Distributed Imprecise Design Knowledge on the Semantic Web

Julian R. Eichhoff[1] and Wolfgang Maass[2]

[1] Furtwangen University, Robert-Gerwig-Platz 1, 78120 Furtwangen, Germany
julian.eichhoff@hs-furtwangen.de
[2] Saarland University, P.O. 15 11 50, 66041 Saarbrücken, Germany
wolfgang.maass@iss.uni-saarland.de

Abstract. In this paper we outline a shared knowledge representation based on RDF. It can be used in a distributed multi-tenant environment to store design knowledge. These RDF-graphs incorporate all necessary information to instantiate Bayesian network representations of certain problem solving cases, which are used to support the conceptual design tasks carried out by a salesperson during lead qualification.

1 Introduction

In industries that offer customized goods and services, which meet their customer's individual business needs, vendors are often required to employ a consultative sales strategy called "solution selling" (cf. [7]). 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 assessing their willingness and ability to buy a solution. This task is termed "lead qualification". Lead qualification in solution selling industries is highly dependent on a salesperson's individual knowledge of a lead's (problem) situation, of goods and services offered by the vendor and its partners, and of how certain bundles of goods and services may be used for problem solving; we term this design knowledge. But 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, we've implemented a shared design knowledge repository based on the Function-Behavior-Structure (FBS) framework [4] and use it for services which support the design activities during lead qualification.

Xue and Xu [8] suggest a web-accessible distributed database to store design knowledge based on the FBS notation. Like other models that operationalize the FBS framework [2, 6], they follow an entity-relationship approach. However, it would require a significant knowledge engineering effort to continuously maintain a design model that builds on a highly detailed and formal knowledge representation (KR), where innovative yet uncertain design beliefs may be left out. Probability theory can provide an adequate framework to model uncertainties in design decisions [5]. Encouraging approaches that represent probabilistic belief networks by means of semantic models exist in other domains [9, 10]. But to the author's knowledge, there exist no Semantic Web representations of the FBS framework that incorporate uncertainty information, and can be managed in a multi-tenant environment. In [3] we defined a Bayesian Network (BN) representation of the FBS model, termed FBS-BN, which encodes design knowledge as probability tables. In the following we outline its shared storage in a distributed RDF-Store.

2 Design Knowledge Representation and Storage

A FBS-BN represents a configurational design space for a specific problem-solving situation in form of a Bayesian Network. Discrete random variables are used to describe possible design object configurations in light of the customer's demands. Every variable is associated with a certain component of the design object to serve as characterizing attribute. There are three different variable types: Function variables (F) represent the purpose for which a solution is designed for, i.e. goals and constraints of the customer. Structure variables (S) represent possible offerings, i.e. product and service bundles that can be provided by the vendor. Behaviors are mediating concepts between Functions and Structures representing the actual solution, i.e. how products and services are meant to achieve goals and fulfill constraints. There are three subtypes of Behavior variables: Be variables describe the solution as expected by the customer; their value is derived from Function variables. Bs variables represent the solution as offered by the vendor; their value is derived from Structure variables. And Bc variables are used for comparing the match of Be and Bs. The design knowledge about how Functions, Behaviors and Structures affect each other is encoded in form of conditional probability distributions (CPDs). These CPDs represent a set of propositions of the form "if concept X is in state x then another concept Y is (or should be) in state y". The associated probabilities express the degree of belief that a proposition holds. Possible relations are $F \rightarrow Be$ (Function expects Behavior), $S \rightarrow Bs$ (Structure exhibits Behavior), and implications within a variable group ($F \rightarrow F$, $Be \rightarrow Be$, $Bs \rightarrow Bs$, and $S \rightarrow S$).

To support the assessment of information in lead qualification, a support service should highlight those concepts that are yet uncertain and thus need further investigation. Therefore we generate a case-specific FBS-BN to characterize the current problem-solving situation. Changes in a node's prior can be used to represent explicit design decisions (evidence), i.e. assigning a relatively high probability to a state would express its preference over other states. Implicit design decisions are then given by Bayesian inference in form of probability estimates for the hidden nodes. Building on this, we highlight (yet) uncertain concepts by rating every hidden node with an uncertainty measure (e.g. [3]).

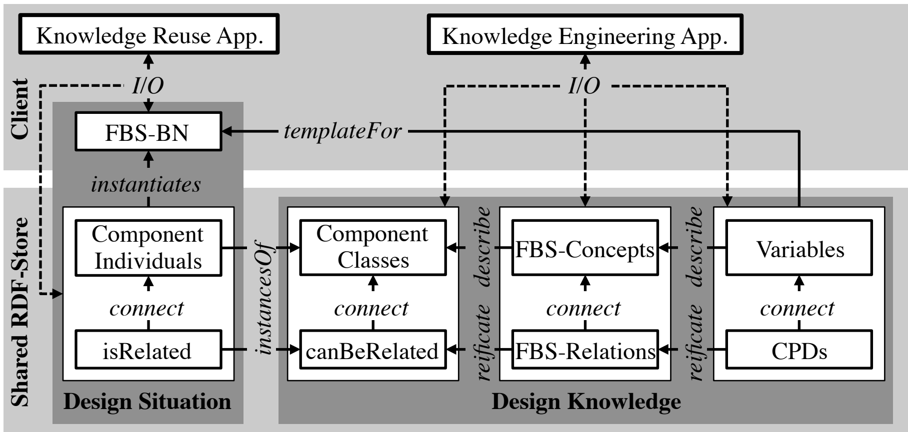

Fig. 1. Architecture of Proposed Distributed Knowledge Based System

For using FBS-BN representations across different participating organizations, we employ a RDF-based shared design KR to store the needed variable, CPD, and design object component definitions. All applications interoperate via this KR. Principally we consider two types of client-side application roles, namely knowledge engineering and knowledge reuse applications. While knowledge engineering applications provide an interface to manage the KR, knowledge reuse applications use it to support designing tasks in problem solving situations (cf. Fig. 1). The KR is stored in a distributed RDF-store is based on S3DB [1]. S3DB provides a sophisticated framework for graph-based permission management. Rather than using coarse all or nothing policies, S3DB allows an organization, department or individual to share certain parts of their design knowledge with designated users. Moreover, S3DB offers a meta-model for cooperatively defining TBoxes and ABoxes for RDF-graphs.

To facilitate the hierarchical formalization of design knowledge on different levels of complexity we employ a formalism for iterative reification: We start from a simple relational model for design object component classes and their individuals. Component classes can be linked with "canBeRelatedTo" relations to denote that they are dependent "somehow". These relations then frame possibilities for "isRelatedTo" relations on instance layer. The first step in clarifying these yet anonymous relations is done by providing FBS-concepts as characterizing attributes for component classes and connect them via expects, exhibits and implicates relations (FBS-relations). These associations determine how the design object components are actually interrelated with each other. In the second step, FBS-concepts are operationalized as discrete variables by specifying a set of possible variable states (or attribute values), which results in a description of the attribute network as configurational variable space. Building on these variables we reificate attribute associations, i.e. we provide a detailed explication of expects, exhibits and implicates relations in form of conditional probability tables. Lastly variable and CPD definitions can be used as templates for FBS-BN instantiation.

3 Conclusion and Future Work

We have outlined a Semantic Web KR of the FBS framework based on RDF, which can be managed in a distributed multi-tenant environment. It is used to employ FBS-BN-based uncertainty reasoning for lead qualification support. Currently we are implementing two prototype applications, and look forward to test their impact on lead qualification performance empirically.

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, Andreas Filler and Tobias Kowatsch for valuable discussions.

References

1. Almeida, J.S., Deus, H.F., Maass, W.: S3DB core: a framework for RDF generation and management in bioinformatics infrastructures. BMC Bioinformatics 11(387), pp. 1-10 (2010)
2. Christophe, F.,Bernard, A., Coatanéa, É.: RFBS: A model for knowledge representation of conceptual design. CIRP Ann.-Manuf. Techn. 59(1), pp. 155–158 (2010)
3. Eichhoff, J.R., Maass, W.: 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, pp. 387–396. Springer, Heidelberg (2005)
4. Gero, J.S., Kannengiesser, U.: The situated function-behaviour-structure framework. Des. Stud. 25(4), pp. 373–391 (2004)
5. Nikolaidis, E., Mourelatos, Z.P., Pandey, V.: Design Decisions under Uncertainty with Limited Information. CRC Press/Balkema, Leiden (2011)
6. Szykman, S., Sriram, R.D., Bochenek, C., Racz, J.W., Senfaute, J.: Design Repositories: Next-Generation Engineering Design Databases. IEEE Intell. Syst. App. 15(3), pp. 48–55 (2000)
7. Tuli, K.R., Kohli, A.K., Bharadwaj, S.G.: Rethinking Customer Solutions: From Product Bundles to Relational Processes. J. Market. 71(3), pp. 1–17 (2007)
8. Xue, D., Xu, Y.: Web-based distributed system and database modeling for concurrent design. Computer-Aided Design 35(5), pp. 433–452 (2003)
9. Zhanga, W.Y., Caib, M., Qiuc, J., Yinb, J.W.: Managing distributed manufacturing knowledge through multi-perspective modelling for semantic web applications. Int. J. of Prod. Res. 47(23), pp. 6525–6542 (2009)
10. Zheng, H.-T., B.-Y. Kang, Kim, H.-G.: An ontology-based bayesian network approach for representing uncertainty in clinical practice guidelines. In: Goebel, R., Siekmann, J., Wahlster, W. (eds.) URSW 2005-2007. LNAI, vol. 5327, pp. 161–173. Springer, Heidelberg (2008)