Ontologies and Scenarios as Knowledge Representations for Use in Design Research

Andrey Soares
College of Information Sciences and Technology
Pennsylvania State University
University Park, PA - 16802
asoares@ist.psu.edu

ABSTRACT

This position paper discusses and suggests directions for doing research in conceptual modeling and knowledge representation within Design Research. It is widely known that one of the reasons for information systems to fail is the misunderstanding and miscommunication between users and developers. Considering that users are usually the domain experts, their knowledge about the domain should be captured and represented in a more effective way for using it throughout the systems development phases. The paper emphasizes the need for better approaches for representing knowledge in information systems, and suggests an alternative approach based on ontologies and scenario-based design.

Author Keywords

Conceptual Modeling, Knowledge Representation, Scenarios-Based Design, Ontologies

ACM Classification Keywords

D.2.10 Software Engineering: Design Tools and Techniques

INTRODUCTION

It is widely known that an impressive high rate of 50%-80% of information systems (IS) usually fail [5,10,12]. Along with some reasons for such failures are the misunderstandings and the miscommunication between users and developers. For decades, researchers have been looking for and suggesting solutions to this issue. However, it still presents some issues and challenges. The workshop aims to address issues related to these areas, and to identify new research activities.

This position paper introduces a potential discussion topic, shows some issues related to the topic, where I stand within it, how my work addresses them, and how the topic can contribute to advance the Design Research.

DISCUSSION TOPIC AND ISSUES

I would like to introduce the topic "Ontologies and Scenarios as knowledge representations for use in design research" for further discussion with the participants of the workshop. Independently of the development phase one is working on, one will certainly need some information and knowledge about devising representations in a given domain. Understanding and being able to share accurate information (concepts, relationships, etc.) about the system is imperative for its success. This set of concepts and relationships constitutes the conceptual model of the system [12], which can be seen as "an idealized view of the how the system works" (p.26). In this sense, I agree and support Johnson & Henderson's view that a conceptual model should be the backbone of an information system [12]. That is, a design should initiate with a conceptual model, and should rely on it throughout all system development phases. Thereby, a conceptual model would make available, to the designers, a repository of concepts and its relationships (glossary), a description of the system in use (scenarios), a translation to the user-interfaces, a refinement of objects used in the implementation, a documentation of the system, and an inner organization point for managing the design [12].

In most cases, a conceptual model is used to describe only the information that will be implemented in some IS design artifact (i.e. database, programming language, etc.), and not the information related to the domain under study. In addition, the conceptual model will be represented in a way that is suitable for the artifact in use. Consequently, this approach makes it difficult to keep up with a common language for integrating domain experts (users) into the design process.

Alternatively, Scenario-Based Design has become an approach that makes use of scenarios to represent the knowledge embedded into the system, and how the system works, as well as to encourage user's participation in design [1,2,7,9,14]. Moreover, scenarios are frequently represented as a textual narrative, which makes it explicitly understood by all stakeholders involved in the project. Achour [1] has shown that the use of scenarios has grown significantly, and that 80% of the scenario-based projects analyzed in her study have employed textual representation.

Although, conceptual modeling and scenarios have been presented as alternative solution for representing knowledge in IS, these approaches have also uncovered some important issues and challenges for Design Research. As the IS design grows or increases its granularity of detail, both approaches will suffer with constant updates [7,9,13,14]. Carroll [3] suggests that "scenarios can be written at multiple levels, from many perspectives, and for many purposes" (p.2). The complexity encountered for keeping a large and detailed system coherent and up to date has lead many project managers to suspend its use, especially for problems with time and resources consumption [9], as well as traceability [14].

The textual narrative is another issue; whereas it can allow scenarios to become easily understood by the stakeholders, it would not permit easily resourceful additions or multiple views of the system [13]. For example, if a developer wants to see the system with a perspective of specific stakeholders (i.e. users, buyers, managers, etc.) the narrative text would make it difficult to present such breakdowns [14].

Weidenhaupt et al. [14] have also raised some issues regarding the use of scenarios that would lead to a valuable discussion during the workshop: partial views, distributed scenario development, quality assurance, traceability, reuse of scenarios, and evolving scenarios.

POSITION AND MY WORK

My work aims to develop methodologies that would enable using ontologies for representing knowledge in information systems, and for driving a variety of activities within the information systems development phases. For example, after modeling a domain, the ontologies created should be able to provide information for user-interface or database generation. Chandrasekaran et al. [4] define ontologies as "content theories about the sorts of objects, properties of objects, and relations between objects that are possible in a specified domain of knowledge" (p.20).

An ontology shares almost the same elements (concepts, relationships, properties, etc.) of a conceptual modeling, as presented in Johnson & Henderson [12]. Mapping and representing the knowledge embedded into the system in a ontology can produce several benefits. First, an ontology can provide a shared vocabulary for the system, so all stakeholders involved in the project would be able to have a common understanding of the system. Second, due its constructs, an ontology can store both structured and non-structured text formats. For instance, it could help ontologies to keep compatibility with previous text narrative scenarios, while it improves and enhances the semantics used to describe things. Finally, because ontologies have constructs for representing properties and relationships, the objects can be better represented, and that information can be used for different phases of development, such as data modeling, activity diagrams, interfaces, etc.

According to Johnson & Henderson [12], a "conceptual model is the single most important part of your design" (p.32). Thus, everything else in an information system should be "grounded in it, feeding it further… building on it" (p.32). Like a conceptual model, ontologies can be equally important part of your design, and also the central point of an IS. Wand & Weber [15], emphasize that "theories of ontology provide us with an artifice for describing a perceived world. Our description will only be as good as our ontologies. Accordingly, our information systems will only be as good as our ontologies." (p. vi). Guarino [8] argues that "every information system has its own ontology…" (p. 3), and suggests a framework for using ontologies, so-called Ontology-Driven Information Systems (ODIS), at development time and at run time.

From a research perspective, Fonseca [6] suggests that the use of ontologies in information systems can be divided into theory of ontology, which can help to build a better conceptual model, and computational ontologies, which can provide a better representation and use of the conceptual model. In addition, Irwin & Turk [11] shows that ontologies for use case (and scenarios) modeling need some work, especially because they lack of proper constructs for representing sequence of events, which is one of the main reasons why someone would use scenarios as a design artifact.

CONCLUSION

This position paper introduced a new topic "Knowledge Representation" for the discussion agenda of the DIS2006 workshop. The issues and challenges presented can lead further research in Design. The expected outcomes of this research, the artifacts, comprise with the fundamental principle of design-science paradigm [10] from which "knowledge and understanding of a problem domain and its solution are achieved in the building and application of the designed artifact" (p. 82).

REFERENCES

1. Achour, C.B., Writing and Correcting Textual Scenarios for System design. in *Proceedings of the Natural Language and Information Systems - DEXA Workshop, Vienna, Austria,* (1998), 166-170.
2. Carroll, J.M., Five Reasons for Scenario-Based Design. *Interacting with Computer*, *13* (1), (2000), 43-60.
3. Carroll, J.M., Scenarios and Design Congnition. in *Proceedings of the IEEE Joint International Conference on Requirements Engineering,* (2002), 3-5.
4. Chandrasekaran, B., Joseph, J.R. and Benjamins, V.R., What are ontologies, and why do we need them?, IEEE Intelligent Systems, 14 (1), (1999), 20-26.

5. Dorsey, P., Top 10 Reasons Why Systems Projects Fail. (2000). From: http://www.dulcian.com

6. Fonseca, F., The Double Role of Ontologies in Information Systems Analysis and Design, College of Information Sciences and Technology, Pennsylvania State University, University Park, PA, (2006).

7. Go, K. and Carroll, J.M. The Blind Men and the Elephant: Views of Scenario-Based System Design. Interactions, 11 (6), (2004), 44-53.

8. Guarino, N., Formal Ontology in Information Systems. in FOIS'98, Trento, Italy, (1998), Amsterdam, IOS Press, 3-15.

9. Hertzum, M. Making Use of Scenarios: A Field Study of Conceptual Design. *International Journal of human-Computer Studies*, *58* (2), (2003), 215-239.

10. Hevner, A.R., March, S.T. and Park, J. Design Science in Information Systems Research. *MIS Quartely*, *28* (1), (2004), 75-105.

11. Irwin, G. and Turk, D., An Ontological Analysis of Use Case Modeling Grammar. Journal of the Association for Information Systems, 6 (1), (2005), 1-36.

12. Johson, J. and Henderson, A. Conceptual Models: Begin by Designing What to Design. *Interactions*, *9* (1), (2002), 25-32.

13. Rosson, M.B. and Carroll, J.M. Scenario-Based Design. in Jacko, J. and Sears, A. (eds.). *The Human-Computer Interaction handbook: Fundamentals, Evolving Technologies and Emerging Applications*, Lawrence Erlbaum Associates, (2002), 1032-1050.

14. Weidenhaupt, K., Pohl, K., Jarke, M. and Haumer, P. Scenarios in System Development: Current Practice. *IEEE Software*, *15* (2), (1998), 34-45.

15. Wand, Y. and Weber, R., Reflection: Ontology in Information Systems - Foreword. *Journal of Database Management*, 15 (2), (2004), iii-vi.