malmqvist, Product Data Management (PDM) System Support for the Engineering Configuration Process–A Position Paper–, in: *Proceedings of 14th European Conference on Artificial Intelligence* (*ECAI*)*–Workshop Configuration*, Berlin, 2000.

[16] H. Österle, Enterprise in the Information Age, in: *Business Networking: Shaping Enterprise Relationships on the Internet*, H. sterle, E. Fleisch and R. Alt, eds, Springer, Berlin et al., 2000, pp. 17–54.

[17] H. Peltonen, T. Männistö, K. Alho and R. Sulonen, Product Configurations–An Application for Prototype Object Approach, in: *Proceedings of the Eighth European Conference on Object-Oriented Programming* (*ECOOP*), Bolgona, Italy, 1994.

[18] A. Picot, Ein neuer Ansatz zur Gestaltung der Leistungstiefe, *Zeitschrift für betriebswirtschaftliche Forschung* (*zfbf*) **43**(4) (1991), 336–357.

[19] F.T. Piller, Mass Customization: ein wettbewerbsstrategisches Konzept im Informationszeitalter, Ph.D. Dissertation, Vol. 2, DUV/Gabler, Wiesbaden, 2001.

[20] B.J.I. Pine, Mass Customization–The New Frontier in Business Competition, Harvard Business School Press, Boston, Massachusetts, 1993.

[21] B.J.I. Pine, B. Victor and A.C. Boynton, Making Mass Customization work, *Harvard Business Review* **71**(5) (1993), 108–119.

[22] J. Rayport, The Truth about Internet Business Models, in: http://www.strategy-business.com/pdf/099301.pdf (07.06.01).

[23] R. Reichwald and B. Dietel, Produktionswirtschaft, in: *Industriebetriebslehre: Entscheidungen im Industriebetrieb*, E. Heinen, ed., Gabler, Wiesbaden, 1991, pp. 395–622.

[24] C. Scheer and P. Loos, Concepts of Customer Orientation–Internet Business Model for Customer-driven Output, in: *Proceedings of the 10th European Conference on Information Systems* (*ECIS*), Information Systems and the Future of the Digital Economy, Gdansk, Poland, 2002.

[25] C. Scheer and P. Loos, Kundenschnittstelle zur Spezifikation kundenorientierter Leistungen im Internetvertrieb– Fachliche Anforderungen und informationstechnische Implikationen, in:

E-Business–Integration industrieller ERP-Architekturen, P. Loos and N. Gronau, eds, Cuvillier, Göttingen, 2002, pp. 137–151.

[26] L. Terveen and W. Hill, Beyond Recommender Systems: Helping People Help Each Other, in: *HCI In The New Millennium*, J. Carroll, ed., Addison Wesley, Reading, MA, 2001.

[27] L. Ungar and D. Foster, Clustering Methods For Collaborative Filtering, in: *Proceedings of the Workshop on Recommendation Systems*, Menlo Park, California, AAAI Press, 1998.

[28] R. Veltkamp and M. Haagedorn, State-of-the-art in Shape Matching, Technical Report, Utrecht University, Utrecht, 1999.

[29] A.K. Wright, Pattern Matching for Scheme, Report, Rice University, Houston, 1996.

[30] K. Yu, X. Xu, J. Tao, M. Ester and H.-P. Kriegel, Instance Selection Techniques for Memory-Based Collaborative Filtering, in: *Second Siam International Conference on Data Mining*, Arlington/Virginia, 2002.