STUDIO: Ontology-Centric Knowledge-Based System

Réka Vas

1 Introduction

Why is STUDIO a knowledge based system? How can STUDIO make an organization more knowledgeable? Knowledge-based systems (KBSs) have been an important topic in research for quite some time. The literature defines these systems in many different ways. The simplest definitions (such as Laudon & Laudon 1997[1]) describe KBSs as organizational information systems that could provide help in managing the knowledge assets of the organization. These definitions, however, are too general, since any information systems used for handling knowledge (e.g.: expert systems, data warehouses, group decision support systems or intranets) are included in the class.

Another set of definitions focus on the architecture of KBSs (Lucas & van der Gaag 1991; Akerkar & Sajja 2009). Usually three major components are distinguished: a *knowledge base* that is a repository of formal knowledge, an *inference engine* that defines the ways how the formal knowledge may be put to use and a *user interface* where "how" and "why" questions are asked. In some cases additional components are added to the above listed ones that provide instruments for filling the knowledge base, support the explanation and reasoning of decisions or enable self-learning for users (Akerkar & Sajja 2009). In practice, however, it is difficult to separate the aspects of an inference engine and knowledge base—just like in the case of STUDIO—that may hamper understanding the role and capabilities of the system

[1] In later versions of their book they have used the expression "knowledge management system" instead of "knowledge-based system".

R. Vas (✉)
Corvinus University of Budapest, Budapest, Hungary
e-mail: reka.vas@uni-corvinus.hu

A. Gábor, A. Kő (eds.), *Corporate Knowledge Discovery and Organizational Learning*, Knowledge Management and Organizational Learning 2,
DOI 10.1007/978-3-319-28917-5_4

in question. In STUDIO the knowledge is stored in the form of a domain ontology that either has inherent reasoning capabilities or may provide a base for independent applications with inference options.

In addition to these definitions several works exist that focus on the knowledge modelling aspects of KBSs. More precisely, the emphasis is placed on finding a way of formal knowledge representation. The most widely used and universally accepted techniques are logical representation, production rule representation, frame representation and semantic networks. The major advantage of defining the KBS as an outcome of a knowledge modelling process is that attention is directed to the identification of *which elements of the organizational knowledge can validly be described* in any of the formalisms for knowledge representation (Hendriks & Vriens 1999). In other words, in the course of knowledge modelling, it has to be established which part of the organizational knowledge can be identified in a formal schemata (and the kind of questions that can be answered using this formal knowledge model). An additional advantage of this approach—besides the formalization of organizational knowledge—is that efforts could be more effectively coordinated in order to explore all potential functionality of the KBS. STUDIO, on one hand provides a framework for the formal representation of knowledge in the form of a domain ontology. On the other hand, based on the context given by the formalized knowledge STUDIO also supports the design and implementation of various knowledge based applications (e.g.: adaptive knowledge testing, learning style detection, human resource preselection, etc.).

Why does STUDIO use ontologies for knowledge representation? Ontology as a tool of artificial intelligence, knowledge management, and a theoretical tool of database modelling, attempts to describe the world on a conceptual level. According to the most quoted definition "ontology is a formal explicit specification of a shared conceptualization" (Gruber 1993 p 199). That is, an ontology states knowledge explicitly to make it accessible for machines; determines knowledge only of a particular domain of interest[2] in a conceptual way applying symbols that represent concepts and their relations. While shared means that there is a consensus concerning all elements of the conceptual model. Corcho and his colleagues—based on Gruber's definition—have constructed a more precise and applicable definition: "ontologies aim to capture consensual knowledge in a generic and formal way, so that they may be reused and shared across applications (software) and by groups of people. Ontologies are usually built cooperatively by a group of people in different locations" (Corcho et al. 2003, p. 44). In other words by developing uniform conceptualizations of the domains of interest, ontologies have *consensus generating power* enabling efficient cooperation even on the organizational level. Besides knowledge sharing, ontologies also play an important role in *keeping accessible knowledge up-to-date* and in enhancing its reuse. Furthermore, through the formalization of ontologies, semantic communication and

[2] In other words the ontology is specified.

co-operation becomes possible not only among humans, but computers as well, enabling the efficient development and maintenance of knowledge-based systems.

Knowledge plays a vital role both in performing day-to-day activities and in reflecting on these daily routines in all organizations. At the same time, the relevance of knowledge (and/or its elements) may differ even between organizational levels and may also change over time. It is also risky to assume that the right knowledge is naturally at the right place and our knowledge workers have all the necessary knowledge at their disposal all the time. Therefore, the need for effective knowledge management tools that enable the creation, application, reuse and evaluation of knowledge is permanently increasing. In this paper we present our work in designing the *STUDIO ontology-centric knowledge-based system for effective knowledge management and personalized learning*. The ontology-based domain models are at the core of the system as they drive the creation, storage, validation and search for relevant knowledge elements.

In our architecture either business process descriptions or training materials can be applied to identify relevant knowledge elements that have to be maintained and enforced in a logical structure using ontologies. Based on the ontology-centric architecture a repository of knowledge related content—including both learning materials and test questions—have been developed to support the implementation of knowledge assessment and personalized learning applications. Our aim is to provide efficient and flexible knowledge repository functionality for supporting knowledge testing in multiple situations (such as preselection or self-assessment) and provide a mechanism for creating and enriching ontological descriptions from various sources (e.g.: business process descriptions) that enhance the storage, distribution and publishing of stored knowledge in a reusable fashion. This chapter provides a detailed description of the ontology-centric architecture and multiple application scenarios of the STUDIO knowledge-based system.

2 The Architecture of the Ontology-Centric STUDIO System

For any knowledge-based system a number of requirements need to be satisfied in order to enable the development of multiple knowledge-based applications. These requirements are the following:

- **Knowledge representation** languages that are responsible for expressing the structure of the given application have to be selected with care.
- **Knowledge organization** tools that allow for the efficient handling of even large and complex knowledge structures are also necessary.
- **Environments** that enable users to create, maintain and query knowledge are also a must in these systems (Jarke et al. 1989).

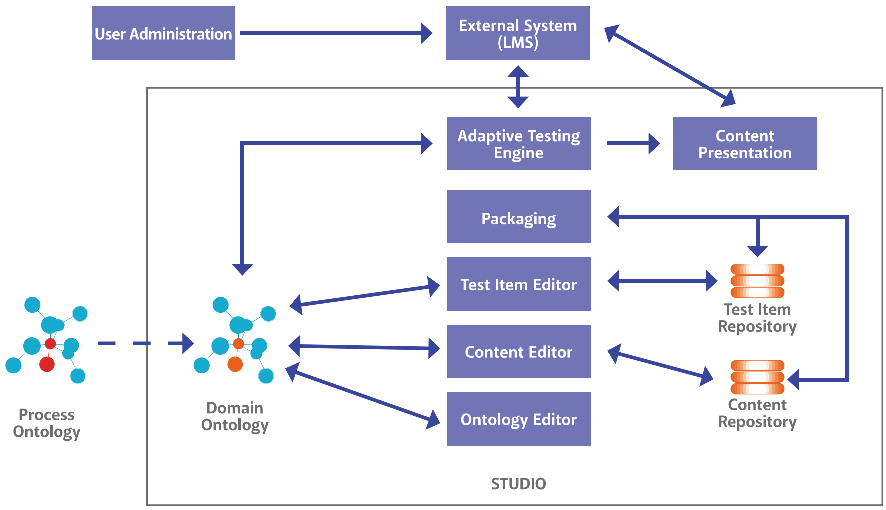

Fig. 1 Architecture of the Ontology-centric STUDIO System

Ontology-centric architecture can satisfy these requirements effectively and efficiently. The most distinguishing characteristic of the STUDIO system is the central role that ontologies play. In our architecture, knowledge is organized in a logical, multi-relational ontology structure, defined either according to business process models or learning materials representing the domain of interest.

The high-level design of this ontology-centric architecture takes a modular approach, as can be seen in Fig. 1. Besides providing the formal description of the domain of interest, the *Domain Ontology* serves as a basis for the *Adaptive Knowledge Testing Engine,* that is the primary application of the system. The structure of content is also determined by the ontology in STUDIO, meaning that every piece of content (a learning material or a test question) is connected to one (and only one) specific concept of the Domain Ontology. Learning materials are stored in the *Content Repository*, while test questions are stored in the *Test Item Repository*. Additionally, editing of the various components (domain ontology concepts, learning materials and test questions) is enabled by the respective modules of the system, namely the *Ontology Editor*, the *Content Editor* and *Test Item Editor*. The *Packaging* component enables power users and/or domain experts to develop customized scenarios for knowledge assessment by selecting certain concepts (and concept trees) from the overall domain ontology that best describe the targeted sub-domain. Finally, the *Content Presentation* module is entitled to present and visualize the stored content pieces (adaptive tests, test results, ontology visualization and learning materials) to the end users. These components will be further discussed in the following sections.

The STUDIO system has been designed to enable the flexible use of its functionality, independently from its form (e.g. workstation- or smart phone-based use). Accordingly STUDIO could be easily integrated with any learning management system that should be responsible for user administration and authentication tasks.

2.1 Meta Model of the Domain Ontology

The wide spectrum of ontology applications clearly proves that both the business and scientific world has acknowledged that the detailed exploration of semantic relations must stand at the focus of exploring organizational knowledge—besides the precise definition of concepts (Corcho & Gómez-Pérez 2000; Gómez-Pérez & Corcho 2002). Ontologies modelling domain specific knowledge can also efficiently enhance the integration of information from different sources.

Ontology Language To effectively support a dynamic conceptual framework, the domain model in the proposed architecture is defined using OWL ontologies (McGuiness & van Harmelen 2004), in which: OWL classes represent such domain concepts that can efficiently support knowledge testing; OWL properties define concept attributes and their relationships; and OWL individuals define concrete domain (such as network management or supply chain management) objects.

Domain Concepts Our approach mainly foresees the following domain concepts and relations: The Knowledge Area class is at the very heart of ontology, representing major parts of a given domain. Each knowledge area may have several sub-knowledge-areas through the *HasSub-knowledgeArea* inclusion relation. Not only inclusion relations, but order relations connecting knowledge areas in a non-hierarchical way are also important as far as knowledge testing is concerned. In the ontology order is described by the *RequiresKnowledgeOf* relation. For example, if KnowledgeArea_1 requires the knowledge of KnowledgeArea_2 and KnowledgeArea_2 requires the knowledge of KnowledgeArea_3, then giving an incorrect answer to any test question related to KnowledgeArea_3 is an indication that there is a lack of knowledge concerning both KnowledgeArea_2 and KnowledgeArea_1.

In order to enable effective knowledge testing, the internal structure of knowledge areas also has to be described in detail. Elements of the Basic concept, Theorem and Example classes form the internal structure of a knowledge area. The *HasPart* inclusion relation connects knowledge areas with their knowledge elements. In order to comprehensively describe the internal structure of the knowledge areas relationship between basic concepts, theorems and examples also have to be identified. The *Premise* and *Conclusion* are order relations that describe basic prerequisites of a theorem (rule or scientific statement) and basic concepts that can be inferred from a theorem. According to CommonKADS[3] methodology such relations have to be presented as option objects in the ontology. These relations are also classes of the ontology that must have special properties, such as the precise description of all attributes on both "ends" of the given relation. While the *RefersTo* reference relation may connect any two individuals of either the Basic Concept or

[3] www.commonkads.org.

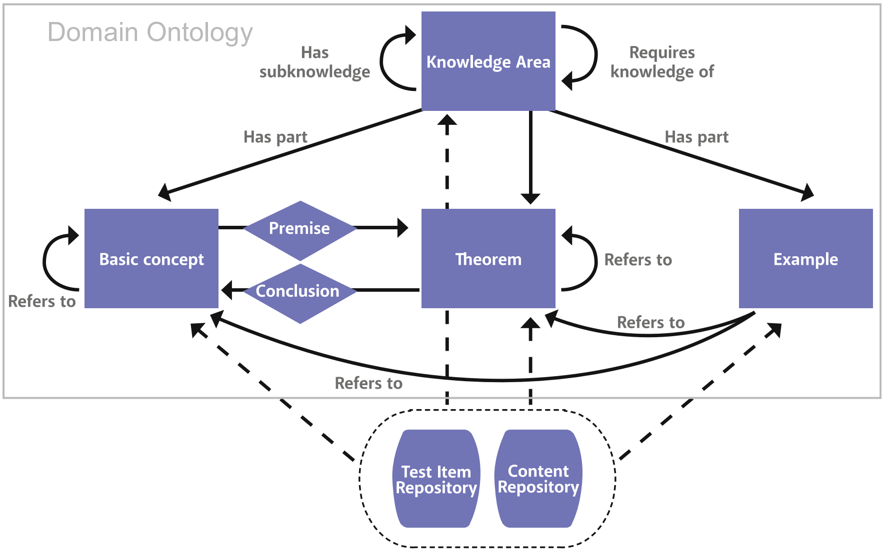

Fig. 2 Meta Model of the Domain Ontology in STUDIO

the Theorem class with each other, individuals from the Example class may also refer to any of the other two knowledge element types. Interrelationships of the major concepts of the domain ontology are shown in Fig. 2, where rectangles represent classes and arrows describe 1-N relations.

The restricted set of relations does not imply limits to knowledge representation but it is a convenient method to improve the computational complexity of the adaptive testing algorithm that has to navigate on the graph provided by the ontology. This knowledge modelling approach is also in accordance with current learning theories. On one hand ontological modelling is nothing other than "connecting specialized information sets and the connections . . . that enable us to learn more" (Siemens 2004, p. 5). On the other hand the ontological model of knowledge areas can contribute to the improvement of certain navigating skills of learners—such as creating inferences and analogies, analyzing pieces of information in various ways and making new connections or distinguishing links between fragments of information to create new relations, etc. (Brown 2006).

2.2 Ontology Engineering Components

A series of approaches have been presented in the literature for building ontologies, such as METHONTOLOGY (Gómez-Pérez et al. 1996), On-To-Knowledge (Staab et al. 2001), or Uschold and King's method (Uschold & King 1995; Uschold 1996) to name but a few. According to several methodologies ontology building is an abstraction process where ontology concepts are extracted from an initial knowledge base. Based on other methodologies ontologies are either built from other ontologies, e.g. by automatically generating an ontology skeleton from a huge ontology (Swartout et al. 1997) or by a process of reengineering them.

The first two phases of ontology development—according to the METHONTOLOGY terminology—are: specification and conceptualization. The goal of specification is to determine why the ontology is built and what its intended use is, while in the conceptualization phase the informally perceived view of a domain is converted into a semi-formal specification by identifying the most important concepts of the domain and their relations. These phases are followed by the formalization and maintenance activities. Ontologies built in STUDIO have been used both in education and in business settings for different purposes. The conceptualization phase of building ontologies in STUDIO also varied taking into consideration expectations determined in the specification case.

Building domain ontologies from text is typically used in educational situations in STUDIO where the primary goal of ontology building is to support either knowledge assessment or provide personalized learning experience based on the results of knowledge assessment and the automatic detection of learning styles. In these cases the development of the semi-formal specification is a result of collaboration between the domain expert and the ontology engineer, where curricula, lecture notes and related literature are used as the initial "knowledge base". However, in the case of on the job training, these resources are not available or they do not accurately represent business requirements stemming from corporate processes. Chapter "Ontology Tailoring for Job Role Knowledge" presents a methodology how to extract task specific knowledge from corporate process models and map the extracted concepts into an ontology structure using domain ontologies of STUDIO—if available—as a base. Test mining tools are applied for extracting task related knowledge elements from process models and related documentations.

Building domain ontologies from other ontologies is more typically used in business situations where process ontologies are already available or have to be built to support further applications. Chapter "Corporate Semantic Business Process Management" discusses a semi-automatic, but well-controlled way of enriching domain ontologies using process ontologies. The presented approach describes how to transform the business process into a process ontology and combine it with the knowledge base that is a domain ontology. At the same time Chapter "Future Development: Towards Semantic Compliance Checking" presents how ontology matching tools could be applied in investigating business processes and improving available process ontologies.

Ontology Editor Tools or any other technology enabled tools—in theory—are not required for ontology development, not even in the case of applying the above described meta-model. At the same time the application of ontology editing tools can significantly facilitate the ontology engineering process. These tools were used to build ontologies with ease even without the detailed knowledge or direct application of formalization languages. Moreover, managing a high number of ontology elements, relations, axioms and constraints is also a challenge without adequate computerized aid.

Seidenberg and Rector (2007) have also revealed that there is a need for user-friendly tools able to support collaborative ontology construction and the arrangement of single-users' asynchronous tasks. Usually, there are two major ways in which ontology editing tools support collaborative ontology engineering (Noy 2007). One way is the so called synchronous mode when every user accesses the same version of the ontology and changes are immediately visible to everyone. At the same time in the asynchronous mode users work on their own sandbox space and integrate their changes with the master version later. In some cases editing tools use mixed approaches. Every approach has its own set of advantages and disadvantages.

Besides the above described concerns the *Ontology Editor* in STUDIO also has to meet the following requirements:

- *Extensible*—Due to rapid economic and technological changes business processes and underlying knowledge structures and elements also evolve over time. Accordingly, such a tool is required that would enable the maintenance and development of domain ontologies in an easy but consistent manner.
- *Capable of treating high volume data*—Even one business process or one related curriculum may consist of several hundreds of concepts that must be presented in the ontology. Modelling all the business process-related knowledge elements or all the curricula of a training program will require substantial capacities.
- *Interoperable*—Ontologies may provide a base for different applications, in this way ontology editing tools must be prepared in order to ensure communication and collaboration with other tools in the system.
- User friendly—A simple but consistent interface helps users to work faster and more effectively. Such a tool needs to be developed that besides editing concepts, relations and other properties in a simple way can also provide an easy to understand visualization of the domain ontology. (Szabó 2006).

The STUDIO Ontology Editor follows a mixed approach and provides techniques for both synchronous and asynchronous ontology engineering. A primarily synchronous mode is applied when all the changes made on the ontology are immediately visible for the users as a draft version. Only power users have the right to save the modified ontology as a next version, after checking consistency. There is also an opportunity to use a sandbox space for ontology development.

The editor is also special in the respect that only concepts and relations identified in the above described domain ontology meta-model could be applied in the course of editing. The aim of applying 'built-in' classes and relations was to provide the kind of tool that can be used by domain experts with few or no competencies in ontology engineering. Consequently domain experts can interact with a user-friendly interface where graphical presentation of the ontology also enhances user experience, as shown on Fig. 3.

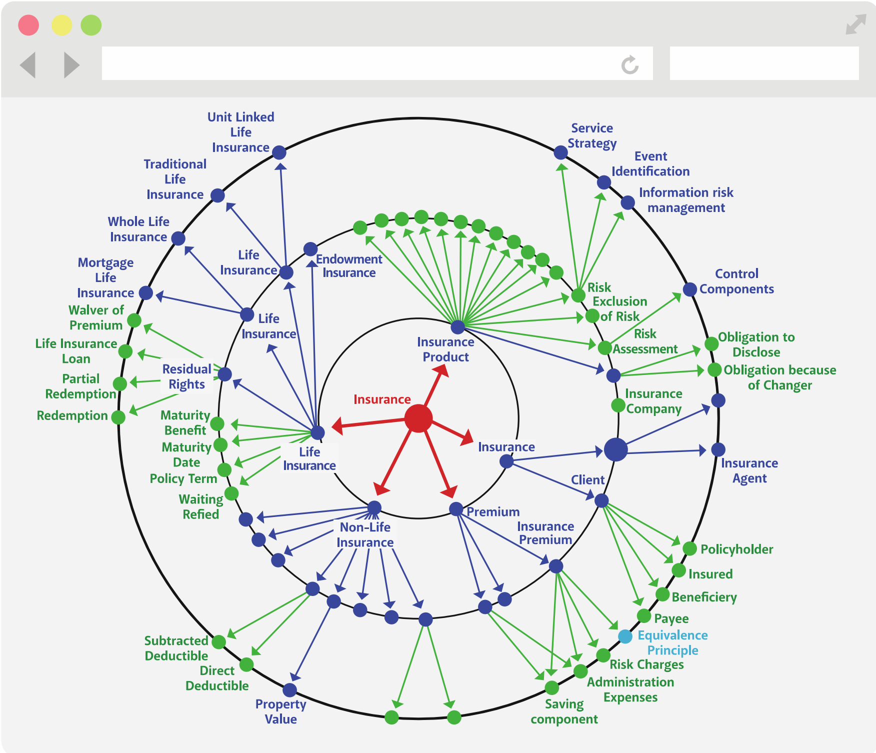

Fig. 3 Ontology visualization in STUDIO—insurance domain ontology

2.3 Content Management Components

The ontology-based domain model is at the core of the STUDIO system as it drives—besides knowledge testing—the creation, storage, query and search for all domain related content as well. Our aim was to provide efficient and flexible content management functionality in STUDIO and to provide a mechanism for developing and maintaining learning materials and test questions in a structured and reusable fashion. The ontology provides the base structure as each single piece of content is connected to one and only one concept of the ontology.

Learning content development starts with the construction of the appropriate domain ontology. As the ontology is finalized, domain experts extend the bare structure with learning materials. Since the structure has already been determined by the domain ontology, the "only" task of the content developer is to assign content elements to the adequate nodes of the ontology. Content elements may have many different formats: images, articles, short texts such as a useful paragraph or a famous quote, audio files or video materials. In order to effectively support learning and knowledge gap fulfillment, learning materials have to be created in such a way that they will also adapt to different learning styles. Visual learners may

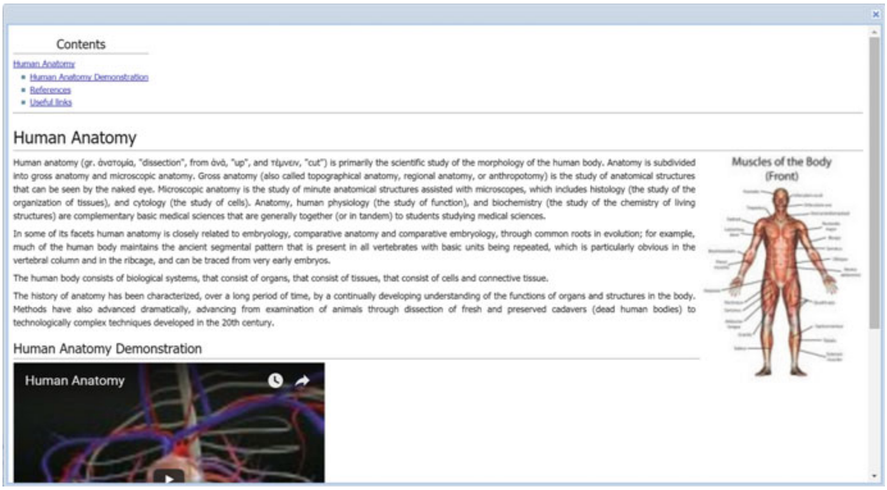

Fig. 4 Learning content in STUDIO

prefer diagrams and presentations, while verbal learners may choose text and lecture notes instead.

The *Content Editor* of STUDIO is a deployment of the Semantic MediaWiki[4] platform that is an extension to the popular MediaWiki engine providing several tools and the special wiki-notation functionality in order to enable the application of ontologies in multiple ways. One of the advantages of using MediaWiki is that it supports multiple types of contents, including text and various multimedia objects broadly used in the learning contents of STUDIO. For Wiki page authors a detailed data formatting and inclusion guideline has also been created, with prewritten html codes. Even if the content developer doesn't have relevant html knowledge it is possible to embed rich media content by simply using a copy-paste mechanism.

The *Content Repository* is responsible for storing and managing these wiki content elements (See an example of wiki content on Fig. 4) and maintaining a rich set of metadata describing them. Each content element can be described with Dublin Core metadata (ISO 2009) and other useful descriptors, like tags or categories. This rich description enables domain experts to easily search the repository for and retrieve already existing contents or create and categorize new elements if needed.

Test item development is crucial in regard to the knowledge testing. In order to adequately support the ontology-based adaptive knowledge testing application every test item must be connected to one and only one concept in the ontology.

[4] https://semantic-mediawiki.org/.

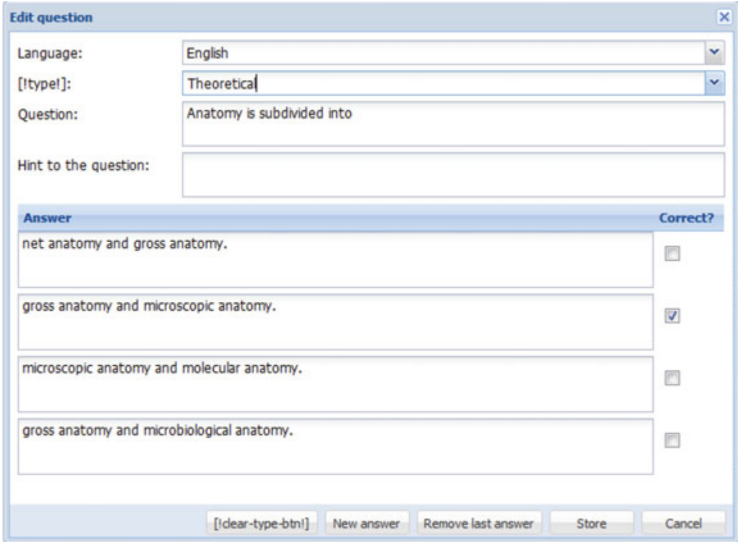

Fig. 5 Test item editor in STUDIO

On the other hand each ontology concept may have several related test questions. In this way the *Test Item Repository* is also structured by the domain ontology. At the same time the Test Item Repository does not form an integral part of the ontology.

Test items are provided in the form of multiple-choice questions. Therefore each test item consists of a question, one correct answer and three false answers. Test item editing and translation into multiple languages is enabled by the Test Item Editor (Fig. 5).

Finally test questions are packaged and deployed in the Adaptive Testing Engine that provides the necessary facilities to execute and evaluate knowledge tests. The ontology is an integral part of the test package, since the execution of tests heavily relies on the underlying ontology structure.

2.4 Packaging Component

The main goal of any application in STUDIO is to create a task or target specific structure extracted from the domain ontology. Accordingly the last phase of content development is packaging, meaning the creation of a set of standard packages that contains the extracted ontology structure of the target (or task-specific) sub-domain, as well as the related learning contents and tests questions. The content package is deployed into the learning management system, while the test package is deployed in the Adaptive Testing Engine.

These target specific structures called *Concept Groups* in STUDIO provide a new layer in the underlying domain ontology. A Concept Group consists of a set of

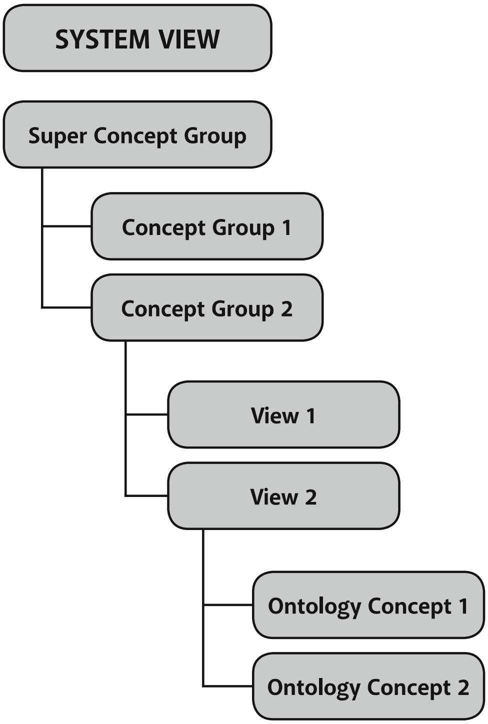

Fig. 6 Concept group representation logic in STUDIO

ontology concepts extracted from the domain ontology, and rearranged based on the target of use. The rearrangement does not have any effect on the semantics provided by the domain ontology. The major role of the Concept Group is to enable customization and/or mapping of the domain ontology to the target use case and to provide the basis for adaptive testing.

Figure 6 presents how the Concept Group hierarchy could be built in the STUDIO, where Super Concept Group may represent the organization, a position in the organization or even a training program of an educational institution. The Concept Group represents the target area that should be tested. A Concept group may embody a specific process, a job role or a training specialization. Views provide an entry point to the ontology representing a task or a specific course. Chapter "Ontology Tailoring for Job Role Knowledge" provides an in depth description of the representation logic and possible use cases of Concept Groups in STUDIO.

2.5 Adaptive Testing Engine

Measuring knowledge in a reliable way has always represented a major challenge in training and education. From the 1970s the emerging field of Computerized

Adaptive Testing (CAT) provided important results about adaptive systems that can be combined with semantic technologies (ontologies). In contrast with the traditional examination the number of test items and the order of questions in an adaptive test is only defined in the course of testing with the goal of determining the knowledge level of the test candidate as precisely as possible with as low a number of questions as possible (Linacre, 2000). More precisely, as the test candidates answer the test items, the test "adapts" itself by selecting the next test item to be presented on the basis of performance on preceding items. Adaptive testing is not a new methodology and despite the fact that it has many advantages compared to traditional testing, its application is not widespread. Adaptive tests are usually computer-based tests that have the following main characteristics, independent of the applied testing methodology:

- The test can be taken at a time convenient to the examinee; there is no need for mass or group-administered testing, thus saving on physical space.
- As each test is tailored to an examinee, no two tests need be identical for any two examinees, which minimizes the possibility of copying.
- Questions are presented on a computer screen one at a time.
- Once an examinee keys in and confirms his answer, (s)he is not able to change it.
- The examinee is not allowed to skip questions nor is (s)he allowed to return to a question which (s)he has confirmed his/her answer to previously.
- The examinee must answer the current question in order to proceed onto the next one.
- The selection of each question and the decision to stop the test are dynamically controlled by the answers of the examinee (Thissen & Mislevy 1990).

The current research focuses on the elaboration of such knowledge assessment methodology that enables the exploration of a test candidate's knowledge gaps in order to help them by complementing their training or educational deficiencies. Accordingly, the *Adaptive Test Engine* is a key application in STUDIO that exploits the advantages of ontological descriptions of the domain of interest. As described in Sect. 2.3 every test item resides in the Test Item Repository and is connected to one specific concept in the ontology. In the course of testing the Adaptive Testing Engine "walks through" the ontology structure and asks questions concerning each affected ontology concept. In this way the test candidate's knowledge of a certain set of concepts can be evaluated.

The testing procedure starts the examination at the top of the hierarchy, meaning that those concepts are tested first that have no parent concepts in the given sub-domain (called Concept Group—See Sect. 2.4 for further details). This means that testing typically starts with the evaluation of concepts from the Knowledge Area class. Accordingly, the adaptive test engine provides a testlet[5] related to each top level knowledge area including as many questions that cover the given concept.

[5] A testlet is a cluster of test items that share a common path, scenario, or other context.

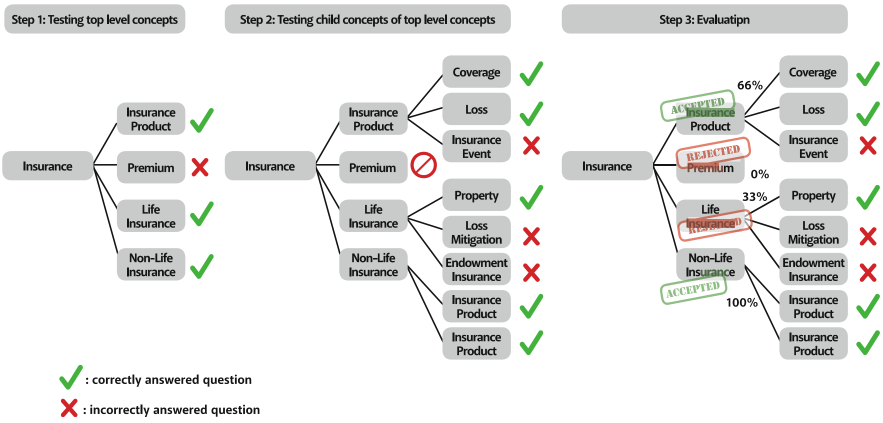

Fig. 7 Illustration of the top-down adaptive testing methodology in STUDIO

For example, if the test candidate was able to correctly answer question (s) concerning the KnowledgeArea_1 in the next stage questions concerning every sub-knowledge area and/or basic concept of the KnowledgeArea_1 will be presented to the examinee. If more than 50 % of all questions[6] (indirectly connected to the KnowledgeArea_1) are incorrectly answered then the KnowledgeArea_1 and its sub-areas will be not accepted. At the same time if certain sub-knowledge area related questions are answered correctly then sub-knowledge areas are tested in the previous manner. In other words the testing engine executes a depth first graph search algorithm in such manner that it closes a branch if the test candidate does not know the given knowledge area or its sub-knowledge-areas and/or given basic concepts at an adequate level. As a result the test candidate's knowledge is thoroughly assessed in respect to the target domain or sub-domain. Figure 7 illustrates the process of adaptive knowledge testing in STUDIO.

Naturally this is not the one and only way in which ontological descriptions could be applied for knowledge assessment. One limitation of the above described adaptive testing methodology is that the test may stop at an early stage, (e.g. in an extreme case, if the KnowledgeArea_1 is the only top level concept in the given Concept Group and the test candidate fails to answer the related question, the test stops and no more questions are presented) which may discourage the test candidate on the one hand while also preventing an insightful exploration of the knowledge structure. For that very reason another knowledge evaluation methodology has also been implemented in STUDIO that follows a bottom-up approach in contrast with the top-down approach of the above described methodology. The bottom-up approach—instead of evaluating single concepts—focuses on analyzing whole assessment paths (that connects a certain concept of the domain with concepts on

[6] The examinee can set a threshold according to the objectives of the test to be taken. The selected threshold is automatically applied by the test evaluation algorithm.

the top of the hierarchy). In the first stage of the testing procedure assessment paths are determined, then paths are assessed from the bottom to the top element. If any element in the given path fails, the related ontology concept will be marked as failed that also blocks the current path to the start-element. This failed concept will also block every other path including this element and as such minimizes the set of future sub-paths to assess. The system accepts every path of concepts which reaches the start-element through the relations provided. Incorrectly answered questions and the relating failed concept essentially splits a path into a "top" part which still could reach the start-element and a bottom part which won't be evaluated for the final result.

Chapter "STUDIO: A Solution on Adaptive Testing" provides a thorough review of the theoretical background and organizational relevance of adaptive testing and also presents a detailed description of the adaptive testing methodologies in STUDIO.

2.6 Automatic Learning Style Detection Application

Different people may prefer different ways to learn, or in other words, different learning styles determining how they process and come understand new knowledge. It also has to be taken into consideration that with similar knowledge tested, people may perform differently, depending on the format and focus of the test questions. Truong (2015), through a systematic literature review, indicated that to develop an automatic learning style detection system, a number of stages are required including:

- Learning styles framework selection
- Learning styles attributes selection
- Classification algorithm developments and evaluations

In STUDIO the Felder-Silverman's (1988) learning styles theories have been applied as the framework for the automatic learning style detection application. In the course of STUDIO development, a systematic review was also carried out, which resulted in over 80 potential learning styles predictors. These variables have been being tested, evaluated and engineered. The initial variable selection then becomes the input for the detection model development and evaluation. All of these results, as a consequence, are integrated into STUDIO in the following way: in the first stage, variables from several sources are collected and fed into a data-integration and -processing unit. The output, in the second stage, is used as input for the learning styles detection model, which classifies student' learning styles accordingly. Finally, the information of learning styles of individuals is used as input for a recommendation unit that aids the adaptive functions of the system.

2.7 Content Presentation Component

Feedback plays an essential role in knowledge assessment either by providing advice and recommendation on individual opportunities for improvement or by inspiring motivation. It is also approved in literature that feedback should be personalized rather than using one, general feedback for all. In STUDIO the *Content Presentation* component (See Fig. 1) is responsible for delivering customized materials for end users. Besides presenting the target specific content packages for tutors, domain experts and other power users—created in the Packaging module of STUDIO—that contains the extracted ontology structure, the related test questions and learning contents, the Content Presentation module also provides test results and evaluations, personalized learning content and the statistical analysis of former activities and performance for end users as well.

Learning Material Adaptation plays a key role in enhancing personalized learning experience. The adaptive testing methodology applied in STUDIO enable the repeated identification and fulfilment of knowledge gaps, in order to be able to provide personalised guidance on how the identified knowledge gaps can be effectively eliminated. Monitoring of learning styles are also crucial in developing personalised learning materials and learning activities for end users (test candidates). For visual learners, for instance, diagrams and presentations etc., can be provided, while for verbal learners, texts and lecture notes can be suggested. At the same time this learning style dependent adaptation is still under development in STUDIO since the number of alternative learning materials—supporting every style—still has to be increased.

Evaluation of test results and statistical analysis in STUDIO provides different approaches to follow users' activities and performance. These statistics help end users in making progress towards achieving their learning goals and also help content developers assess the created learning content. Currently, the following functionality is available

- Test evaluation—After each completed test the results are presented and explained in detail, and access is provided to related learning materials.
- User activity analysis—Content developers have the right to analyze how many times test candidates accessed the system, how many tests they have started and how many of them were suspended and/or finished, which questions were included in the test, what the answers to these questions were, etc.
- Data exportation—Content developers can also use the built-in query language of STUDIO to write customized queries for exporting data in a comma-separated value file format (CSV file) for further processing. The query language in STUDIO is based on SQL, accordingly a basic knowledge of SQL is required.
- Connection to external systems—Statistics components of STUDIO could be made available for external systems too. In this case an external system should call this component of STUDIO with an HTTP request, which contains a query. Results can be provided in different formats. In this way results and user

activities can be accessed in the external system as well, without manually exporting data from STUDIO.

STUDIO provides a systematic solution for both controlled knowledge assessment and personalized self-assessment, using a domain ontology to capture the various areas of education providing feedback in multiple ways and in a user-friendly manner.

3 Application Scenarios of STUDIO

In the ontology-centric STUDIO system the behaviors of domain concepts are identified completely using ontological entities, around which different knowledge management tasks could be carried out. Semantic technologies and the underlying applications offered by STUDIO are domain independent, and in this way application scenarios could be elaborated in respect to both business and education. The first scenario was situated in education, while further scenarios were deployed in the Organizational Knowledge Management and Human Resource domains.

3.1 The Educational Setting

Competition in e-learning solutions is increasing at an alarming rate, while social and economic changes and the expectations of both students and the labour market are frequent and diverse. Therefore, there is a great deal of pressure on educational institutions to turn towards the development and application of innovative and modern technologies that enable students to easily access, understand and apply complex curricula and other teaching materials. STUDIO can support education in several ways.

Consensus-based Knowledge Structures are essential in improving interaction among teachers and students. The proposed ontology model (See Sect. 2.1) enables educational institutions to create a comprehensive, unambiguous description of each curricula, or training program of the institution. The resulting domain ontologies are ready to be deployed in managing and improving the educational portfolio and teaching contents of the institution and in enhancing spontaneous learning of students and better understanding of learning materials and their interrelations.

Knowledge Assessment and measuring knowledge in a reliable way is an evergreen issue in education. In order to measure how much students have learned, it is not enough to assess their knowledge at the end of the course. Teachers also have to find out what students know when starting a course. Identifying the prior knowledge of students makes it possible to more precisely identify the knowledge students have gained during the course or training program. The *Adaptive Testing Engine* of STUDIO could be applied both for prior and subsequent knowledge testing. *Concept Groups*—determining the target sub-domain and test package—

can be set up to enable direct evaluation of students' knowledge and performance before and after the given course.

The adaptive testing methodology of STUDIO can also support *self-assessment* providing students with the opportunity to make adjustments to their progress prior to graded evaluation. Taking adaptive tests on their own, students can receive comprehensive feedback on their knowledge gaps. More precisely, a detailed list of those ontology concepts will be provided where the student may have deficiencies according to the test results.

Personalized Learning Experience will only be appropriate if besides supporting what students wish to learn it is also determined how they should learn it. To enable personalized learning, STUDIO system—making use of sthe domain ontology and adaptive test engine—can compile and re-compile self-assessment lessons with personalized sets of learning materials and assessment questions. In the first stage the student's knowledge in respect to the selected domain or sub-domain has to be tested and evaluated in order to identify those (ontology) concepts where the test candidate has deficiencies. Based on the result of this knowledge test a set of personalized learning materials is provided with guidelines on how the learner should "walk through" the ontology structure. In other words, access is provided to the learning material of those ontology concepts where the students incorrectly answered the related test question. Since the results are represented using the ontology visualization tool of STUDIO (See Fig. 3) not only concepts but also their interdependencies are presented to define the proposed paths of learning. The learning experience could be further enhanced—making use of the Automatic Learning Style Detection Application—by adapting learning materials to the learning style of the user.

3.2 The Business Setting

Knowledge acquisition, creation, and transferring together with its sharing have always been a challenge for organizations. It is dangerous to assume that the available knowledge is the right knowledge and it is in the right place. Moreover, the relevance of knowledge may also differ between organizational levels and may also change over time. STUDIO can help organizations to overcome these challenges and to use and reuse organizational knowledge in multiple ways by combining its tools with semantic process modelling techniques (Chapter "Corporate Knowledge Discovery and Organizational Learning: The Role, Importance, and Application of Semantic Business Process Management—The ProKEX Case" provides an overview of the process modelling approach and the related ProKEX solution).

On the Job Training has the benefit of providing direct knowledge and experience for the employee under real working conditions. At the same time this kind of training could be costly since work activities are interrupted by training activities causing delays as well as increasing the number of mistakes. The ProKEX solution—which also makes use of STUDIO's functionality—enables the organization to extract knowledge from organizational processes in order to enrich the organizational knowledge base. This will provide the basis of online, on the job training that allow employees to easily acquire their job role specific knowledge in a customized and efficient manner. More precisely, using the ProKEX toolkit, the domain ontology in STUDIO—representing organizational knowledge—can be improved either by directly extracting business process-related knowledge applying text mining techniques (See Chapter “Ontology Tailoring for Job Role Knowledge”) or by matching process ontologies—formally representing knowledge embedded in business processes—with the domain ontology (as also indicated on Fig. 1 and detailed in Chapter “Corporate Semantic Business Process Management”).

By using adaptive knowledge tests that are based on the enriched domain ontology the employee’s knowledge gaps can be identified, mapped with job role related requirements and addressed with appropriate learning objects. Upon completing an assessment, a knowledge gap report is produced for the test candidate by comparing the knowledge of the employee with organizational requirements. In the event of a discrepancy, the STUDIO system provides the employee with a personalized learning path so that (s)he may improve his/her proficiency level.

Allocating Human Resources is difficult and often fraught with problems despite the fact that there are numerous methods for both short- and long-term resource allocation. At the same time, in most cases implementation issues are not addressed in the literature or the proposed implementation solutions heavily rely on managers’ expertise lacking detachment. By using the adaptive testing solution of STUDIO, the knowledge of each worker could be compared with knowledge required by business processes providing an objective basis for matching resource claims with resource offers. As a result, upgraded management of corporate intellectual capital and a better return on investment in human capital can be expected that will lead to more efficient execution of processes and higher improvement in revenues.

Preselection aims at screening suitable applicants where the majority of applicants are eliminated in order to leave only those people most likely to be selected. There are several strategies and tools for preselection (such as Résumés, letters of application, test results etc.) but in any case, job specification and description should form the basis of the applied strategy. Evidently, process models and knowledge extracted from these models provides an objective and complete description of job role related requirements. Accordingly the STUDIO toolkit can provide a knowledge gap analysis of applicants, also enabling mapping test results to current and valid job roles. In the course of preselecting suitable applicants it is important to be unprejudiced and tolerant about the potential each applicant has to be successful in the job. Any specific knowledge gap identified by adaptive tests can be noted and raised during the interview and individually customized learning content can be provided in STUDIO for the applicant if selected for the position.

4 Conclusion

In this work our contribution to organizational knowledge management is threefold: firstly the proposal of the ontology-centric architecture for developing an extensible knowledge-based system to support the use and reuse of organizational knowledge; secondly the development of a meta-model of the ontology that defines fundamental concepts in a domain independent way; thirdly the development of ontology-based applications to support adaptive knowledge testing, automatic learning style detection, personalized learning both in an educational and business context and human resource allocation and preselection. Further improvements have also been elaborated, designed and prototyped in the context of the ProKEX Project including the application of text mining techniques to enrich domain ontologies (See Chapter "Ontology Tailoring for Job Role Knowledge"), semantic ontology matching (See Chapter "Future Development: Towards Semantic Compliance Checking") and semantic process modelling methods (See Chapter "Corporate Semantic Business Process Management").

Following the completion of several successful pilots, the STUDIO ontology-centric knowledge-based system is being used on a regular basis providing a solid base for maturing the following concepts: (1) Knowledge workers, tutors or teachers cannot be forced to have ontology engineering competencies. Accordingly, a user-friendly ontology editing tool has been developed with a built-in meta-model of ontology. (2) Exploiting the potentials of personalized learning requires the development of alternative knowledge testing methodologies to fit different requirements and the application of learning style detection methods. (3) In order to enable the reuse of organizational knowledge taken into consideration its evolution, as well as knowledge embedded in business process also have to be built into the organizational knowledge base. Therefore, semantic techniques for enriching an organizational knowledge base with process-related knowledge have been developed.

Future works will consist of ontology validation and testing activities in order to improve the application of semantic technologies both in knowledge management and e-learning.

References

Akerkar, R. A., & Sajja, P. S. (2009). *Knowledge-based systems*. Sudbury, MA: Jones & Bartlett.

Brown, T. H. (2006). Beyond constructivism: Navigationism in the knowledge era. *On the Horizon, 14*(3), 108–120.

Corcho, O., Fernández-López, M., & Gómez-Pérez, A. (2003). Methodologies, tools and languages for building ontologies. Where is their meeting point? *Data & Knowledge Engineering, 46*, 41–64.

Corcho, O., Gómez-Pérez, A. (2000). Evaluating knowledge representation and reasoning capabilities of ontology specification languages. In: *Proceedings of the ECAI 2000 Workshop on Applications of Ontologies and Problem-Solving Methods, Berlin, 2000.*

Felder, R. M., & Silverman, L. K. (1988). Learning and teaching styles in engineering education. *Engineering Education, 78*(7), 674–681.

Gómez-Pérez, A., & Corcho, O. (2002). Ontology languages for the Semantic Web. *IEEE Intelligent Systems, 17*(1), 54–60.

Gómez-Pérez, A., Fernández-López, M., De Vicente, A. J. (1996). Towards a method to conceptualize domain ontologies. In: *Proceedings of ECAI-96 Workshop on Ontological Engineering. Budapest, 13 Aug 1996.*

Gruber, T. R. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition, 5*(2), 199–220.

Hendriks, P. H., & Vriens, D. J. (1999). Knowledge-based systems and knowledge management: Friends or foes? *Information and Management, 35*, 113–125.

ISO (2009). ISO 15836:2009(E): Information and documentation—The Dublin Core metadata element set. https://www.iso.org/obp/ui/#iso:std:iso:15836:ed-2:v1:en. Accessed 10 Jun 2015.

Jarke, M., Neumann, B., Vassiliou, Y., & Wahlster, W. (1989). KBMS requirements of knowledge-based systems. In J. W. Schmidt & C. Thanos (Eds.), *Foundations of knowledge base management* (pp. 381–394). Berlin, Heidelberg: Springer.

Laudon, K. C., & Laudon, J. P. (1997). *Management information systems, organization and technology* (5th ed.). New Jersey, NY: Prentice Hall.

Linacre, J. M. (2000). Computer-adaptive testing: A methodology whose time has come. In S. Chae, U. Kang, E. Jeon, & J. M. Linacre (Eds.), *Development of computerized middle school achievement tests, MESA Research Memorandum No. 69*. Seoul, South Korea: Komesa.

Lucas, P., & Van der Gaag, L. (1991). *Principles of expert systems*. Wokingham, UK: Addison-Wesley.

McGuiness, D., van Harmelen, F. (eds) (2004). *OWL Web ontology language overview, W3C recommendation*. http://www.w3.org/TR/owl-features/. Accessed 21 May 2015.

Noy, N. (2007). What users want: Collaborative development of ontologies. In: *Position Papers of SemGrail 2007 Workshop*. Redmond, 21–22 Jun 2007.

Seidenberg, J., Rector, A. (2007). The state of multi-user ontology engineering. In: *Proceedings of 2nd International Workshop on Modular Ontologies. Whistler, Canada, 28 Oct 2007.*

Siemens, G. (2004). *Connectivsm: A learning theory for the digital age*. http://www.elearnspace.org/Articles/connectivism.htm. Accessed 15 May 2015.

Staab, S., Studer, R., Schnurr, H. P., & Sure, Y. (2001). Knowledge processes and ontologies. *IEEE Intelligent Systems, 16*(1), 26–34.

Swartout, W. R., Patil, R., Knight, K., Russ, T. (1997). Towards distributed use of large-scale ontologies. In: *AAAI-97 Spring Symposium on Ontological Engineering*, Stanford University, Stanford 1997.

Szabó, I. (2006). Implementation of the educational ontology. In: *Proceedings of the The 7th European Conference on Knowledge Management*. Corvinus University of Budapest, Hungary, 4–5 Sept 2006.

Thissen, D., & Mislevy, R. J. (1990). Testing algorithms. In H. Wainer (Ed.), *Computerized adaptive testing: A primer* (pp. 103–135). New Jersey: Lawrence Erlbaum Associates.

Truong, M. H. (2016). Integrating learning styles and adaptive e-learning system: Current developments, problems and opportunities. *Computers in Human Behavior, 55*(Pt. B), 1185–1193.

Uschold, M. (1996). Converting an informal ontology into Ontolingua: Some experiences, ECAI-96 Workshop on Ontological Engineering. Budapest, 13 Aug 1996.

Uschold, M., King, M. (1995). Towards a methodology for building ontologies. In: *Proceedings of IJCAI-95 Workshop on Basic Ontological Issues in Knowledge Sharing, Montreal, Canada, 1996.*