Ontology-based requirements representation in the process of building information systems

Andrzej Sobczak
Department of Business Informatics,
Warsaw School of Economics
sobczak@sgh.waw.pl

ABSTRACT

The article presents a technique of representation of requirements using an ontology-based group memory system. The technique draws on achievements of a few research areas, namely requirement engineering, ontology engineering, and building group memory systems. In line with the project assumptions, the research focused on management information systems tailored to the customer needs. Additionally, the paper provides an overview of a reference ontology for requirement presentation, implemented in the OWL language and constituting a component of the FAROS technique. Also, the paper describes a pilot deployment of the proposed research tools for a selected component of an electronic platform for municipal public services system.

Keywords

Requirements Engineering, Ontology, OWL, Group Memory System, public administration.

INTRODUCTION

One of the critical factors influencing successful implementation of a management information system is the area of requirement engineering in a broad sense, and in particular the correct representation of requirements elicited from the stakeholders. Such representation should constitute a consistent whole, despite the fact that it is based on information gathered from various groups participating in the project implementation, which often are very diversified in terms of their professions and competencies. A significant obstacle is the fact that the requirements gathered from key stakeholders have often a very imprecise and poorly structured form. In addition, such stakeholders typically are found in multiple geographically dispersed locations, potentially resulting in problems with ensuring proper quality of the prepared requirement documentation.

These considerations led to undertaking research works[1] to develop a technique of formal representation of requirements using an ontology-based group memory system[2], known as the FAROS technique (in Polish: FormAlna Reprezentacja wymagań z zastosowaniem bazującego na Ontologii Systemu pamięci grupowej). In developing this technique, the following assumptions have been adopted:

- Defining and representing requirements is a very intellectual and creative activity, much more knowledge-intensive than any other activity related to building an IT system.
- The difficulties in preparing the software requirement specification result from the lack of mutual understanding between the requirement engineer and the customer:
  - often, the customer uses a jargon or vocabulary different to that of the requirement engineer;
  - each party may use overlapping or mismatching concepts.
- Lack of well-arranged knowledge of the subject matter is often a significant problem at the requirement specification stage.

During the FAROS development, the following two objectives of using the technique in gathering requirements for a management information system have been identified:

- obtaining a list of properly formulated requirements for the management information system;
- accumulation of knowledge about requirements for management information systems in order to reuse some already defined requirements and therefore reduce the costs of building next similar systems.

[1] Research project No. 1 H02D 016 27 financed by the Polish Ministry of Science and Higher Education.

[2] A group memory system is an IT system which integrates knowledge representation with semantic data modeling techniques in order to organize, store, and distribute, on a continuous basis, knowledge related to a collectively implemented project (in particular a project aimed to gather and represent requirements) (Yu, Thom and Alem, 2004).

This paper has the following structure: Section 2 presents the current state of knowledge in the area of requirement representation. Sections 3 through 7 describe the FAROS technique — its structure, components (including the reference ontology), and issues related to its adaptability for a specific requirement-documenting project (in this respect, the Observe-Orient-Decision-Act Loops are proposed). In Section 8, the two-level verification of FAROS, performed within the framework of the project, is discussed. Section 9 contains a summary and presentation of the directions of further works on FAROS.

CURRENT STATE OF KNOWLEDGE IN THE AREA OF REQUIREMENT REPRESENTATION

The research and practical applications in the area of requirement representation have been focusing to-date on the following two fields:

- using the natural language to document requirements;
- using semi-formal and formal structures, including graphical notations, to represent requirements.

Using the natural language in the requirement documenting process makes it impossible in many cases to obtain a correct (i.e. unambiguous and consistent) list of requirements. This is because of a barrier created by ambiguousness of understanding the same text by different stakeholders.

That disadvantage does not exist in case of techniques based on semi-formal and formal requirement-representation structures. Such structures include (among others) the following:

- pseudo-code (Lamsweerde 2001; Leffingwell and Widrig 2000);
- special requirement documenting notations (Greenspan, Mylopoulos and Borgida 1994; Kowalczykiewicz and Nawrocki 2001; Lamsweerde 2001; Tse and Pong 1991);
- mathematical formalisms (ESA Board for Software Standardisation and Control 1995, Lamsweerde 2000; Lamsweerde 2001);
- finite-state machines (ESA Board for Software Standardisation and Control 1995; Leffingwell and Widrig 2000);
- decision trees and decision tables (ESA Board for Software Standardisation and Control 1995; Leffingwell and Widrig 2000);
- graphical decision tables (ESA Board for Software Standardisation and Control 1995; Leffingwell and Widrig 2000);
- use case diagrams (ESA Board for Software Standardisation and Control 1995, Santander and Castro 2001).

At present, attempts are being made in various countries to develop new requirement representation techniques using knowledge engineering (inter alia, (Herlea, Catholijn, Treur and Wijngaards 2002; Robinson, Pawlowski and Volkov 1999). This is related to the fact that more and more significance is being attached to an approach according to which "the most important knowledge is hidden in people and in the organization; that knowledge needs to be articulated and formalized and only on this basis [an IT system] should be built" (Cempel 2003).

FAROS STRUCTURE AND COMPONENTS

The FAROS technique draws on achievements of a few research areas, namely requirement engineering, ontology engineering, and group memory system building. Figure 1 presents the high-level structure of FAROS. It is assumed in this technique that the ontology building process may be iterative and incremental. The horizontal axis represents time and corresponds to the successive stages of ontology building. The vertical axis represents main areas (referred to as "disciplines") of ontology building.

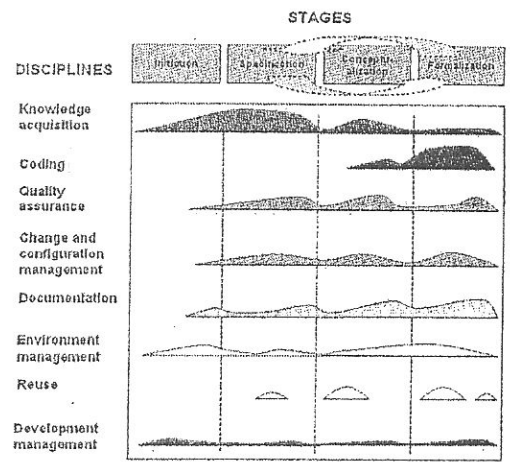

Figure 1. Stages and disciplines in FAROS
Source: own study inspired by (IBM 2005)

In the above figure, fillings represent estimated levels of stakeholders' engagement in individual disciplines at various stages of ontology building.

The FAROS technique has been described using the following concepts: role, action, discipline, artifact, source document, tools (o reference ontology for requirement representation, checklist, template, software).

The concepts listed above have been defined for the purposes of FAROS as follows:

- A role defines actions to be performed by individual stakeholders or stakeholder groups under FAROS. One stakeholder may play multiple roles. A role has a defined list of actions which may be performed by the stakeholder and a defined list of artifacts for which the stakeholder is responsible or co-responsible.
- Action - a portion of work, which may be performed under a given role. Each action has a clearly defined purpose. In result of performing an action, a new artifact is created or an existing artifact is modified. Individual actions may be combined into action groups.
- Discipline - a systematic combination of a number of action groups.
- Artifact - a result of a specific action performed under a specific role.
- Source document - a permanently recorded piece of knowledge, represented in the natural language (such as documentation, messages, text files, books, operation manuals, notes, protocols, etc.). Some roles use source documents for everyday work and other roles analyze them for the purpose of ontology building.
- Tool - a deliberately developed element (simple or complex), which enables performing a specific task or work.

The following tools are used in FAROS:

- Reference ontology for requirement representation - this is an ontology created for the purpose of identification and definition of main concepts related to the requirement representation, to be subsequently used for building requirement representation ontology for a specific IT system.
- Checklist - an instrument used to ensure high quality of artifacts prepared under individual roles.
- Template - a model artifact supporting a specific action group.
- Software - computer software supporting a specific action group. FAROS is augmented by appropriate computer software, running chiefly in a distributed environment, in particular including OntoMemory, a program developed by the project manager, used to export an ontology from the ontology building tool to a group memory system. The software is based on open source solutions.

FAROS DISCIPLINES

The actions to be performed under FAROS are grouped into eight disciplines:

- Knowledge acquisition - within this discipline, knowledge on the implemented system is being elicited from the stakeholders.
- Coding - within this discipline, an informal version of the ontology is built, which is subsequently presented with a specific formal language suitable for computer processing.
- Quality assurance - this discipline includes actions performed in order to assure high quality of the ontology and its documentation.
- Change and configuration management - this discipline includes actions performed in order to ensure effectiveness of the ontology building process, with particular focus on consequences of any changes to the developed ontology.
- Documentation - this discipline includes actions performed in order to ensure high quality of the developed ontology documentation.
- Environment management - this discipline includes actions performed in order to build and operate an effective IT platform used for building the ontology and related artifacts.
- Reuse - this discipline includes actions performed in order to reduce the ontology building costs by reusing previously developed ontologies.
- Development management - this discipline includes organizational measures undertaken in order to ensure that the ontology is delivered on time, within the specified budget, and within the specified resource limits.

Each of the FAROS disciplines is described in a unified manner, with the following items:

- Discipline overview - a short description of the scope of the given discipline.
- Actions performed within the discipline - a detailed description of the actions to be performed within the given discipline, with identification of the roles under which such actions are performed.
- Timeframe - identification of those stages of the requirement representation works when the given discipline is particularly important.
- Supporting tools - a description of any supporting tools used within the given discipline.
- References to other disciplines - identification of how actions or action groups within the given discipline are related to other disciplines

FAROS STAGES

FAROS includes four successive stages. Within each stage, specific actions are performed under specific disciplines (some actions are combined into action groups). The number of such actions is different at various stages.

Stage I (Initiation) is performed once, using FAROS to represent the requirements for the given system. In contrast, Stages II through IV may be repeated iteratively until the assumed quality level of the developed ontology is achieved.

Educating the stakeholders on the knowledge engineering is outside the scope of FAROS. It is assumed that the stakeholders commence using the proposed technique after being acquainted with:

- the basic concepts of ontology engineering
- assumptions of the FAROS technique
- software supporting FAROS.

It is proposed that the necessary education is provided within the so-called Stage 0 (Preparation).

Each of the FAROS stages is described in a unified manner, with the following items:

- main objectives of the stage
- main roles in the stage
- input and output artifacts
- a description of actions performed within the stage.

Stage I - Initiation

Main objectives of the Initiation stage:

- definition of the scope of the developed ontology
- identification of the stakeholders of the implemented system
- specification of the goals of the implemented system
- specification of the formalism level of the developed ontology
- preparation of the ontology development plan
- preparation of the risk register
- preparation of the quality assurance plan for the developed ontology
- preparation of the change and configuration management plan for the developed ontology
- providing templates customized to the needs of the developed ontology
- providing an IT environment to be used for ontology development.

Stage II - Specification

Main objectives of the Specification stage:

- identifying and defining key concepts and relationships in the domain of the implemented system and representing them in the natural language
- developing motivation scenarios and corresponding informal competence questions
- assessment of the possibilities of reusing previously developed ontologies.

Stage III - Conceptualization

Main objectives of the Conceptualization stage:

- building the glossary of terms
- building the taxonomy
- building the concept dictionary
- structuring the acquired knowledge in the form of an informal version of the ontology for the implemented system.

Stage IV - Formalization

Main objectives of the Formalization stage:

- explicit representation of the conceptualization, using a specific formal language suitable for computer processing
- expressing formal competence questions using the ontology-specific terminology
- adding information about the newly developed ontology to the ontology library.

REFERENCE ONTOLOGY FOR REQUIREMENT PRESENTATION

One of the core FAROS tools is the reference ontology for requirement presentation, known as ORFA (in Polish: Ontologia ReFerencyjna do reprezentacji wymAgań). It is used as a starting point for developing a requirement representation ontology for a specific IT system.

The reference ontology has been developed in order to identify and define the main concepts related to requirement representation. It is based on the concept presented in (Andersson, Bergholtz, Edirisuriya and Ilayperuma 2006).

While developing ORFA, the following observations have been taken into consideration:

- the necessity to identify the objectives of introducing requirements for the implemented system;
- the necessity to tightly link the business and technological aspects using a service-based approach;

- the increasing importance of non-functional requirements and, at the same time, absence of measurable criteria for them.

The ORFA reference ontology has been implemented in the Protégé tool, using the OWL language.

The reference ontology is divided into three layers:

- strategic layer;
- business layer;
- technology layer.

The purpose of the ORFA strategic layer is to provide a terminology to describe, at the highest level of abstraction, the business goals of stakeholders, who may represent specific organizations.

The ORFA business layer provides a terminology for a high-level description of the manner of implementing the business services by a selected part of the organization (the business component).

The purpose of the ORFA technology layer is to provide a terminology for a high-level description of the manner of implementing the system services by a selected part of the IT system (the system component).

ADAPTABILITY OF FAROS

The FAROS technique, its supporting IT tools, and the ORFA reference ontology should not be regarded as a closed solution. Those elements should be adjusted to the implementation needs of a specific project of documenting the requirements for a deployed IT system.

It seems appropriate to suggest an approach based on the so-called OODA Loop, which includes the following steps (Stokalski 2005):

- Observation — how the generic version of FAROS and its related elements function in real conditions and what are the needs of the specific project of documenting requirements for the implemented IT system.
- Orientation — drawing conclusions on the extent to which the current version of FAROS and its related elements ensure fulfillment of their basic task, namely an effective representation of requirements in the given project.
- Decision — planning the activities to shape the modified version of FAROS and its related elements for the purposes of the specific project.
- Act — performing actions to shape the modified version of FAROS and its related elements.

VERIFICATION OF FAROS

A preliminary verification of FAROS usability has been carried out using the benchmarking method. As a reference, the requirement engineering best practices method, developed by Ian Sommersville and Pete Sawyer (Sommerville and Sawyer 1997), has been assumed. It has been found that if FAROS is adopted as an obligatory standard in an organization, such organization will achieve the "defined level" (the highest level in the three-level scale for assessing maturity of requirement engineering processes, as proposed by Sommersville and Sawyer). This result provides a preliminary confirmation of FAROS usability for the purposes of documenting requirements for IT systems in organizations.

Our project included also a pilot deployment of the proposed research tools (both the FAROS technique and computer software) for a selected component of the e-PUMA electronic platform for municipal public services (in Polish: Elektroniczna Platforma Usług publicznych MiAsta) . FAROS has been verified in a sample city district in central Poland. Using FAROS for the e-PUMA system, it was possible to initiate the process of accumulating knowledge on requirements for e-government systems for cities. Continuation of this process will enable reuse of already specified requirements in various public administration units, resulting in cost reduction related to building other IT systems of the same type.

CONCLUSION AND DIRECTIONS OF FURTHER WORKS

In comparison with the existing methods of representing requirements for management information systems, using FAROS brings the following benefits:

- common understanding of terms used by the stakeholders during the system development, in particular thanks to: understanding by the system developer the statements used by the customers, reduction of conceptual confusion between the customer and system developer, specifying guidelines requiring the customers to express their needs in a systematic manner;
- a possibility to reuse the requirements;
- facilitation of the process of automatic requirement verification thanks to a formal representation of the knowledge.

The experience gathered from the pilot deployment indicates that introduction of a systematized approach to requirement management, based on formal or semi-formal solutions, is appropriate mainly in those public institutions which have complex IT solutions.

Requirements gathered with FAROS may be:

- used as a basis for preparing a description of the subject matter of an order — such description is

a key element of the "Specification of essential order conditions", which in public administration constitutes a basis for conducting a public procurement procedure for building an IT system;

- helpful in ensuring compliance with the Parliamentary Act regarding computerization of entities performing public tasks, to the extent of the following aspects:

  - specification of functional assumptions for IT projects used in public administration

  - definition of technical assumptions for IT projects used in public administration, specifying functions, hardware components, and software components of IT systems.

The initial expenditures required by FAROS, related to training the employees, software deployment, and putting the necessary procedures in place, will be recovered if multiple projects based on this technique are carried out. Another category of FAROS users may be IT firms implementing projects for the public administration. Some requirements are unique, but other ones may be "reused", i.e. already built ontologies (from previously implemented projects) may be used as a basis for subsequent solutions.

At the same time it is predicted that future works related to the FAROS development will include, among other things, the following:

- integration of the FAROS technique with the RUP and OpenUP methodologies (at the level of products, roles, production processes, and management processes);
- creating a hypertext knowledge base for FAROS (for this purpose, it is proposed to use EPF Composer, an open-source tool developed under the Eclipse project);
- development of a public repository, maintained on a non-commercial basis, containing ontologies from various areas of computerization of public administration entities, following the example of the OntoSelect Ontology Library solution.

REFERENCES

1. Andersson, B., Bergholtz, M., Edirisuriya, A., Ilayperuma, T., (2006) Towards a Reference Ontology for Business Models, *Proceedings of the 25th International Conference on Conceptual Modeling (ER2006)*, Tucson, Arizona, USA.
2. Cempel, Cz., Knowledge Society: New Dimension in Create and Use Knowledge (2003) (in Polish: Społeczeństwo wiedzy - nowy wymiar kreowania i użytkowania wiedzy), *eGov.pl website*.
3. ESA Board for Software Standardisation and Control, (1995) Guide to the user requirements definition phase, *European Space Agency*.
4. Greenspan, S., Mylopoulos, J., Borgida, A., (1994) On Formal Requirements Modeling Languages: RML Revisited, Software Engineering, 1994. *Proceedings of the ICSE-16*, 16th International Conference on Software Engineering, Italy.
5. Herlea, D., Catholijn, J., Treur, J., Wijngaards, N., (2002) A Compositional Knowledge Level Process Model of Requirements Engineering, International *Journal of Software Engineering and Knowledge Engineering*, Vol. 12, No. 1, 41-75.
6. IBM Corporation (2005), Rational Unified Process, version 2003.06.15.
7. Kowalczykiewicz, K., Nawrocki, J., (2001) Requirements documentation Using XML (in Polish: Dokumentowanie wymagań w języku XML), *III Krajowa Konferencja Inżynierii Oprogramowania KKIO 2001*, Mikom, Warszawa, 83-92.
8. Lamsweerde, A., (2000) Formal Specification: a Roadmap, *Proceedings of the International Conference on Software Engineering*, Limerick, Ireland, 147-159.
9. Lamsweerde, A. (2001), Building Formal Requirements Models for Reliable Software, *Invited Paper for 6th International Conference on Reliable Software Technologies*, Ada-Europe 2001, Leuven.
10. Leffingwell, D., Widrig, D., (2000) Managing Software Requirements: A Unified Approach, Addison-Wesley Pub Co.
11. Robinson, W., Pawlowski, S., Volkov, V., (1999) Requirements Interaction Management, Georgia State University Working Paper 99-7, USA.
12. Santander, V., Castro, J., (2001) Developing Use Cases from Organizational Modeling, *WER'2001 - IV Workshop de Engenharia de Requisitos*, Buenos Aires.
13. Sommerville, I., Sawyer, P., (1997) Requirements Engineering: A Good Practice Guide, John Wiley & Sons, 1st edition.
14. Stokalski, B., (2005) Architektura korporacyjna, *Seminarium dla Ministerstwa Finansów zrealizowane przez firmę Infovide*, Warszawa.
15. Tse T., Pong, L., (1991) An Examination of Requirements Specification Languages, *The Computer Journal*, 34, 143-152.
16. Yu, J., Thom J., Alem, L., (2004) Group Memory Based on the Task Information, *Proceedings of the Recherche d'Information Assistée par Ordinateur (RIAO)*, France, April 26-28, 329-347.