ONTOLOGY-DRIVEN INFORMATION SYSTEMS: AT DEVELOPMENT TIME

Andrey Soares[a] and Frederico Fonseca[b]

[a]College of Information Sciences and Technology, Pennsylvania State University
University Park, PA, USA
asoares@ist.psu.edu

[b]College of Information Sciences and Technology, Pennsylvania State University
University Park, PA, USA
fredfonseca@ist.psu.edu

ABSTRACT

This paper introduces a framework for creating and using ontologies throughout the conceptual modeling phase of Information Systems. The research contends that ontology performs a central role in IS, and is neither a substitute nor a competitor of existing conceptual schemas, but a significant and complementary resource that can provide formalized knowledge about the universe of discourse of an information system. In this research, we explore the creation of ontologies that are suitable for representing knowledge of a given IS domain, and the use of these ontologies to help the creation of the IS components (i.e. application programs, information resources, and user interfaces). The research methodology follows the Design Science Paradigm, as it values all steps in the process of developing our framework, and produces a body of knowledge that is suitable for both theory and practice.

Keywords: Ontology, Information Systems, Information Systems Analysis and Design, Design Science

INTRODUCTION

Guarino's (1998) concept of Ontology-Driven Information Systems (henceforth, ODIS) included the use of ontologies in two distinct phases: first, at development time, and second, at run time of Information Systems (IS). Nevertheless, little was done in terms of developing "a comprehensive framework for dealing with ontologies, so that programmers can start right away with the development of ontology-driven IS" (Yildiz and Miksch, 2007). The increasing use of ontologies in computational environments (e.g. Computer Science, Information Systems, Bioinformatics, etc.) and the lack of theoretical foundations for creating ontologies brought the attention of researchers to the development of theories of ontology in IS. Despite of the common sense that ontology plays a central role in Information Systems Analysis and Design (ISAD), researchers have not yet produced suitable guidelines for using ontology in ISAD. This research focuses on the development of an architecture for Ontology-Driven Information Systems at development time (henceforth, ODIS-Dev), where ontology is the backbone of conceptual modeling activities. We investigate the construction of ontologies of a given domain, and their use to support the creation of conceptual schemas and other IS artifacts during development time.

Our notion of ontology follows Studer et al. (1998), who define an ontology as "a formal, explicit specification of a shared conceptualization. A 'Conceptualization' refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that

phenomenon. 'Explicit' means that the type of concepts used and the constraints on their use are explicitly defined... 'Formal' refers to the fact that the ontology should be machine-readable, which excludes natural language. 'Shared' reflects the notion that an ontology captures consensual knowledge, that is, it is not private of some individual, but accepted by a group" (p.25). In this sense, ontology can be seen as an engineering artifact composed of concepts, relationships and axioms that are used to describe facts (accepted by a community) of a giving domain (Guarino, 1998).

Furthermore, ontologies **of** IS describe the basic concepts of the conceptual modeling upon which an ontology is created. These ontologies are intended to guide the creation of domain-specific ontologies for its use in IS. Examples of ontologies explored not only for domain modeling but also for IS modeling include: OntoClean-DOLCE (Gangemi, et al., 2003), BWW-Bunge-Wand-Weber (Wand, et al., 1995), and UFO-Unified Foundational Ontology (Guizzardi and Wagner, 2004). Second, ontologies **for** IS describe ontologies that represent the knowledge of a given domain (e.g. pharmacy), that can provide support for the creation of the IS components.

We see an information system as having its own ontologies that are usually applied across different domains. Wand and Weber (1989) argue that "we model two different domains in systems analysis and design: reality and the information systems" (p.82). Therefore, we argue that ontology of IS is necessary in order to support this interoperability and mapping between domains, as well as to support the creation of IS components.

The aim of this research is three fold. First, we look what kind of ontologies of IS have the constructs to represent the knowledge needed for modeling IS. Next, we suggest a methodology for creating ontologies for IS that are suitable for ISAD. Finally, we investigate how the two kinds of ontologies can be used to create or to support the creation of IS components (Guarino, 1998). To achieve these goals, we will address the following research question (RQ) and related sub-questions:

RQ: *How can we use ontologies to drive Information Systems Analysis and Design?*

RQ1: Which are the proper constructs that an ontology of IS has to have in order to support and represent the knowledge suitable for IS modeling?

RQ2: How can we use an ontology of IS to create a domain ontology for IS?

RQ3: How can ontology for IS and ontology of IS be used to create specific IS Components?

The contribution of this project will support research in the areas of ontology engineering, knowledge acquisition, conceptual modeling, and Information Systems Analysis and Design. In this paper:

- we present the underlying background that supports the research in the uses of ontology in Ontology-Driven Information Systems (Section 0).
- we establish a significant problem of conceptual modeling concerning the use of ontologies during development time of IS, and we present a two-phase framework to illustrate how ontologies can drive Information Systems Analysis and Design (Section 0).
- we present a research methodology, based on Design Sciences paradigm, which will produce artifacts for IS practitioners (Section 0).
- we highlight the outcomes of this research, and we conclude the paper (Section 0).

Background

Guarino (1998) coined the term Ontology-Driven Information Systems for systems that make use of formally defined ontologies. According to him an explicit ontology plays a central role in this kind of system thus driving all of its aspects and components. Guarino distinguishes two orthogonal dimensions of ontologies in IS.

The temporal dimension concerns to the use of ontologies either at run-time or development time. Ontologies at *run-time* refer, for example, to ontologies used to facilitate the process of mapping and sharing database schemas and web services structures, or to enable the communication between software agents (Guarino, 1998). Ontologies at *development-time* refer to the process of creating ontologies that describe a given domain, and the use of these ontologies to support the creation of IS components. On the one hand, designers can make use of ontology as a valuable shared and common knowledge base of a specific domain and its related tasks, available as an ontology library (Guarino, 1998). On the other hand, designers can exploit ontology, as a powerful tool to automatically create or to support the creation of IS components (Fonseca, 2007; Guarino, 1998; Kishore, et al., 2004).

The structural dimension concerns the way an ontology can affect the main IS components (Guarino, 1998), either at run time or development time. First, for the application program component, ontology representing the knowledge embedded in a domain, can help the development of Information Systems. Second, for the information resources component (e.g. databases), ontology can provide knowledge that represent the data structure of a given IS domain, as well as it can provide mechanisms for the integration of different data sources and applications. Finally, for the user interface component, the knowledge about the application program and the information resources can provide information for creating user interfaces, as it can identify which data are needed during a specific part of the application. Also, the ontology can help the user to navigate through the application or to visualize a specific set of data.

This research aims not only to strengthen the importance and benefits of applying ontologies in ISAD (Fonseca, 2007; Guarino, 1998; Guizzardi, 2005; Kishore, et al., 2004; Wand and Weber, 1990), but also to illustrate how ontologies can drive ISAD activities. An important reason for employing ontologies at development time is that "when domain and task ontologies are used during development time, the semantic content about the domain contained within those ontologies can be easily transformed and translated into IS components, thereby enabling knowledge reuse, reducing cost of conceptual analysis, and assuring the ontological adequacy of the IS" (Kishore, et al., 2004). After all, ontology represents the knowledge of a given domain, which ends up being the same knowledge used by designers in their activities (Fonseca, 2007; Zlot, et al., 2002). Hence, we argue that ontology is more than a common vocabulary used by designers to understand the domain being modeled and to support communication between stakeholders. A closer look to the ontological constructs and the relation between them can uncover the IS components of the system under investigation. Thus, we consider ontology to be neither a substitute nor a competitor of existing conceptual schemas, but a significant and complementary resource that can provide formalized knowledge about the universe of discourse (Fonseca and Martin, 2005; Wand, et al., 1995).

During the conceptual modeling phase of an information system, designers face with different levels of generality regarding the knowledge of the domain under investigation. These levels are classified from the most generic (i.e. top-level ontology) to the most specific ones (i.e. application ontology), as shown in Figure 1. Guarino (1998) suggests four "kinds of ontologies, according to their level of dependence on a particular task or point of view" (p.9).

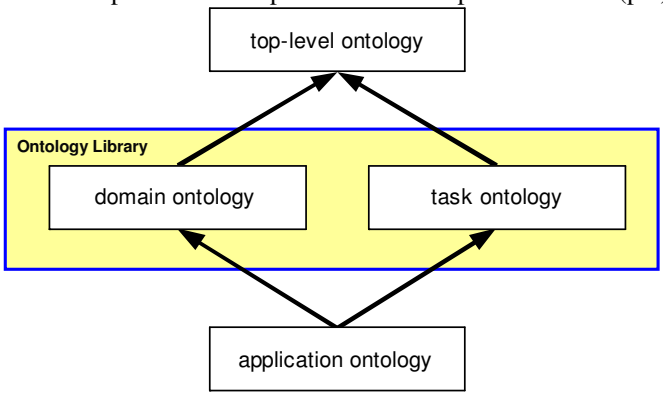

Figure 1: Kinds of Ontologies (adapted from Guarino, 1998)

- *Top-level ontologies* describe very general concepts like space, time, matter, object, event, and actions which are independent of a particular problem or domain.
- *Domain and task ontologies* describe, respectively, the vocabulary related to a generic domain (like medicine, or automobiles) or a generic task or activity (like diagnosing or selling), by specializing the terms introduced in the top-level ontology.
- *Application ontologies* describe concepts depending both on a particular domain and task, which are often specializations of both the related ontologies. These concepts often correspond to roles played by domain entities while performing a certain activity, like replaceable unit or spare component.

The scope of this project, as shown in Figure 2, is to investigate the use of ontologies in the conceptual modeling phase of an IS. Mylopoulos (1992) define conceptual modeling as "the activity of formally describing some aspects of the physical and social world around us for purpose of understanding and communication" (p.3). The main objectives of this activity is to "capture relevant aspects of some world, say an office environment and the activities that take place there, and can serve as points of agreement among members of a group, for example the workers in that office, who need to have a common understanding of that world" (p.3) which is more in line with our view of ontology as a central role in the conceptual modeling phase. Thus, we agree and support Johnson and Henderson's view (2002) that a conceptual model is the backbone of an information system, and "all the other design and implementation activities can and should be grounded in it, feeding it further…, building on it" (p.32).

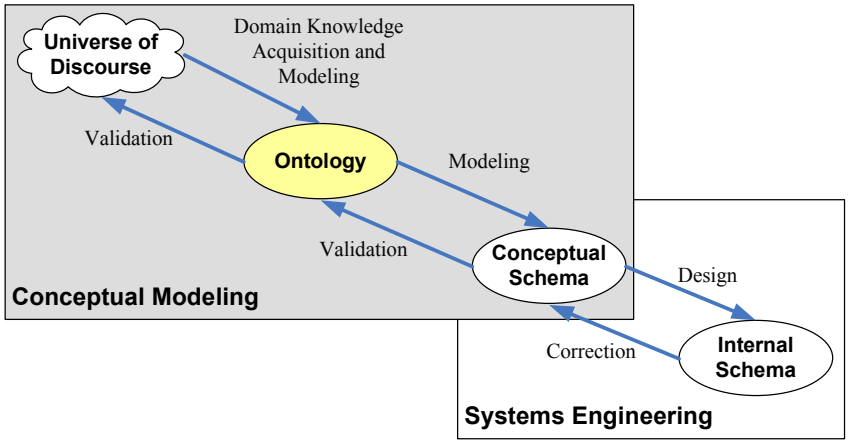

Figure 2: The role of an ontology in the Information Systems life-cycle (extended from Rolland & Prakash's two-phase organization of system life-cycle (Rolland and Prakash, 2000))

It is important to notice that although similar in nature conceptual schemas and ontologies are different in a way. Fonseca and Martin (2007) argue that "ontologies have a broader scope than conceptual schemas" (p.133). Ontologies focus on the universe of discourse and are used to guide the creation of conceptual schemas, while the conceptual schemas focus on (and are limited by) the information system being modeled.

The research proposed in this paper takes into consideration the four elements proposed by Wand and Weber's (2002) research framework, which are:

- *Conceptual-Modeling Grammar* provides a set of constructs and rules that show how to combine the constructs to model real-word domains;
- *Conceptual-Modeling Method* provides procedures by which a grammar can be used. Usually one major aspect of a method prescribes how to map observations of a domain into a model of the domain;

- *Conceptual-Modeling Script* is the product of the conceptual-modeling process. Each script is an statement in the language generated by the grammar;
- *Conceptual-Modeling Context* is the setting in which conceptual modeling occurs and scripts are used.

In the research project we will (1) investigate the existing ontological constructs (grammars) to see if they can represent the knowledge need for IS modeling, (2) identify methods for acquiring knowledge from the domain, (3) propose a methodology (scripts) for creating ontologies and for using the ontologies in the process of creating IS components, and (4) identify which constructs are best suited to the IS modeling and show its feasibility through an experiment (context).

ODIS-DEV FRAMEWORK

We argue that the ontology embedded in the designers mind during the conceptual modeling of the system many times is not correctly represented as an IS artifact, and consequently it cannot be transferred or reused in other ISAD activities. In our work ontology is an important IS artifact responsible for representing the knowledge of a giving domain, and for providing knowledge during IS modeling (Figure 2). In this way, we extend Rolland and Prakash's (2000) two phase organization of system life-cycle (i.e. conceptual modeling and systems engineering). We agree with the view that every information system embeds knowledge about some application and its respective domain (Sowa, 2000), which means it has its own ontology (Guarino, 1998). In fact, modelers often rely on ontological questions to build conceptual modeling (Guarino and Guizzardi, 2006), even though sometimes, they are not aware of the ontological character of these questions. Examples of ontological questions are: Is there a unique identifier for all objects? Is this a whole-part relation? Is this object a property of another object? Therefore, ontology is not often used as an IS artifact in the development phase of an IS. Instead, what happens is the creation of conceptual schemas, which are based on the knowledge produced during the conceptual modeling phase of an IS life cycle.

Each conceptual schema carries and represents a fraction of the knowledge about what a domain is or how it works. Rolland et al (1980) suggest that "the representation of real problems by a model and its related formalism is always an incomplete one because a model act as a filter" (p.93). Thus, in order to create a conceptual schema, only the knowledge that is suitable for a specific schema will be used. An Entity-Relationship schema, for example, only needs three constructs to be represented: Entities, Attributes, and Relations (Dennis, et al., 2005). If not used for representing another conceptual schema, or formalized in another way, the remaining knowledge remains tacit in the minds of users and designers. Thus, the knowledge already acquired would be neither transferable nor reusable in the other IS design phases.

This research proposes a two-phase framework (Figure 3). Phase 1 employs a methodology for building domain ontologies suitable for IS modeling and Phase 2 employs a methodology for using these ontologies to create IS components. We rely on ontologies **of** IS to guide the construction of ontologies **for** IS.

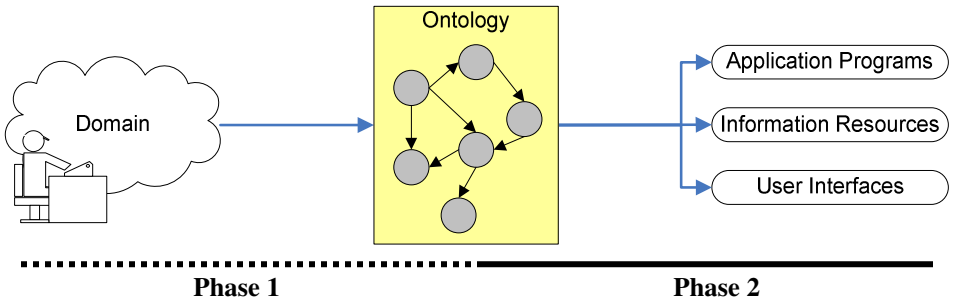

Figure 3: Ontology-Driven Information Systems at development time

One of our objectives is to help designers to create ontologies that are appropriate to be used in ISAD. These ontologies will improve the process of designing systems and consequently the quality of the system delivered. We agree with the view that "theories of ontology provide us with an artifice for describing a perceived world", and that "our description [of the world] will only be as good as our ontologies", similarly "our information systems will only be as good as our ontologies" (Wand and Weber, 2004 p.xiii).

In this project, we envision the creation of ontologies guided by theories of ontology and their following representation as machine-readable ontologies (i.e. computational ontologies) for supporting the process of building IS components. Whereas ontologies are created, we can expose them to specialized reasoning mechanisms that are able to create the IS components. The endeavor of this paper is to provide an outline of the architecture for ODIS-Dev with emphasis on the usefulness of ontology in Information Systems development. The implementation description is beyond the scope of this paper.

We see ontology as a repository of the system's knowledge that provides relevant information regarding to the data structure, the interface, and the processes of the system. We envision an automatic mapping of the ontology into the IS components that is performed with the support of reasoners. Specialized reasoning mechanisms can perform inferences over the ontologies to create IS components. According to Uschold and Gruninger (2004), "the fundamental role of a reasoning engine is to derive new information via automated [logical rules of] inference" (p.60). Figure 4 presents our view of an ontology mapped into IS components:

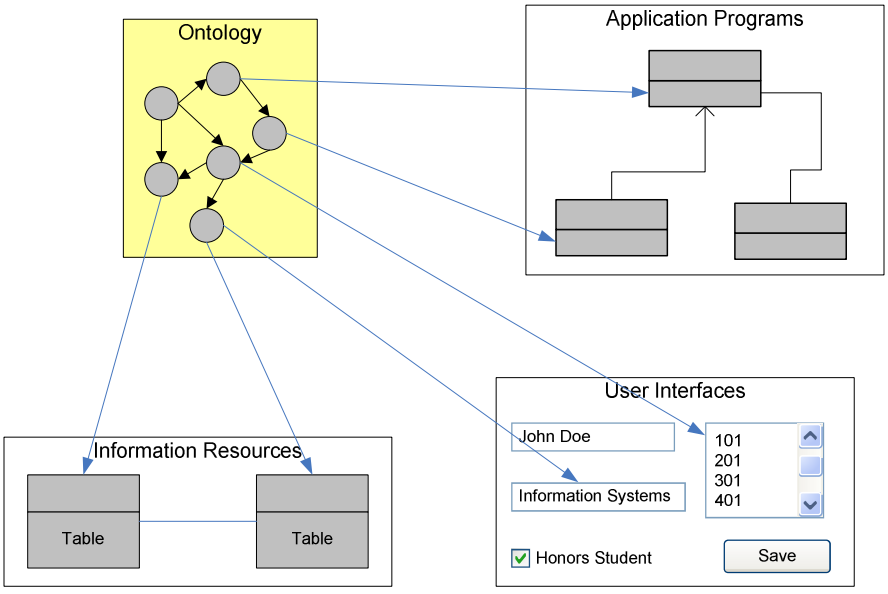

Figure 4: Ontology driven IS components

- The *Information Resources* component represents the structure used to store the data of the system. The most common structure is the Entity-Relationship Model (Chen, 1976). Here, ontology constructs can be mapped to E-R constructs to help generate the information resources. Sugumaran and Storey (2006) show the feasibility and usefulness of domain ontologies to support database design.
- The *Application Programs* component usually "contain a lot of domain knowledge, which, for various reasons, is not explicitly stored in the database. Some parts of this knowledge are encoded in the static part of the program in the form of type or class declarations, other parts (like for example business rules) are implicitly stored in the (sometimes obscure) procedural part of the program" (Guarino, 1998 p.13). The

ontologies created in Phase 1 provide knowledge to build application programs, as it reflects the processes that occur in a given domain.

- The *User Interfaces* component convey the inputs and outputs of the communication between the system and the users, and are based on the constraints imposed by the other two components, especially from the application programs, which is where and when the information is needed. Interfaces are formed through the information available in the domain ontology (e.g. classes, attributes and relationships).

Ontology, in phase 2, acts as a link connecting the concepts of a domain with the IS components, this way supporting reuse and traceability of existing concepts. If an IS component needs to be modified, it must occur, first in the ontology that generated the component. Thus, because of the links between the ontology and the IS components, the ontology can help to propagate the changes throughout the other components, and to keep the components synchronized with the conceptual model.

RESEARCH METHODOLOGY

The aim of this research is to use ontology as an IS artifact that can help designers in the process of conceptual modeling of IS. To achieve this goal, we propose a research methodology that is founded in the paradigm of the Design Science or Design Research (Hevner, et al., 2004; Simon, 1996). According to March and Smith (1995), Design Science is an approach with two main activities, "building is the process of constructing an artifact for a specific purpose; evaluation is the process of determining how well the artifact performs" (p.254). The ultimate objective of Design Science is "to develop valid and reliable knowledge to be used in designing solutions to problems" (Aken, 2004 p.225), which is the knowledge that is used by practitioners in their own field. The most important characteristics of Design Science, according to Jarvinen (2007), are:

- Design Science solves constructions problems (producing innovations) and improvement problems (improving the performance of existing entities).
- Design Science produces design knowledge (concepts, constructs, models, and methods).
- Building and evaluation are the two main activities of Design Science.
- Design science's products are assessed against criteria of value or utility.
- Design science research is initiated by the researcher(s) interested in developing technological rules for a certain type of issue. Each individual case is primarily oriented at solving the local problem in close collaboration with the local people.
- Knowledge is generated, used and evaluated through the building actions.

In this project, we propose two distinct phases (Section 0), where each phase produces a distinct artifact. For the sake of simplicity, we will briefly introduce each step of the design life cycle shown in Figure 5 (Vaishnavi and Kuechler, 2004), and then we will illustrate the steps with examples from both phases.

- *The awareness of problem step* is the beginning of the methodology of the design research. Here we define a problem that can be solved through the development of an artifact. In phase 1, our problem is that ontologies should correctly represent knowledge of IS and domain modeling. In phase 2, the problem is that we need specialized reasoning mechanisms to read the ontologies, in order to create the IS components.
- *The suggestion step* is a creative process, where designers envision a potential solution for the problem intended to be solved. In phase 1, we propose to develop a methodology for creating ontology that takes into account the knowledge need by the IS components. In phase 2, we suggest a methodology for translating the ontologies created in phase 1 into IS components.
- *The development step* is where the artifact is implemented accordingly to the suggested solutions from the previous step. Phase 1 implement a methodology for creating ontology in the form of a tool that can help designers to better capture the knowledge of a domain and place it to the right ontological constructs. Phase 2 also

implement a tool, but in this case, the tool carries specialized reasoning that are able to create or to support the creation of the IS components by reasoning over the ontologies.

- *The evaluation step* is responsible for testing the newly implemented artifact against the specification in the proposal step. If the result is considered unsatisfactory, the design cycle goes back to the awareness of problem step for reviewing the initial problem, bringing together the new knowledge (i.e. circumscription) acquired throughout the steps. Otherwise, if the result is satisfactory, the cycle moves on to the conclusion step. This loop may happen many times until the evaluation of the artifact is considered satisfactory. In phase 1, the evaluation occurs by creating an ontology of a given domain, and also by comparing this ontology with ontologies created with other existing methodologies. In phase 2, a successful evaluation is achieved when we can demonstrate the creation of IS components from the ontologies. Moreover, we can compare the IS components created from the ontology with the components built by a set of experienced IS designers.
- *The conclusion step* is the end of the Design Research cycle. Now, that the artifact has been built and the evaluation turned out to be satisfactory, the designer will put together the knowledge acquired throughout the design cycle, providing a guideline (i.e. body of knowledge) for practitioners to use the artifact in their field. In both phases, we generate a guideline (i.e. methodology) explaining how to use the artifacts in similar situations.

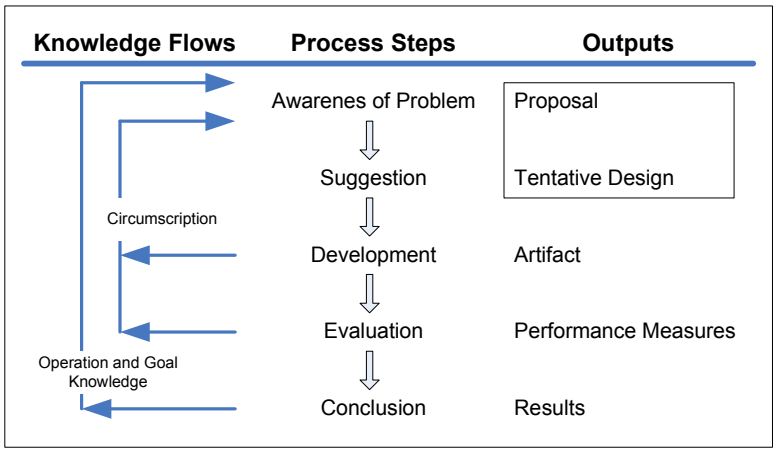

Figure 5: The General Methodology of Design Research (Vaishnavi and Kuechler, 2004)

The Design Science research produces four general outputs (March and Smith, 1995): *Constructs* (i.e. the vocabulary of a problem and its solutions), *Models* (i.e. the relationship among the constructs), *Methods* (i.e. a guideline to perform a task), and *Instantiation* (i.e. is the arrangement of constructs, models and methods). A fifth output, called *better theories* is proposed by Purao (2002). It suggests that design research can provide better theories, as the artifacts can be proved, tested, and reproduced in the same sense as in the natural sciences.

CONCLUSION

In this paper we presented an outline of a framework for Ontology-Driven Information Systems at development time (ODIS-Dev). We discussed the process of creating ontologies for IS modeling, as well the process of creating IS components from the knowledge stored in the ontologies. The research has demonstrated that ontologies can definitely play an important role in ISAD. Moreover, we hope to have shown that ontology is not a new thing in the process of

conceptual modeling, but an already existing and reliable source of knowledge that can be represented as an IS artifact, and used throughout ISAD.

Although this research shows the use of ontologies as a two-phased framework, practitioners can use the outcomes of each phase independently. The main idea behind phase 1 is to provide a methodology for constructing ontologies for IS modeling, so practitioners can use whatever tool (e.g. ontology editor) they are comfortable with, as long as they can provide ontologies with the same characteristics suggested in phase 1. Similarly, practitioners can bring their ontologies (built elsewhere, but comprising with the guidelines of phase 1) to phase 2, so they can use the methodology to create IS components. This flexibility emphasizes the importance of Design Science to the creation of a body of knowledge (regarding to the problem under investigation) that is suitable for practice.

Besides the benefit of using ontologies to create the IS components, we also observed that ontology can work as a bridge connecting the IS components, where the links between the ontology and the components create a powerful mechanism for knowledge reuse and traceability.

REFERENCES

Aken, J. E. v. (2004). "Management Research Based on the Paradigm of the Design Sciences: The Quest for Field-Tested and Grounded Technological Rules". *Journal of Management Studies,* **41**(2), 219-246.

Chen, P. S. S. (1976). "The Entity-Relationship Model: Towards a Unified View of Data". *ACM Transactions on Database Systems,* **1**(1), 9-36.

Dennis, A., Wixom, B. H., and Roth, R. M. (2005). *Systems Analysis Design* (Third ed.): John Wiley & Sons, Inc.

Fonseca, F. (2007). "The Double Role of Ontologies in Information Science Research". *Journal of the American Society for Information Science and Technology,* **58**(6), 786-793.

Fonseca, F., and Martin, J. (2007). "Learning the Differences between Ontologies and Conceptual Schemas through Ontology-Driven Information Systems". *Journal of the Association for Information Systems,* **8**(2), 129-142.

Fonseca, F. T., and Martin, J. E. (2005). "Toward an Alternative Notion of Information Systems Ontologies: Information Engineering as Hermeneutic Enterprise". *Journal of the American Society for Information Science and Technology,* **56**(1), 46-57.

Gangemi, A., Guarino, N., Masolo, C., and Oltramari, A. (2003). "Sweetening WorldNet with DOLCE". *AI Magazine,* **24**(3), 13-24.

Guarino, N. (1998). *Formal Ontology in Information Systems.* FOIS'98, Trento, Italy.

Guarino, N., and Guizzardi, G. (2006). "In defense of Ontological Foundations for Conceptual Modeling". *Scandinavian Journal of Information Systems,* **18**(1), 115-126.

Guizzardi, G. (2005). *Ontological Foundations for Structural Conceptual Models.* Ph.D. Thesis, University of Twente, Twente, The Netherlands.

Guizzardi, G., and Wagner, G. (2004). Some Applications of a Unified Foundational Ontology in Business Modeling. In P. Green & M. Roseman (Eds.), *Business Systems Analysis with Ontologies* (pp. 345-367). Hershey, PA: Idea Group Publishing.

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

Jarvinen, P. (2007). "Action Research is Similar to Design Science". *Quality & Quantitiy,* **41**, 37-54.

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

Kishore, R., Sharman, R., and Ramesh, R. (2004). "Computational Ontologies and Information Systems: I. Foundations". *Communications of the Association for Information Systems,* **14**(8), 158-183.

March, S. T., and Smith, G. F. (1995). "Design and natural science research of information technology". *Decision Support Systems,* **15**, 251-266.

Myloupolos, J. (1992). Conceptual Modeling and Telos. In P. Loucopoulos & R. Zicari (Eds.), *Conceptual modeling, databases, and CASE* (pp. 49-68): Wiley.

Purao, S. (2002). *Design Research in the Technology of Information Systems: Truth or Dare*. University Park, PA: College of Information Sciences and Technology, Penn State University.

Rolland, C., and Prakash, N. (2000). "From conceptual modelling to requirements engineering". *Annals of Software Engineering,* **10**, 151-176.

Rolland, C., Richard, C., and Leifert, S. (1980). "A Proposal for Information Systems Designs and Management". *ACM SIGDA Newsletter,* **10**(2), 83-109.

Simon, H. A. (1996). *Sciences of the Artificial* (3rd ed.): MIT Press.

Sowa, J. F. (2000). *Knowledge Representation: Logical Philosophical, and Computational Foundations*. Pacific Grove, CA: Brooks Cole Publishing Co.

Studer, R., Benjamins, V. R., and Fensel, D. (1998). "Knowledge Engineering: Principles and Methods". *IEEE Transactions on Data Knowledge and Engineering,* **25**(1-2), 161-197.

Sugumaran, V., and Storey, V. C. (2006). "The Role of Domain Ontologies in Database Design: An Onotlogy management and Conceptual Modeling Environment". *ACM Transactions on Database Systems,* **31**(3), 1064-1094.

Uschold, M., and Gruninger, M. (2004). "Ontologies and Semantic for Seamless Connectivity". *SIGMOD Record,* **33**(4).

Vaishnavi, V., and Kuechler, W. (2004). Design Research in Information Systems, http://www.isworld.org/Researchdesign/drisISworld.htm (accessed May 08, 2007)

Wand, Y., Monarchi, D. E., Parsons, J., and Woo, C. (1995). "Theoretical Foundations for Conceptual Modelling in Information System Development". *Decision Support Systems,* **15**, 285-304.

Wand, Y., and Weber, R. (1989). *An Ontological Evaluation of Systems Analysis and Design Methods.* Information Systems Concepts: An in-Depth Analysis.

Wand, Y., and Weber, R. (1990). "An Ontological Model of an Information System". *IEEE Transaction on Software Engineering,* **16**, 1282-1292.

Wand, Y., and Weber, R. (2002). "Research Commentary: Information Systems and Conceptual Modeling: A Research Agenda". *Information Systems Research,* **13**(4), 363-376.

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

Yildiz, B., and Miksch, S. (2007, Feb 21-23). *Ontology-Driven Information Systems: Challenges and Requirements.* International Conference on Semantic Web and Digital Libraries (ICSD-2007), Bangalore, India.

Zlot, F., Oliveira, K. M. d., and Rocha, A. R. (2002). *Modeling Task Knowledge to Support Software Development.* SEKE'02, Ischia, Italy.

Received: February 4th 2007
Accepted in final form: December 15th 2007

About the authors

Andrey Soares is a Ph.D. candidate in the College of Information Sciences and Technology at the Pennsylvania State University. He holds a master's degree in Computer Science from the Federal University of Santa Catarina (Brazil). His areas of research include Ontology in Information Systems, Analysis and Design and Design Science.

Dr. Frederico Fonseca is an Associate Professor in the College of Information Sciences and Technology at the Pennsylvania State University. He earned his Ph.D. from the Department of Spatial Information Science and Engineering at the University of Maine. His areas of research include Ontology in Information Systems and Geographic Information Systems.