Functional Design Space Representations for Lead Qualification Situations

Julian R. Eichhoff
Furtwangen University, Germany

Wolfgang Maass
Saarland University, Germany

For businesses offering complex customized solutions the capability of their sales force to engage in problem-solution discovery is a crucial success factor in selling. In this context we investigate the application of functional representations to model design spaces related to situations where a salesperson is screening for potential customers (lead qualification). Therefore we present a conceptual approach on how to cast functional representations in the domain of lead qualification. We propose computational design space representations based on probability theory that take account for the uncertainties inherent in lead qualification. And we show results from a case study in which we test the practicability of the presented approach.

Introduction

In industries that offer customized products and services, which meet their customer's individual business needs, vendors are often required to employ a consultative sales strategy called "solution selling" (cf. [1]). It comprises mainly four interdependent processes carried out on a per project basis: requirements definition, customization and integration, deployment and post-deployment support. The groundwork for these processes is laid by the vendor's sales force screening for potential customers (leads) and

Design Computing and Cognition DCC'12. J.S. Gero (ed), pp. xx-yy.

assessing their willingness and ability to buy a solution. This task is termed “lead qualification”.

Lead qualification in solution selling industries reflects a critical and well-known problem, where the customer isn’t fully aware of its needs and the vendor doesn’t exactly know what to offer. In this situation both sides jointly explore the design domain while trying to minimize the investigations needed to gain a shared understanding. However, the usual approaches to this problem are not applicable to lead qualification: Organizing a series of collaborative design workshops would be too expensive to be carried out for every potential customer. And further, requirement lists that are available at this stage are most often incomplete and ambiguous. Therefore the salesperson himself is required to take the role of a designer and estimate the fit between lead and vendor.

Well, this in turn is highly dependent on a salesperson’s individual knowledge of a lead’s needs, of products and services offered by the vendor and its partners, and of how certain bundles of products and services may be used for problem solving, we term this design knowledge. Using his design knowledge the salesperson starts a dialogue with the lead to explore the design domain. This pretty much resembles Schön’s notion of having a conversation with the design situation [2]: a cyclic process of “seeing-moving-seeing”, i.e. acquiring and interpreting information from the lead about the design situation (seeing), followed by performing changes on the conceptualized offer and making these explicit to the lead (moving), and gaining further insights on the design situation by rediscussing the lead’s feedback (seeing).

A designer constructs the required connections between problem, solution and its realization through experience. However, salespersons may lack this experience for several reasons: Especially external salespersons are not directly involved in product development at the employing vendor, and thus may have narrow insights on how their work affects downstream processes. Experiences from other salespersons may not be considered, due to limited reporting or inconsequent knowledge reuse. And limited possibilities or rigid policies for inter-organizational communication may exclude design insights from partnering organizations.

To overcome these shortcomings in intra- and inter-organizational design knowledge reuse for lead qualification, we suggest the use of a formalized design space representation, which can be accessed by services that support the conceptual design tasks carried out by a salesperson during lead qualification. In this paper we present a conceptual approach and computational representations of such design spaces that specifically address the uncertainty inherent in a salesperson’s picture of a lead. Results from a case study in the field of office fit-out projects underline the practicability of the approach but also highlight further issues.

Related Work

The literature on design research holds several approaches to represent the reasons behind design decisions made during a design process, the so-called design rationale. Existing approaches can principally be categorized in process-oriented and feature-oriented representation methods [3]. Instead of capturing the history of design processes, feature-oriented methods focus on representing designed artifacts. Functional representations are one prominent type of feature-oriented representations [4]. Yet a number of functional concept ontologies have been proposed that all share the idea of describing a design object in the context of its purpose to solve a problem or show a wanted effect [5]. Such representations seem especially appropriate for describing design spaces of lead qualification situations as they intrinsically support the idea of drawing a conceptual link between a lead's problem space and the solution space that may be offered by a vendor. Though functional concept ontologies have been applied to domains of conceptual design [6], to the best of our knowledge there exists no concise approach on how to cast functional representations in the domain of lead qualification.

Related problems to lead qualification are contractor pre-qualification and the bid/no-bid problem. Contractor pre-qualification views the problem from the lead's perspective. It deals with measuring the capabilities of potential vendors with respect to a given procurement task (cf. [7]). The bid/no-bid problem in turn is a task in organizational selling that follows lead qualification. After having assessed enough information about a lead's problem situation the vendor often needs to decide whether it is economically feasible to attend in a tender process (cf. [8]). So far there has been no approach in the field of contractor pre-qualification or the bid/no-bid problem that makes use of functional representations. However, especially the works in the field of contractor pre-qualification highlight critical representation issues, which apply to lead qualification as well: an appropriate representation should account for error-prone and subjective judgments of decision makers and it should be able to cope with noisy and uncertain data [7].

In a previous work [9] we presented a computational functional representation of the (situated) Function-Behavior-Structure (FBS) framework [10]. It is based on first-order probabilistic belief networks (cf. [11]) in order to explicitly model uncertainties and thus address the mentioned representation issues. Like other models that operationalize the FBS framework [12–17] we provided a method to describe the components of a design object, their properties, and relations among both. In contrast to previous approaches we took especially account of the fact that the certainty of a design component being associated with an attribute may vary throughout the design process and we proposed a method to highlight needed information that would most useful for reducing uncertainty.

Similar to our work the parameter analysis approach of [18] describes how to reveal the customer's problem situation (need identification and analysis) and how to generate conceptual designs grounding on these insights (parameter analysis). However, there are fundamental differences: To promote the ideation of innovative solutions their methodology suggests to maximize the set of potential solutions and thus favors need statements that are least constrained, whereas our approach focuses on finding concrete offers as fast as possible and thus favors need statements that are most effective in narrowing the set of potential solutions. To avoid the generation of need statements that are not solvable, our approach directly integrates the problem and solution parts in a single representation. This allows for "micro" design cycles, where every added constraint is directly evaluated against the current set of potential solutions. A salesperson may then discuss complicating constraints just when they arise.

The contribution of this paper is an extension to the works of [9]. In the following we give a detailed description of the conceptual relations between functional representations and lead qualification, we provide additional methods for supporting design tasks during lead qualification, and we report our findings from an ongoing case study in the domain of office fit-out projects.

Conceptual Approach

At the beginning of lead qualification the vendor has rather limited insights on the lead's problem situation, and vice versa detailed knowledge about solutions mostly resides exclusively at the vendor and its partner organizations. Thus the design space is ill-structured [19] and adequate offers cannot be directly identified. In this context a salesperson is asked to engage in problem-solution discovery, i.e. developing a clear idea of the lead's issues and formulating a battery of potential responses [20]. This is typically done in a series of sales calls, where the salesperson not only asks for problems to be solved but also proves whether the suggested solution candidates actually fit the lead's expectations. Here a suggested solution is a salesperson's individual projection of how a certain bundle of products and services will help to satisfy the lead's business needs at the time when it is implemented in the lead's organization.

However, time and resources for problem-solution discovery in lead qualification are limited. And if problem-solution discovery is carried out by a salesperson that does not have the required design knowledge lead qualification is likely to fail, because (1) the salesperson does not assess the information needed to narrow down the set of potential solutions or (2) the salesperson may draw wrong conclusions on the gathered information leading to inadequate solution candidates.

As mentioned above, our conceptual approach to support the identification of needed information and the selection of adequate candidate solutions is based on functional representations [4] that model the design space of possible solutions. However, especially interpretations of "function" vary between the proposed functional representations [5]. The discussion of [21] and [22] provides clarification. We now cast their notion in our domain.

Functional Design Space Representation

Let the world $\mathbf{W}$ be a set of variables[1] $\{W_1, W_2, \ldots\}$ that describe the aspects of a generic organization that may be affected by the products and services of a design object, and let α be a conceptual model that defines dependency relations over these variables.

Let the design object D be a set of design components $\{d_1, d_2, \ldots\}$, and let $R \subset D \times (D \cup \mathbf{W})$ be a dependency relation over the set of design components and the world, which denotes a structural relationship between two design components or between a design component and the world. Moreover, every design component d_i is associated with a set of variables $\mathbf{C}_i = \{C_{i,1}, C_{i,2}, \ldots\}$ describing properties of the design component (note same index i). A subset of these properties describe the vendor's products and services used for realizing the design component, we refer to this subset as "offer". Further, let β be a conceptual model that defines dependency relations over all design component properties.

Building on this we consider a design object as being a potential solution if it is integrated in the world, i.e. it is implemented in the lead's organization. To define how it is integrated let the mode of deployment γ be

[1] Notation: upper-case letters represent variables (e.g. X), bold upper-case letters represent sets of variables (e.g. $\mathbf{X}$), lower-case letters represent value assignments to variables (e.g. x).

a conceptual model that defines dependency relations between design component properties and world variables.

The lead's problem situation is defined as a set of constraints on the world and the design object. Let N, F_W, and U, represent sets of logical constraint functions on world variables **W**, where a constraint function evaluates to true if the value of the variable in its domain meets the constraint (otherwise it evaluates to false). F_D represents a similar set of constraint functions but over variables $\mathbf{C}_1 \cup \mathbf{C}_2 \cup \ldots$ of design component properties. N, F_W, U, and F_D differ in terms of abstraction from the design object. F_D has the lowest abstraction and specifies desired behaviors of the design object itself. Still close to the design object, F_W specifies desired effects of the design object's behavior on the behavior of the world in which it is embedded; this is also termed "function-as-effect" [21]. Finally, business needs N and usage constraints U of the lead have the highest degree of abstraction from the design object's structure. Together N, F_W, U, and F_D are used to characterize the lead's organization and to set the goals and objectives of solution development.

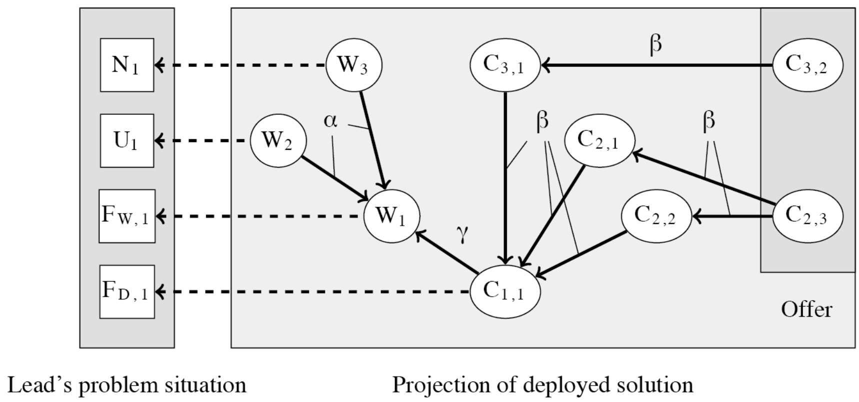

Fig. 1 Exemplary design space

Example

Consider the following example from the domain of office fit-out projects, which is also graphically depicted in Fig. 1. Office fit-out projects deal with the design and construction of scenery (interior elements such as ceilings, partitions and finishes) and settings (furniture and equipment) for office accommodation [23]. Office accommodation has several influences on the business operations of an organization (e.g. [24]). In this simplified example we describe the world **W** as follows:

- W_1: Workspace flexibility can be low or high.
- W_2: User mobility can be low or high.
- W_3: High office productivity can be mandatory or neglectable.
- Conceptual model α:
  - $W_2, W_3 \rightarrow W_1$: Workspace flexibility should be high if and only if high office productivity is a mandatory objective ($W_3 \rightarrow W_1$) and user mobility is high ($W_2 \rightarrow W_1$). In any other case workspace flexibility cannot be determined.

Design object D consists of three components: workspace d_1, furniture d_2, and partitions d_3. They stand in structural relation $R = \{(d_1, \mathbf{W}), (d_1, d_2), (d_1, d_3)\}$, i.e. the designed workspace will be implemented in the lead's organization $(d_1, \mathbf{W})$, and the workspace is made up of workstation furniture (d_1, d_2) and partition elements (d_1, d_3). The following variables and conceptual models describe the components and their dependencies:

- Workspace d_1:
  - $C_{1,1}$: Workspace layout can be territorial, semi-territorial or non-territorial.
- Furniture d_2:
  - $C_{2,1}$: Noise reduction of partitions can be low, moderate or high.
  - $C_{2,2}$: Adjustability of partitions can be low or high.
  - $C_{2,3}$: Partition offer can be none, curtains, screens or walls.
- Partitions d_3:
  - $C_{3,1}$: Adjustability of furniture can be low or high.
  - $C_{3,2}$: Furniture offer can be system A or system B.
- Conceptual model β:
  - $C_{2,3} \rightarrow C_{2,1}$: Noise reduction will be low if no partitions are included in the offer, moderate if curtains or screens are offered, and high in case of solid walls.
  - $C_{2,3} \rightarrow C_{2,2}$: Adjustability of partitions will be low if walls are offered, and high in cases of curtains, screens or no partitions.
  - $C_{3,2} \rightarrow C_{3,1}$: Adjustability of furniture will be low if system A is offered, and high in case of system B.
  - $C_{2,1}, C_{2,2}, C_{3,1} \rightarrow C_{1,1}$: Semi-territorial and non-territorial workspaces require both a high adjustability of furniture ($C_{3,1} \rightarrow C_{1,1}$) and partitions ($C_{2,2} \rightarrow C_{1,1}$) as well as a high noise reduction of partitions ($C_{2,1} \rightarrow C_{1,1}$).
- Conceptual model γ:
  - $C_{1,1} \rightarrow W_1$: When implemented in the lead's organization, workspace flexibility will be high if the workspace layout is semi-territorial or non-territorial and low in case of a territorial layout.

If we consider this design space with respect to a specific lead qualification situation where the lead's central need N_1 is high office productivity (W_3), we would constrain W_3 with $N_1(W_3)$ to be in state mandatory. Now the conceptual dependency models α, β, and γ allow us to identify constraints on a lower abstraction level and even identify candidate offers: α suggests that need N_1 will be satisfied if the external effect of the design object on the lead's organization is a high workspace flexibility (W_1) under the premise of the lead's organization having a high user mobility (W_2). However, we cannot be sure about W_1 unless we know the state of W_2. Therefore it would be important to assess the information (e.g. in the next sales call) whether user mobility (W_2) is high or low. Given that W_2 was constrained with $U_1(W_2)$ to be in state high, we can implicitly identify the constraint $F_{W,1}(W_1)$ that workspace flexibility should be high. In the other case, i.e. $U_1(W_2)$ was set to be low, workspace flexibility (W_1) is yet unknown and $F_{W,1}(W_1)$ needs to be assed explicitly. In this example we continue with workspace flexibility (W_1) being constrained to state high. Now considering the mode of deployment γ the desired "function-as-effect" ($F_{W,1}$) can be achieved with a design object that has either a semi-territorial or non-territorial workspace layout ($C_{1,1}$), which leads to constraint $F_{D,1}(C_{1,1})$. In the end β allows us to identify system B ($C_{3,2}$) in combination with curtains or screens ($C_{2,3}$) as an adequate offer, as they provide the needed adjustability ($C_{3,1}$, $C_{2,2}$) and noise reduction properties ($C_{2,1}$) for implementing semi- and non-territorial workspaces ($C_{1,1}$). See below for the complete list of identified constraint functions:

- $N_1(W_3)$ = {(mandatory, true), (neglectable, false)}
- $U_1(W_2)$ = {(high, true), (low, false)}
- $F_{W,1}(W_1)$ = {(high, true), (low, false)}
- $F_{D,1}(C_{1,1})$ = {(semi-territorial, true), (non-territorial, true), (territorial, false)}

Probabilistic Functional Design Space Representation

The exemplified reasoning tasks can also be expressed in probabilistic terms. Probability theory enables us to explicitly address the open world assumption regarding a salesperson having incomplete knowledge about the actual design space (cf. [25]). The idea is to represent the solution part of the design space (design object embedded in the world) as a probabilistic belief network and represent the problem part as constraining prior probabilities on the random variables of the belief network [9]. This provides a framework to automatically infer probability estimates for those variables that have not been constrained yet. The computed probability estimates can then be used to generate a list of candidate solutions that are most likely to solve the lead's problem. Further, metrics can be defined to determine the uncertainty of variables and their influence on other variables (herein termed "conclusiveness"). By combining both metrics we're able to identify variables that should be constrained in order to reduce overall uncertainty about the design space. A salesperson can then be advised to specifically assess more information about those variables.

α as conditional probability

high office prod. W_3	user mobility W_2	workspace flexibility W_1	$P(W_1 \mid W_2, W_3)$
mandatory	high	high	1.0
mandatory	high	low	0.0
mandatory	low	high	0.5
mandatory	low	low	0.5
neglectable	high	high	0.5
neglectable	high	low	0.5
neglectable	low	high	0.5
neglectable	low	low	0.5

Sales call 1

N_1 as prior probability (high office productivity)

W_3	$P(W_3)$
mandatory	1.0
neglectable	0.0

Estimate for W_2 (user mobility)

W_2	$P(W_2)$
high	0.5
low	0.5

Estimate for W_1 (workspace flexibility)

W_1	$P(W_1)$
high	0.75
low	0.25

Sales call 2

N_1 as prior probability (high office productivity)

W_3	$P(W_3)$
mandatory	1.0
neglectable	0.0

U_1 as prior probability (user mobility)

W_2	$P(W_2)$
high	1.0
low	0.0

Estimate for W_1 (workspace flexibility)

W_1	$P(W_1)$
high	1.0
low	0.0

Fig. 2 Probabilistic representation of reasoning steps carried out in the world part of exemplary design space depicted in Fig. 1

See Fig. 2 for an example: The upper part shows the encoding of conceptual model α as conditional probability distribution $P(W_1 \mid W_2, W_3)$. The lower part shows how it is used in combination with prior probabilities to generate estimates for yet unconstrained variables. After the first sales call high office productivity is known to be mandatory, which is reflected by the prior probability $P(W_3)$. Multiplying $P(W_3)$ with $P(W_1 \mid W_2, W_3)$ produces estimates for $P(W_2)$ and $P(W_1)$. $P(W_2)$ is now the most uncertain variable since all of its states are equally probable. Given that the conclusiveness for all three variables is equal, because the only considered influence is $P(W_1 \mid W_2, W_3)$, the salesperson would be asked to assess W_2. After the second sales call the answer for W_2 is represented as new prior probability $P(W_2)$. Multiplying $P(W_2)$ and $P(W_3)$ with $P(W_1 \mid W_2, W_3)$ now suggests that W_1 should be very likely in state high.

However, the structure of the design space needs to be configured for each lead qualification situation individually. Therefore we use first-order probabilistic models, which provide formalisms for composing probabilistic belief networks [11]. First-order probabilistic models combine probabilistic graphical models (Bayesian networks, Markov random fields, or more generally factor graphs) with a relational language (e.g. first-order logic) to represent probabilistic dependencies among attributes of multiple entities. In this case we're interested in the dependencies among design components and the world. In the following we describe how the design space model can be implemented as first-order probabilistic model.

Implementation

We represent $\mathbf{W}$ and $\mathbf{C}_1 \cup \mathbf{C}_2 \cup \ldots$ as random variables, where each possible value assignment of a variable is mapped to a probability. This probability expresses the belief of the vendor that the variable is in a certain state. The notion of parameterized random variables (par-variables) allows us to define random variables as a function of one or more logical variables, which are called the par-variable's parameters (cf. [26]). In this sense a par-variable $A(X) \in \mathbf{A}$ represents a set of normal random variables, one for each possible parameter assignment $A(x_1)$, $A(x_2)$, …, $A(x_n)$. In our model we use the parameter X to assign a variable to the world or a specific part of the design object.

To represent the dependencies of α, β, and γ we use parametric factors (par-factors) that define probability distributions over sets of random variables. A par-factor is a triple $\langle \mathbf{B}, \phi, \omega \rangle$ where $\mathbf{B} \subset \mathbf{A}$ is a set of par-variables, ϕ is a probability distribution that maps from the Cartesian product of ranges of variables in $\mathbf{B}$ to a positive real value (a probability), and ω is a set of logical terms that constrain the possible value assignments to the parameters of $\mathbf{B}$.

The used par-variables and par-factors form a parameterized belief network. It can be transformed into a normal belief network (Bayesian network) by considering every possible combination of value assignments to the used parameters (grounding). In the following we show how parameterized belief networks facilitate different inference tasks for design support (see also [10] for a discussion of the mentioned design processes).

Formulation

In the formulation process a salesperson acquires information about a lead's problem situation through discussions with the lead's representatives and possibly identifies implicit problems. Since time and possibilities for information exchange are limited the salesperson should ask for information that is yet uncertain and which allows to draw as many conclusions on the solution space as possible.

To provide a support service that highlights such important questions we first compose a design space: By defining par-factors for each type of dependency relation in α, β, and γ we can construct a design space with respect to some given component structure R:

$$\langle \{A_1(X_1), A_2(X_2), \ldots, A_n(X_n)\}, \phi, \{(X_1, X_i) \in R\} \rangle \quad (1)$$

This type of par-factor represents a dependency relation between its first par-variable $A_1(X_1)$ and some further variables $A_2(X_2)$, …, $A_n(X_n)$, where the component (or world) X_i of each variable has to stand in structural relation to the component (or world) X_1 of the first variable. The actual dependency is expressed as a conditional probability distribution ϕ that maps the possible combinations of value assignments of the par-variables to a probability. The conditionally dependent variable in ϕ is the first par-variable $A_1(X_1)$.

In addition to par-factors describing dependencies among variables we introduce par-factors representing the constraints of N, U, F_W and F_D. These provide prior probability distributions π for every par-variable in the model:

$$\langle \{A(X)\}, \pi, \emptyset \rangle \quad (2)$$

Initially π is uniformly distributed. But changing π by assigning a relatively high probability to some value of the variable would express its preference over other values. Information gathered in a sales call may now be expressed in terms of high prior probabilities for variable values that represent assessed information. Implicit information about the design space can then be determined by applying a belief propagation (BP) algorithm [28]. BP configures the probabilities of each variable in the grounded network with respect to the defined prior probabilities and dependency relations.

Having every variable configured with BP we can apply a scoring function on each variable to determine its importance for being assessed in future dialogues with the lead. In [9] we presented a measure that can be used to calculate a variable's uncertainty and conclusiveness. We define $U(\phi)$ as measure of uncertainty for a discrete probability distribution ϕ of size n:

$$U(\phi) \stackrel{\text{def}}{=} \left(\sum_{\phi} \phi \left(\log \phi - \log \tfrac{1}{n}\right)\right) / \log \tfrac{1}{n}$$

$U(\phi) \rightarrow [-1,0]$ measures the Kullback-Leibler divergence [27] of ϕ with respect to a uniform distribution of same size n. Its maximum 0 is reached if ϕ is close to the uniform distribution, and thus all variable states being equally probable. To assess a variable's uncertainty we simply calculate $U(\phi)$ for its probability estimates, which were inferred through BP.

Assessing the conclusiveness of some variable A(X) is based on its influence on other variables. A dependency relation par-factor (Eq. 1) has a high influence on its variables if its conditional probability distribution scores low on the uncertainty measure, i.e. $U(\phi)$ is close to -1. Now given all dependency par-factors involving A(X), the sum of their uncertainty scores should be minimal to have a high conclusiveness.

Combining both a variable's uncertainty and conclusiveness we're able to highlight important variables in need of further investigation by the salesperson.

Synthesis

Having formulated the lead's problem situation to a certain extent the salesperson gives consideration to candidate offers that seem capable of providing the expected "function-as-effect". Aiding the selection of a candidate offers that are likely to serve as a solution is intrinsically supported by the design space model: BP can be used to propagate gathered evidence on the problem situation towards variables describing offerable products and services. The joint probability estimate over all offer variables can then be seen as a configurational space of alternative offers, where offers that are likely to meet the given constraints are given a relatively high probability. The most likely offer is the maximum a posteriori (MAP) estimate.

Analysis & Evaluation

After choosing a candidate offer, e.g. the MAP-estimate, the salesperson should evaluate to what extent the offer fits to the given problem situation. To measure this fit we first need to decouple the parts of the design space describing the offer from the parts describing the problem situation so that we can perform inference on both parts separately. Therefore we simply duplicate the design space representation. The first copy represents the problem part, which is actually identical to the belief network used for formulation and synthesis. The second copy represents the offer. Here we remove all prior probabilities (Eq. 2) standing for constraints of N, U, F_W and F_D. Instead we change the prior probabilities of offer variables. By choosing prior probabilities of exactly 1 or 0 for offer variables we can express the selection of a certain candidate offer (clamping). Now we can apply BP on both copies separately and compare the resulting probability distributions for every pair of corresponding variables. Especially we're interested in differences of variables that can be assigned to N, U, F_W and F_D constraints.

We use a conditional probability distribution to measure the fit of such a variable pair. Two copies A(X) and A′(X) of the same variable, which has n possible values, are compared by conditioning a third random variable M with range {match, mismatch}:

$$P(M \mid A(X), A'(X)) \stackrel{\text{def}}{=} \begin{cases} 1/2n & \text{if } A(X) = A'(X) \text{ and } M = \mathit{match}; \\ \frac{1}{n-1}/2n & \text{if } A(X) \neq A'(X) \text{ and } M = \mathit{mismatch}; \\ 0 & \text{otherwise.} \end{cases}$$

Multiplying this with the probability estimates for A(X) and A′(X) gives the joint probability for "expected variable" A(X) and "offered variable" A′(X) having "matching" or "mismatching" values. We can determine such probabilities for each pair of variables separately and then compute a mean value to assess the overall fit of an offer with respect to a problem situation. The result can be used to guide the process of reformulation.

Case Study

The computational representation of the design space and the described inference techniques have been integrated in a prototype application. It realizes an information system for supporting the lead qualification process. This section presents preliminary results of an ongoing case study, in which we test this prototype with designers, consulting experts and salespersons from a major furniture manufacturer that is concerned with office fit-out projects.

Prototype Application

To define par-variables and par-factors that can be used for instantiating a design space the prototype application provides a web-based interface for knowledge engineering shown in Fig. 3.

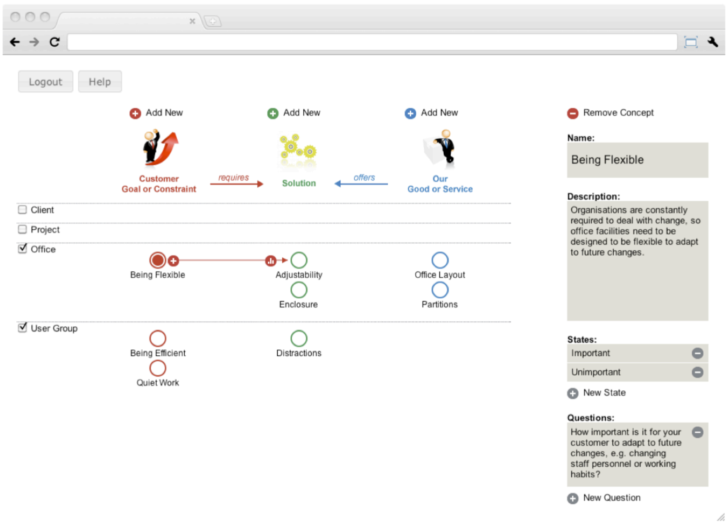

Fig. 3 Screenshot of knowledge engineering interface

Variables are defined by providing (1) a name and description of the underlying concept, (2) a rating of the concepts general importance with respect to the lead qualification process, (3) a set of questions that will guide a salesperson in assessing information about the variable, and (4) the variable's range that also frames the possible answers to the questions. Each variable is assigned to the world or a distinct type of design component. In this case the world is termed "Client" and we have three types of components: "Project", "Office" and "User Group". Further, the knowledge engineering interface discriminates between variables that are closely related to the vendor's offer ("Our Good or Service") and variables that are affected by abstract constraints ("Customer Goal or Constraint"). All other variables reside under the category "Solution".

Having described a set of variables in this way a knowledge engineer may define dependency relations among those variables. By specifying which variables are affected by the dependency relation the knowledge engineering interface generates a tabular representation of all possible value combinations as shown in Fig. 4. Each combination is phrased as a conditional statement and assigned with a rating that represents its probability. Initially set to “I don’t know” (0.5) these ratings can be adjusted by the knowledge engineer. This is done by choosing a statement as being “always true” (1), “often true” (0.75), “seldom true” (0.25) or “never true” (0). Leaving the rating at 0.5 represents a neutral state, i.e. either the statement may be as often true as it is false or the knowledge engineer is unable to determine. However, these ratings cannot be directly used as probabilities of a par-factor’s probability distribution. First they need to be normalized so that all probabilities sum to 1, where in the case of all ratings being 0 the probabilities are reset to be uniformly distributed.

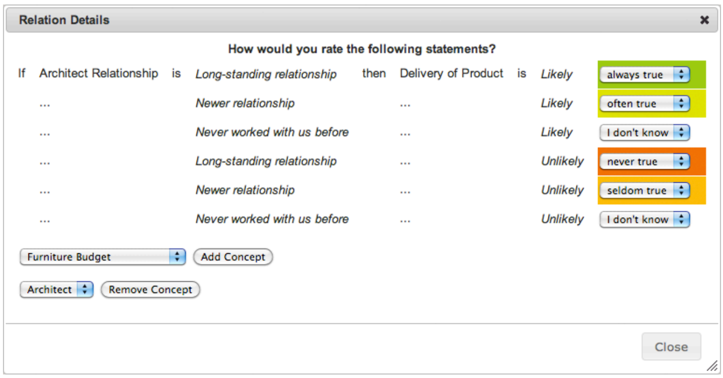

Fig. 4 Screenshot of dependency relation definition

In total the furniture manufacturer’s experts defined a set of 33 par-variables and 44 par-factors. These provide the basis for the second part of the prototype application, which resembles a dynamic sales questionnaire. This sales questionnaire (cf. Fig. 5) is also web-based and can be used by a salesperson in preparation and follow-up to a sales call or even during sales calls via a tablet computer.

The central part of the questionnaire is a list of questions, which is generated from those questions defined by the knowledge engineers. Its sortation is dynamically determined by using the scoring function for assisting the formulation process. To setup the necessary design space a salesperson simply defines the projects associated to a client, the offices that are subject to these projects, and the user groups, which are planned to work in the offices. When giving answers to the questionnaire the prior probabilities of associated variables are adjusted and the list is automatically resorted. Questions of variables with a widespread influence that haven't been answered explicitly or implicitly, and which have been defined as being important by the knowledge engineers are placed on the top. This sortation is intended to guide the salesperson's priority in assessing information from the lead.

Further, the techniques described for synthesis, analysis and evaluation are used to generate ratios that are of special interest to a salesperson during lead qualification. First, to give the salesperson an orientation about the progress of lead qualification the mean of all computed uncertainty scores is used to indicate the current knowledge about the lead. Second, probability estimates of certain variables are presented that help to determine opportunities for business, i.e. the potential for providing consultation services or selling furniture to the lead. Third, the mean of all "match"-probabilities is used to show a simplified overall score for the fit of the most probable offer (MAP-estimate) with respect to the problem situation.

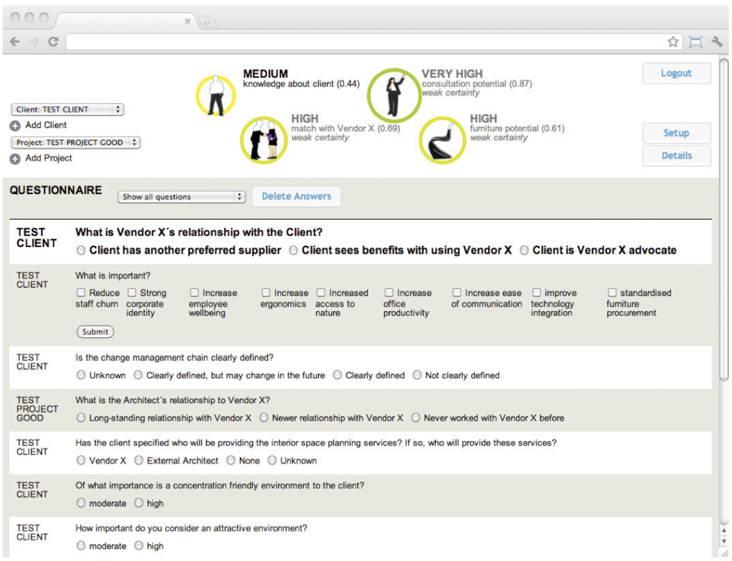

Fig. 5 Screenshot of sales questionnaire

Preliminary Results

Main parts of the knowledge engineering have been conducted in a three-day workshop with an interior-design architect, a representative from the sales training department, and a representative from the project-consulting department. After an introduction to the usage of the knowledge engineering interface, participants were asked to use the system to model the concepts being most relevant to lead qualification situations. Thereby participants took the role of knowledge expert and knowledge engineer at the same time. The parts that should be described were predefined as “Client”, “Project”, “Office” and “User Group”. Since there was no predetermined process that would have guided this task, the participants had to find a strategy on how to develop the knowledge base from scratch and how to do this cooperatively. The participants decided to proceed inductively by using an example project of the interior-design architect. Following several discussions about the details, the participants generalized important issues of this case under the architect’s lead to defined variables of the knowledge model. After a period of using the system jointly to enter one variable at a time and discussing its semantics, participants decided to work in parallel. Therefore every predefined component “Client”, “Project”, “Office” and “User Group” was assigned to a certain participant whose responsibility was to model the variables and dependency relations within that component. Afterwards the participants gathered again to define dependency relations between the components jointly.

During this process several obstacles occurred, which are now subject for our ongoing refinement of the prototype:

First, in some cases the participants found it hard to assign a variable to the given categories. Some variables could be assigned to one or another category depending on the point of view. E.g. storage capabilities may be defined for each user group individually or for an office as a whole. In this case the knowledge engineers decided to model storage capabilities on the lowest detail level (user group) to allow a greater flexibility at the cost of a salesperson having to answer more questions.

Second, dependency relations used in the current state of the prototype are limited to multinomial probability tables. However, when using multinomial distributions in a Bayesian network for encoding dependency relations, parents of an influenced variable affect each other (explaining away phenomenon [28]). In some cases this is not desirable. E.g. probability estimates for furniture potential and consultation potential are derived from different variables of the whole design space. Now when some variable (e.g. providing storage) has a positive influence on furniture potential this should not automatically affect another semantically unrelated parent (e.g. relationship to architect). In this case each parent should contribute independently to furniture potential. Thus knowledge engineers should be able to choose from different types of dependency relations, like causally independent noisy-OR, noisy-MAX or noisy-addition distributions [29].

Third, in close relation to issue two, knowledge engineers experienced unwanted side effects in the reaction of the dynamic questionnaire while modeling the knowledge base. To make these effects comprehendible it was suggested to implement a mechanism that explains the inference results produced by the questionnaire. It should offer the possibility to easily retrace the causes for a specific suggestion of the questionnaire.

Conclusion and Future Work

Lead qualification in solution selling industries marks the very beginning of a design process that is initiated by discussions of a salesperson with representatives of a lead. Depending on the results of lead qualification this design process may or may not be carried on by design experts in a successive project. Thus, supporting the process in its initial phase is an important issue that underlines the need for representing design spaces in lead qualification situations.

Though functional representations were originally developed to model devices, products, objects, and processes based on their functionalities [5], this approach can also be applied to model problem-solution discovery in lead qualification situations. As demonstrated these models can help to bridge the gap between abstract business needs of a lead and the potential product and service bundles offered by a vendor and its partner network.

Throughout lead qualification salespersons try to iteratively structure a design space that is only vaguely defined in the beginning. To support this process a design space model should explicitly account for the representation of uncertainties. We showed that first-order probabilistic models are suitable for its implementation. The preliminary results from a case study suggest that the proposed representation can be practically used to formalize design knowledge in form of a knowledge base. This in turn can be used by inference mechanisms during lead qualification to simulate and assist formulation, synthesis and evaluation tasks.

However, the generation the proposed first-order probabilistic representation requires a substantial knowledge engineering effort. In our future work we'll refine the prototype application to simplify the knowledge acquisition process. Therefore we seek to realize an explanation component that helps knowledge engineers and salespersons to understand the inference results. Further, we'll implement machine-learning capabilities that make use of the answers given to the questionnaire in order to inductively learn new dependency relations. And we seek to generalize the definition of dependency relations to allow the specification of individual types of conditional probability distributions.

Acknowledgement

This work was partially funded by the German Federal Ministry for Education and Research (BMBF, contract 17N0409). The authors would like to thank Sabine Janzen, Simone Orth, Andreas Filler and Tobias Kowatsch for valuable discussions and an anonymous reviewer for helpful comments on earlier versions of this paper.

References

1. Tuli, K.R., Kohli, A.K., Bharadwaj, S.G. (2007) Rethinking Customer Solutions: From Product Bundles to Relational Processes. J. Market. 71(3): 1–17
2. Schön, D.A. (1992) Designing as Reflective Conversation with the Materials of a Design Situation. Res. Eng. Des. 3(3): 131–147
3. Regli, W.C., Hu, X., Atwood, M., Sun, W. (2000) A Survey of Design Rationale Systems: Approaches, Representation, Capture and Retrieval. Eng. Comput. 16(3): 209–235
4. Chandrasekaran, B., Goel, A., Iwasaki, Y. (1993) Functional Representation as Design Rationale. Computer 26(1): 48–56
5. Erden, M.S., Komotoa, H., van Beeka, T.J., D'Amelioa, V., Echavarriaa, E., Tomiyama, T. (2008) A review of function modeling: Approaches and applications. AI EDAM 22(2): 147–169
6. Wang, L., Shen, W., Xie, H., Neelamkavil, J., Pardasani, A. (2002) Collaborative conceptual design—state of the art and future trends. Comput. Aided Design 34(13): 981–996
7. El-Sawalhi, N., Eaton, D., Rustom, R. (2007) Contractor pre-qualification model: State-of-the-art. Int. J. Proj. Manag. 25(5): 465–474
8. Mohemad, R., Hamdan, A.R., Ali Othman, Z., Mohamad Noor, N.M. (2010) Decision Support Systems (DSS) in Construction Tendering Processes. IJCSI 7 (2): 35–45
9. Eichhoff, J.R., Maass, W. (2011) Representation and Reuse of Design Knowledge: An Application for Sales Call Support. In: König, A., Dengel, A., Hinkelmann, K., Kise, K., Howlett, R.J., Jain, L.C. (eds.) KES 2011, Part I. LNAI, vol. 6881. Springer, Heidelberg: 387–396
10. Gero, J.S., Kannengiesser, U. (2004) The situated function-behaviour-structure framework. Des. Stud. 25(4): 373–391
11. De Salvo Braz, R., Amir, E., Roth, D. (2008) A Survey of First-Order Probabilistic Models. In: Holmes, D.E., Jain, L.C. (eds.) Innovations in Bayesian Networks, Studies in Computational Intelligence, vol. 156. Springer, Heidelberg: 289–317
12. Christophe, F., Bernard, A., Coatanéa, É. (2010) RFBS: A model for knowledge representation of conceptual design. CIRP Ann.-Manuf. Techn. 59(1): 155–158

13. Goel, A.K., Rugaber, S., Vattam, S. (2009) Structure, behavior, and function of complex systems: The structure, behavior, and function modeling language. AI EDAM 23(1): 23–35
14. Gorti, S.R., Gupta, A., Kim, G.J., Sriram, R.D., Wong, A. (1998) An object-oriented representation for product and design processes. Comput. Aided Design 30(7): 489–501
15. Labrousse, M., Bernard, A. (2008) FBS-PPRE, an Enterprise Knowledge Lifecycle Model. In: Bernard, A., Tichkiewitch, S. (eds.) Methods and Tools for Effective Knowledge Life-Cycle-Management. Springer, Heidelberg: 285–305
16. Szykman, S., Sriram, R.D., Bochenek, C., Racz, J.W., Senfaute, J. (2000) Design Repositories: Next-Generation Engineering Design Databases. IEEE Intell. Syst. App. 15(3): 48–55
17. Xue, D., Xu, Y. (2003) Web-based distributed system and database modeling for concurrent design. Computer-Aided Design 35(5): 433–452
18. Kroll, E., Condoor, S.S., Jansson, D.G. (2001) Innovative Conceptual Design: Theory and Application of Parameter Analysis. Cambridge University Press, Cambridge.
19. Zdrahal, Z. (2007) The Role of Knowledge in Design Problems. In: Wagner, R., Revell, N., Pernul, G. (eds.) DEXA 2007. LNCS, vol. 4653. Springer, Heidelberg: 884–894
20. Bonney, F.L., Williams, B.C. (2009) From products to solutions: the role of salesperson opportunity recognition. Eur. J. Marketing 43(7/8): 1032–1052
21. Chandrasekaran, B., Josephson, J.R. (2000) Function in Device Representation. Eng.Comput. 16(3): 162–177
22. Chandrasekaran, B. (2005) Representing function: Relating functional representation and functional modeling research streams. AI EDAM 19(2): 65–74
23. Baccarini, D., Bateup, G. (2008) Benefits management in office fit-out projects. Facilities 26(7/8): 310–320
24. Peponis, J., Bafna, S., Bajaj, R., Bromberg, J., Congdon, C., Rashid, M., Warmels, S., Zhang, Y., Zimring, C. (2007) Designing space to support knowledge work. Environ. Behav. 39(6): 815–840
25. Nikolaidis, E., Mourelatos, Z.P., Pandey, V. (2011) Design Decisions under Uncertainty with Limited Information. CRC Press/Balkema, Leiden
26. Kisyński, J., Poole, D. (2009) Constraint processing in lifted probabilistic inference. In: Proc. of the Twenty-Fifth Conf. on Uncertainty in Artificial Intelligence (UAI '09): 293–302
27. Kullback, S., Leibler, R.A. (1951) On Information and Sufficiency. Ann. Math. Stat. 22(1): 79–86
28. Pearl, J. (1988) Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, San Francisco
29. Heckerman, D., Breese, J.S. (1994) Causal Independence for Probability Assessment and Inference Using Bayesian Networks. IEEE T. Syst. Man. Cyb. 26(6): 826–831