RAPID ONTOLOGY DEVELOPMENT MODEL BASED ON BUSINESS RULES MANAGEMENT APPROACH FOR THE USE IN BUSINESS APPLICATIONS

Paper for Doctoral Consortium

Dejan Lavbič, Marjan Krisper

Faculty of Computer and Information Systems, University of Ljubljana, Tržaška 25, 1000 Ljubljana, Slovenia

Dejan.Lavbic@fri.uni-lj.si, Marjan.Krisper@fri.uni-lj.si

Keywords: ontology development, business rules, Semantic Web, rapid application development

Abstract: Ontologies as means for knowledge manipulation in IT have gained its popularity in recent years. The scenarios of successful implementation can mainly be found in the World Wide Web domain and within academia while there are only a few in business environment. This paper introduces Rapid Ontology Development model and accompanying intelliOnto support tool to facilitate ontology construction for inclusion in business applications. Emphasis is given to simplification of the development process of functional components by bridging the gap between formal syntax of captured knowledge and acquisition of knowledge in semi formal way. The primary steps of the process therefore adopt informal modelling methods such as mind maps approach with several transformations and interfaces introduced. That enables business users to manipulate the ontology without any detailed technical knowledge in building ontology with higher semantic expressiveness. While majority of existing approaches end with successful construction of ontology in this approach steps for deployment of developed ontology in a form of functional component and redeployment with versioning is foreseen and supported. Verification of the model will be presented with running examples from the domain of financial portfolio management and organisation of a rent-a-car etc.

1 INTRODUCTION

The aim of this doctoral dissertation is to define a model for rapid ontology development for the use in business applications. The research incorporates several research areas – knowledge management (du Plessis, 2005), ontologies (Abou-Zeid, 2003; Davies, Studer, & Warren, 2006) and business rules approach (Bajec & Krisper, 2005; Dorsey, 2002) in information systems development.

1.1 Research problem and objectives

Nowadays applications based on ontologies and Semantic Web technologies are unfortunately limited to academic environment, while there is still no wide adoption in industry. From the words of W. Edwards Deming "If you can't describe what you are doing as a process, you don't know what you're doing" we can conclude that a part of the problem lies in existing methodologies. Existence of verified procedures is a good indicator of maturity.

The simplicity of using approaches for ontology construction and accompanying tool support is another issue, which also needs a lot of attention and further work. The use of business rules management approach seems like an appropriate way to simplification of development and use of ontologies in business applications. Thus this work will define a Rapid Ontology Development model based on Business Rules management approach and we will tackle the following research questions:

- What are suitable ontology based approaches for rapid development of functional components in business applications?
- Can the use of Semantic Web technologies decrease the required level of user expertise in defining problem domain and capturing knowledge?
- How to improve the process of ontology manipulation with business rules management

approach for the purpose of information system development?

- How to support the use of various data sources in building business vocabulary?

1.2 Expected outcome

After completing the research the following outcomes are expected:

- The definition of Rapid Ontology Development model and improved ontology construction process with business rules management approach,
- Guidelines for tool support and prototype to aid constructing ontologies and the use of ontology based components in business applications.

The rest of the paper is structured as follows. In the next section state of the art is presented with the review of existing methodologies and approaches. Furthermore some details about research methodology are given by presenting phases and results that will result in completing each step. Elements of Rapid Ontology Development model are depicted in section 4, while further information concerning tool support for the process of ontology construction and using in business applications is given in section 5. Finally in section 6 current stage of research is outlined and conclusions with future work are given.

2 STATE OF THE ART

Ontology is a vocabulary that is used for describing and presentation of a domain and also the meaning of that vocabulary. The definition of ontology can be highlighted from several aspects. From **taxonomy** (Corcho, Fernandez-Lopez, & Gomez-Perez, 2003; SanJuan & Ibekwe-SanJuan, 2006; Veale, 2006) as knowledge with minimal hierarchical structure, **vocabulary** (Bechhofer & Goble, 2001; Miller, 1995) with words and synonyms, **topic maps** (Dong & Li, 2004; Park & Hunting, 2002) with the support of traversing through large amount of data, **conceptual model** (Jovanović & Gašević, 2005; Mylopoulos, 1998) that emphasizes more complex knowledge and **logic theory** (Corcho et al., 2003; Waterson & Preece, 1999) with very complex and consistent knowledge.

Ontologies are used for various purposes (Brambilla, Celino, Ceri, & Cerizza, 2006; Dahlgen, 1995; Davies et al., 2006; Heflin & Hendler, 2000; Rao, Dimitrov, Hofmann, & Sadeh, 2006) such as natural language processing, knowledge management, information extraction, intelligent search engines, digital libraries, business process modelling etc. While the use of ontologies was primarily in the domain of academia, situation now improves with the advent of several methodologies for ontology manipulation.

Existing methodologies for ontology development in general try to define the activities for ontology management, activities for ontology development and support activities. More detailed insight into wide spectrum of methodologies can be found in (Davies et al., 2006; Gomez-Perez, Fernandez-Lopez, & Corcho, 2003; Sure, 2003), whilst here only the most representative are depicted.

CommonKADS (Schreiber et al., 1999) is in fact not a methodology for ontology development, but is focused towards knowledge management in information systems with analysis, design and implementation of knowledge. CommonKADS puts an emphasis to early stages of software development for knowledge management. **DOGMA** (Jarrar & Meersman, 2002) is a methodology based on database approach and decomposes ontological resources into ontology bases. **Enterprise Ontology** (Uschold & King, 1995) recommends three simple steps: definition of intention; capturing concepts, mutual relation and expressions based on concepts and relations; persisting ontology in one of the languages. This methodology is the groundwork for many other approaches and is also used in several ontology editors. **KACTUS** (Bernaras, Laresgoiti, & Corera, 1996) approach requires an existing knowledge base for ontology development and is appropriate for bottom up strategy. **METHONTOLOGY** (Fernandez-Lopez, Gomez-Perez, Sierra, & Sierra, 1999) is a methodology for ontology creating from scratch or by reusing existing ontologies. The framework enables building ontology at conceptual level and this approach is very close to prototyping. **SENSUS** (Swartout, Ramesh, Knight, & Russ, 1997) is a very specialized approach for manipulation of large ontologies, while it does not cover entire cycle of ontology development. Another approach is **TOVE** (Uschold & Grueninger, 1996) where authors suggest using questionnaires, that describe questions to which ontology should give answers. That can be very useful in environments where domain experts have very little knowledge of knowledge modelling. **HOLSAPPLE** (Holsapple & Joshi, 2002) methodology uses collaborative approach in building static ontologies. Moreover authors of **HCONE** (Kotis & Vouros, 2003) present decentralized approach to ontology development by introducing regions where ontology is saved during its lifecycle. **OTK Methodology** (Sure, 2003) defines steps in ontology development into detail and introduces two processes – Knowledge Meta Process and Knowledge Process. The steps are also supported by a tool. **UPON** (Nicola, Navigli, & Missikoff, 2005) is an interesting methodology that is based on Unified Software Development Process and is supported by UML language, but it has not been yet fully tested. The latest proposal is **DILIGENT** (Davies et al., 2006) and is focused on different approaches to distributed ontology development.

None of the aforementioned methodologies addresses all aspects of ontology development, therefore unresolved issues still exists (Davies et al., 2006). There is also a lack of Rapid Application Development (RAD) approaches in ontology development, the use of ontologies in business applications and approaches analogous agile methodologies in software engineering. There is also an evident lack of approaches that do not require extensive technical knowledge of formal languages and techniques for capturing knowledge from domain experts. The majority of approaches require an additional role of knowledge engineer that transfers the knowledge into formal syntax within knowledge base.

One of the criticisms of current methodologies is also not supporting the maintenance of developed ontology. Aforementioned methodologies are mainly focused on ontology development for specific application and the development cycle usually ends with the last successful iteration. The domain expert is dependent on knowledge engineer in case of any modifications, due to experts' lack of knowledge of languages for formal representation. Business rules and Model Driven Architecture (MDA) (Frankel, 2003) are interesting approaches to solve this problem.

In the last decade business rules have become very popular and widely used, especially in the field of information systems. The main advantage is definitely flexibility that they introduce and capability of altering business logic. Business rules can be defined as declarations of policy or conditions that must be satisfied (Kardasis & Loucopoulos, 2004) and their role is to define how operational decisions are executed within an organisation. Like ontologies, business rules originate from artificial intelligence where they are used for knowledge representation. While having common roots, business rules approach and ontology construction with Semantic Web technologies address different targets. The main purpose of business rules is to improve communication between people, while the use of ontologies on the Semantic Web tries to facilitate computer-to-computer communication. According to that both approaches are situated at different MDA levels. Ontology fits the execution level – Platform Specific Model (PSM) while data vocabulary and business rules are located on Computation Independent Model (CIM).

Both approaches share the same goal to capture semantics independently of specific application or task. In this way they differ from conceptual modelling approaches (e.g. UML) that describe domain knowledge for the use in specific applications. Especially the idea of application independence leads us to reuse business vocabularies, business rules and ontologies. Both approaches use similar form – they consist of mutually linked concepts and rules (e.g. identity, cardinality, taxonomy etc.) that restrict and define the meaning of those concepts. Business rules approach is focused in natural language and human readable syntax, while in ontologies every element has to be consistent with formal language or it cannot be used in execution environment.

3 RESEARCH METHODOLOGY

The analysis phase of current research includes the review of state of the art in methodologies for information system development, ontology development and business rules approaches for the use in information systems.

Based on the results of current state of the art the next step includes identification of steps and elements that will become candidates for the approach. The goal to improve existing processes for ontology creation is to be followed and integrated with business rules approach. At this stage the results include guidelines and rapid ontology development model respectively. The emphasis is given to intuitive use, suitable for business user that will be able to construct knowledge base and will serve as foundation for the use in business applications. The ontology is persisted in standardised ontology languages (e.g. OWL Lite, OWL DL etc.) on the Semantic Web which will result in achieving as high level of reuse and exchange as possible. Important aspect that is taken under consideration is reuse of existing resource and ontologies while constructing business vocabulary and belonging rules.

To support the defined process a prototype tool will be developed. The tool will aid business user to follow the steps, defined in a model in developing ontology and use it in business application. The main purpose of a tool is a simple and intuitive use, suitable for business user with minimal knowledge of technical knowledge. The process of ontology construction includes reusing existing sources and definition of problem domain at the highest abstract level.

The model with the accompanying tool will be verified on the real case studies from several problem domains (e.g. financial portfolio management, organisation of a rent-a-car etc.).

4 RAPID ONTOLOGY DEVELOPMENT MODEL

The Rapid Ontology Development model for ontology manipulation in general defines 3 simple steps as depicted in Figure 1.

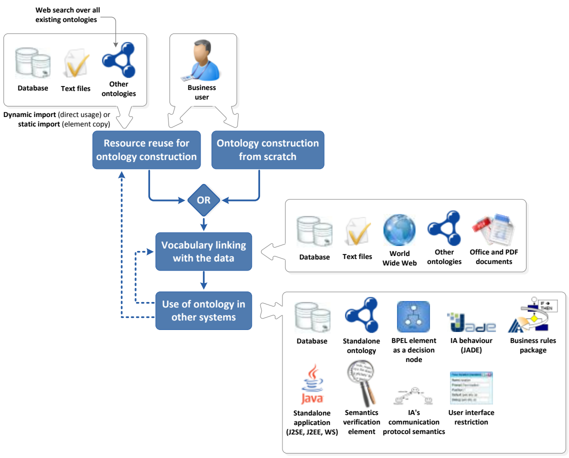

Figure 1: Simplified view of the process of Rapid Ontology Development model

Business user starts with capturing concepts, mutual relations and expressions based on concepts and relations. This task can include reusing elements from various resources or defining them from scratch. When the model is defined, schematic part of ontology has to be binded to existing instances of that vocabulary. This includes data from relational databases, text files, other ontologies etc. The last step in bringing ontology into use is creating functional components for the employment in other systems.

With every step of the process user has the ability to return to previous step and repeat the iteration until satisfactory results are obtained.

4.1 Ontology completeness

To aid users and to simplify progressing through steps in process of Rapid Ontology Development model ontology completeness indicator is introduced:

$$OC = f(C, P, R) \in [0,1].$$

In $OC's$ calculation set of concepts C, set of properties P and set of rules R are considered. Based on that input the output value in an interval [0,1] is calculated. The higher the value, more complete the ontology is and therefore can be used in more complex environment.

If the ontology completeness level is too low the ontology cannot be used as a functional component in other system or only in restricted scenarios (e.g. create database schema). While OC increases towards 1, the developed ontology starts showing as appropriate for more complex tasks (e.g. BPEL element as decision node, intelligent agent's communication protocol etc.).

4.2 Ontology construction

The primary step of ontology construction provides us two options – construction from scratch or reuse of existing resources. While ontology construction from scratch is practically very rare, the emphasis was given to reuse of existing elements for ontology construction.

Some authors (Hefke & Stojanovic, 2004; Mao, Wu, Chen, & Zheng, 2006) have argued that early stages of ontology construction should be treated in unstructured and informal way. Due to that reason the model for rapid ontology development will adopt mind maps (Buzan & Buzan, 1996; Farrand, Hussain, & Hennessy, 2002) approach.

4.2.1 Ontology construction from scratch

Business user in the primary step **defines concepts** which can be found within the problem domain by simply sketching a mere outline (e.g. car movement, sending branch etc.). At this stage concepts are enumerated and optionally described in default language. While links between concepts exist, they are not described or labelled, because this is only a simple mind map.

To further define ontology **additional information about concepts** is added by user again using the mind map view. All attached properties are by default *datatype properties* with *string* type and optional cardinality.

At this stage the ontology is only roughly defined which is denoted by the ontology completeness (OC) level. While the OC level is unsatisfactory, the user has to traverse through all properties p_i of specific concept C_j. With every property a conclusion has to be made if we are dealing with a ***simple*** or ***complex*** property. To classify the property we only have to answer the question whether property contains atomic data (e.g. year, description, number of elements etc.) or it is composed (e.g. rental car) which leads to further decomposition. The cardinality of property is set to optional in both directions.

In case of simple property first the type is defined with the possibility to enter some sample data and trigger automatic identification or select the type on your own. Optionally the domain of values can also be defined.

When dealing with complex property the *Datatype property* is replaced with *Object property* type. Now we face two options – whether linking the property to existing concept or creating a new concept that will define our complex property.

To achieve better domain description restriction rules are introduced. First type of rules that user is able to enter is a **structural rule** that specify what things are (e.g. rental has exactly one renter) and are true by definition. The second type is an **operative rule** that specify what an enterprise must do (e.g. each rental car that is assigned to a rental must be at the pick-up branch of the rental) and can be broken.

The construction of a rule always starts with a verb concept (e.g. rental has driver). After applying an obligation or necessity (e.g. it is obligatory that rental has driver) qualifications, quantifications and conditions are added if necessary (e.g. it is obligatory that each rental has at most 4 drivers).

A typical business person does not talk about quantifications, but expresses quantifications in almost every statement he makes. He does not talk about conjuncts, disjuncts, negands, antecendents and consequents – but these are all part of the formulation of his thinking. Therefore transformation to formal semantics structure of business discourse must be introduced and will be further discussed in section 4.3.

Finally the problem of dividing concepts into subsets is also tackled to achieve inheritance.

4.2.2 Resource reuse for ontology construction

Rare cases can be found where ontology is constructed from scratch because there are always some elements that can be reused. Resources containing ontology elements range from databases, data models, text files, World Wide Web, other ontologies, reports written in Office documents etc. Candidates for reuse include concepts, relations and logical restrictions.

In reusing existing ontologies Rapid Ontology Development model defines a quick overview of existing ontologies and offers two distinct ways of reemploy. First is ***dynamic import***, where ontology is referenced using a selected namespace. In this case all elements of imported ontology are available for usage. The second approach is ***static import*** where user can select which elements are needed and are thereafter copied to ontology.

When importing from **database schema** there is always a possibility to select tables, relations, attributes and enforced referential integrities as depicted in Figure 2.

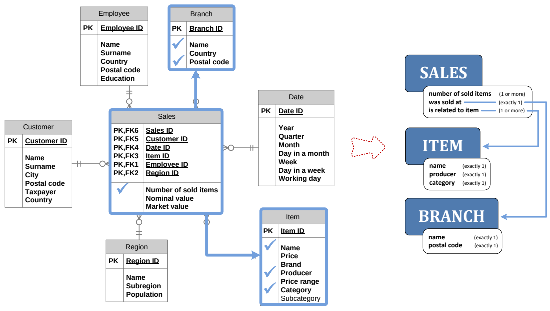

Figure 2: Reuse of relational database scheme for ontology construction

In case of importing comma separated value **(CSV) files** (see Figure 3) a transformation process of normalization from 1[st] normal form to 3[rd] normal form is executed. Based on normalized data, concepts, properties and relations are created. Property types are also automatically extracted from provided data.

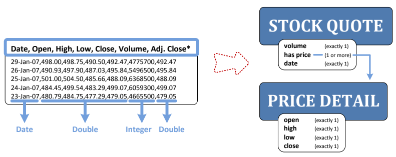

Figure 3: Reuse of CSV file for ontology construction

To enable reuse of existing ontologies integration with Semantic Web search engines and repositories (e.g. Swoogle, Cyc, GALEN, WordNET etc.) is also provided.

4.3 From business statements to formal ontology

As already discussed in section 4.2.1 the mind maps and business rules approach is proposed. The Business Rules Mantra in Semantics of Business Vocabulary and Business Rules (SBVR) defined by OMG (OMG, 2006) states that (see Figure 4) “Rules are built on Facts. Facts are built on Terms”.

Business Rules Mantra offers a great simplification and emphasizes describing businesses, not IT systems that serve them in a language understandable by business people.

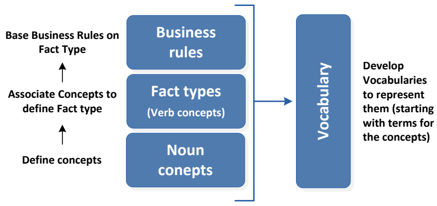

Figure 4: Business Rules Mantra

Based on semi-formal methods, similar to natural language, business users are able to capture the knowledge. With the aid of intelliOnto tool they will be able to codify it in one of the standard Semantic Web Ontology languages (e.g. OWL + SWRL, RDF etc.). As depicted in Figure 5 the perception of the problem domain as business users comprehend it differs from a formal presentation of the knowledge in IT system.

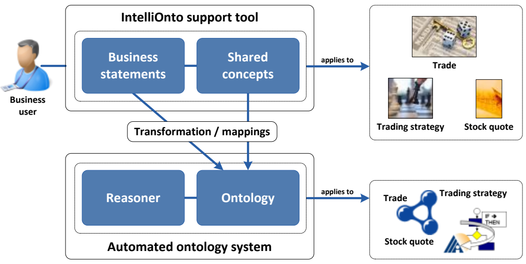

Figure 5: Business view with transformation to information system level

Therefore transformations and mappings with regards to MDA approach will be introduced. This will enable users without extensive technical knowledge to manipulate ontology and even deploy altered versions and test them. Transformation process will therefore follow these simple steps:

1. **Start with a business oriented statement.** (e.g. It is prohibited that a barred driver is a driver of a rental.)
2. **Identify symbols in vocabulary.** (e.g. It is prohibited that a <u>barred driver</u> *is* a <u>driver</u> of a <u>rental</u>.)
3. **Parse according to language rules.** (e.g. It is prohibited that (is (a barred driver, a driver of a rental))
4. **Restate as facts of logical formulation.**
5. **Represent facts of logical formulation as objects.**
6. **Write objects in one of Semantic Web Ontology languages** (e.g. OWL + SWRL, RDF etc.).

To achieve the level of language as similar as possible to natural language several operators (e.g. It is necessary that … and It is possible that …) will be introduced. Predefined templates will be used to enter business statements while mind map approach is used to add information about concepts and relations. There will also be a great simplification in entering rules by introducing decision tables and decision trees (besides if-then statements).

4.4 Linking with the data at instance level

Until this stage our ontology is defined mainly from terminological aspect (T-BOX), while it still lacks assertion aspects (A-BOX). This will be improved by linking the ontology with actual instances of data in several repositories – database, text file, imported ontologies etc.

Primary support that will be offered is persisting ontology in widely used repositories like **Joseki** and **Sesame**. These repositories besides persistence also support querying and are preferred ways of storing ontology instances.

To support linking with instances located in **relational databases** or **files** interfaces are introduced. The purpose of interface is to map schematic part of ontology to actual data located in tables in a database or lines in a file.

4.5 The use of ontology as a functional component in other systems

After completing the terminological and assertion aspect of building ontology our vocabulary consists of enough information that can be efficiently used as a functional component in other systems. Schema and instances can be used for various purposes:

- BPEL element with a role of decision node,
- intelligent agent’s (IA’s) behaviour (JADE),
- business rule package,

- standalone application (J2SE, J2EE, WS),
- standalone ontology and
- database,
- semantics verification,
- intelligent agent's (IA's) communication protocol semantics,
- user interface restriction.

In the latter use cases (database, semantics verification, IA's communication protocol semantics and user interface restriction) only schematic part of ontology is to be used. The purpose is to employ semantics in different scenarios.

For database creation only a portion of schematic part of ontology is feasible for usage – concepts (to tables), properties (to relations) and some rules (to reference integrity restrictions). It is similar with other scenarios such as semantics verification, where ontology can for example be used for semantic confirmation of external resources (e.g. CSV files, databases etc.) if they correspond to restrictions. With IA's communication protocol semantics also the vocabulary elements can be used, while scenario of user interface restriction is mainly focused on enforcing data types and relations between input data.

More advanced use of ontology besides schematic part includes also the individuals and external resources as depicted in Figure 6, where portfolio management ontology is presented.

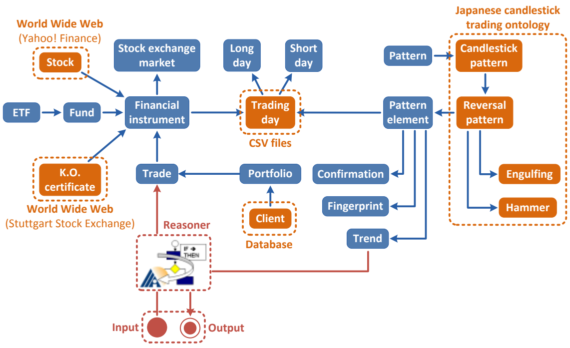

Figure 6: Sample ontology from portfolio management domain, linked to several external resources

This example of a simple broker has several financial instruments (fund, stock, certificate etc.) defined that can be traded on a stock market. A client can execute multiple trades within his portfolio on a trading day. While there are numerous users they all use some trading strategy that is defined in its own ontology and can be derived from well known strategies. The schematic part of ontology is centrally defined but the data in a form of instances is located in distributed sources. The financial instrument information can be found online at various web pages (e.g. stock at Yahoo! Finance[1], K.O. certificates at Stuttgart Stock Exchange[2] etc.). Client information is also available in the database that CRM application uses, so ontology employs that data directly. The information about current stock quotes is found in CSV files that can also be used in ontology under trading day concept. The current user employs Japanese candlestick trading ontology; therefore an external ontology is used, where all the trading rules are already defined.

The following example can be packed as a component with parts of ontology coupled with reasoner and selected input and output parameters (see Figure 6). Target components range from business rule package, standalone Java application, BPEL element (see Figure 7), IA's behaviour etc.

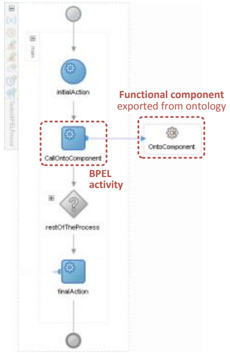

Figure 7: Exporting functional component as BPEL activity

5 INTELLIONTO SUPPORT TOOL

To facilitate the use of Rapid Ontology Development model, IntelliOnto tool is being developed. The tool is a web based application

[1] http://finance.yahoo.com
[2] http://www.boerse-stuttgart.de

employing several open source technologies for ontology manipulation (Protégé[3], JENA[4], KAON2[5] etc.) and Web 2.0 technologies for user interface (e.g. AJAX). The 3-tier architecture has been chosen due to easier integration into existing development environments and to include it into existing social network applications (e.g. Facebook).

The user interface is divided into several parts as depicted in Figure 8. There is concept neighbourhood view, list of concepts, ontology completeness, and steps of ontology construction process, status and drawing board for concepts and rules manipulation.

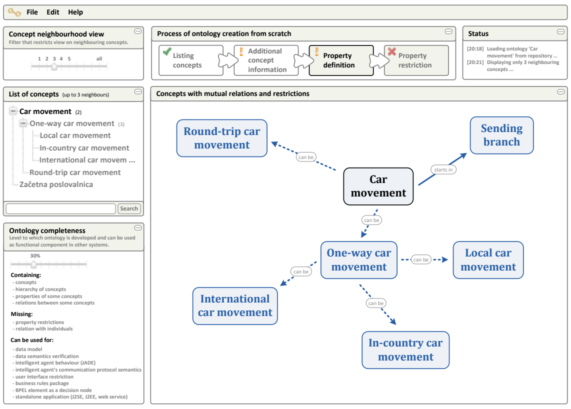

Figure 8: IntelliOnto support tool

The concepts neighbourhood view enables as to cope with the complexity of ontology by using filter to restrict view on neighbouring concepts. With that approach the locality is introduced so the user does not have to deal with all other concepts that are not related to the concept of observation. Only concepts that are semantically related are shown.

List of concepts simply display all elements that are available for inclusion in problem domain modelling. The list is also sorted according to subconcept hierarchy by introducing parent child relationship.

Ontology completeness defines the level of ontology which tells the business user to what extent ontology is completed and can be used as a functional component in other systems. Very simple lists of what ontology includes and what it lacks are displayed. Besides those facts recommendations are also presented, e.g. the developed ontology could be used for data model creation, data semantics verification or with linking to instances business rules package can be created.

Steps of ontology construction process provide users to follow the Rapid Ontology Development model step-by-step. That is achieved by displaying the current step and what is needed to progress to the next one. There is always a possibility to return to previous steps or to skip a step. With every step in ontology construction process the user interface is adapted, meaning that in early stages of the process only simple editing of concepts is allowed while getting along you are given more options to manipulate the ontology.

Status part of intelliOnto tool gives you some details about executed actions and displays some feedback system information.

Drawing board for concepts and rules manipulation is central part of user interface and it is graphically oriented. It enables users to simply create concepts, create relationships, dependencies and restrictions in a way similar to natural language. Business users are able to enter their statements in several ways. Simple business statement in a semi structured language, restrictions in a form of decision tables and decision trees. The support for synonyms and homonyms is also essential. Synonyms reference preferred terms, whereas business can require that official communications use preferred terms.

When creating a new ontology business user can select one of the commonly used languages or use a custom one and define the default language for the project. All important facts from the business vocabulary and accompanying business rules with restrictions are automatically translated into commonly used languages. The integration with existing ontologies on the Internet is provided at the beginning of a construction process. At project creation a search over existing ontologies on the Internet (e.g. Swoogle) and in the local repository is executed to facilitate reuse of ontological elements.

After the process of ontology construction is concluded the final step is to export ontology in a form of a functional component as discussed in

[3] http://protege.stanford.edu
[4] http://jena.sourceforge.net
[5] http://kaon2.semanticweb.org

section 4.5. Very important aspects which will also be supported by the tool are versioning, altering the ontology and redeployment. When some business logic (e.g. pricelist in rent-a-car organization) changes, the business user has the ability to alter the ontology (e.g. add some new rules) and redeploy a new version to production environment.

6 CONCLUSIONS

The definition of Rapid Ontology Development (ROD) model and development of intelliOnto tool is fully in progress. After completing the first phase of defining ROD the development of intelliOnto has begun in parallel. The ROD definition has past the early stage into the mature stage, while the intelliOnto tool is still in its early development stage.

At this stage rough outline of ROD is completed with some general steps as depicted in section 4. Further work will include detail decompositions of steps already defined. The work on the ontology completeness indicator is also in progress and has yet to be yet fully defined.

The current functionalities of intelliOnto tool include primary steps of ontology construction with adding concepts, relations and some basic rule parsing in semi natural language. The work on vocabulary linking with the data and the use of ontology in other systems is in progress and has high priority in future work plans.

Working examples were already partly presented and are from the domain of financial portfolio management and organisation of a rent-a-car. Some additional effort is needed to fully introduce all functionalities of intelliOnto tool and accompanying Rapid Ontology Development model. The aforementioned examples will be used for verification of the approach and a comparison to other approaches of ontology development and information system development.

REFERENCES

Abou-Zeid, E. S. (2003). What can ontologists learn from knowledge management? *Journal of Computer Information Systems, 43*(3), 109-117.

Bajec, M., & Krisper, M. (2005). A methodology and tool support for managing business rules in organisations. *Information Systems, 30*(6), 423-443.

Bechhofer, S., & Goble, C. (2001). Thesaurus construction through knowledge representation. *Data & Knowledge Engineering, 37*(1), 25-45.

Bernaras, A., Laresgoiti, I., & Corera, J. (1996). *Building and resusing ontologies for electrical network applications.* Paper presented at the European Conference on Artificial Intelligence (ECAI '96).

Brambilla, M., Celino, I., Ceri, S., & Cerizza, D. (2006). *A Software Engineering Approach to Design and Development of Semantic Web Service Applications.* Paper presented at the 5th International Semantic Web Conference, Athens, USA.

Buzan, T., & Buzan, B. (1996). *The Mind Map Book: How to Use Radiant Thinking to Maximize Your Brain's Untapped Potential, 3rd edition*. New York, USA: Plume.

Corcho, O., Fernandez-Lopez, M., & Gomez-Perez, A. (2003). Methodologies, tools and languages for building ontologies. Where is their meeting point? *Data & Knowledge Engineering, 46*(1), 41-64.

Dahlgen, K. (1995). A linguistic ontology. *International Journal of Human-Computer Studies, 43*(5/6), 809-818.

Davies, J., Studer, R., & Warren, P. (2006). *Semantic Web Technologies - trends and research in ontology-based systems*. Chichester, England: John Wiley & Sons.

Dong, Y., & Li, M. S. (2004). HyO-XTM: a set of hyper-graph operations on XML Topic Map toward knowledge management. *Future Generation Computer Systems, 20*(1), 81-100.

Dorsey, P. (2002). *The Business Rules Approach to Systems Development*. Retrieved 30.4.2007, from http://www.dulcian.com/BRIM%20Documents/The%20Business%20Rules%20Approach%20to%20Systems%20Development.htm

du Plessis, M. (2005). Drivers of knowledge management in the corporate environment. *International Journal of Information Management, 25*(3), 193-202.

Farrand, P., Hussain, F., & Hennessy, E. (2002). The efficacy of the 'mind map' study technique. *Medical Education, 36*(5), 426-431.

Fernandez-Lopez, M., Gomez-Perez, A., Sierra, J. P., & Sierra, A. P. (1999). Building a chemical ontology using Methontology and the Ontology Design Environment. *Intelligent Systems, 14*(1).

Frankel, D. S. (2003). *Model Driven Architecture: Applying MDA to Enterprise Computing*: Wiley.

Gomez-Perez, A., Fernandez-Lopez, M., & Corcho, O. (2003). Ontological Engineering. *Advanced Information and Knowledge Processing*.

Hefke, M., & Stojanovic, L. (2004). *An Ontology-based approach for competence bundling and composition of ad-hoc teams in an organisation.* Paper presented at the I-KNOW, Graz, Austria.

Heflin, J., & Hendler, J. (2000). Searching the web with SHOE. In *Artificial Intelligence for Web Search* (pp. 36-40). Menlo Park, USA: AAAI Press.

Holsapple, C. W., & Joshi, K. D. (2002). A collaborative approach to ontology design. *Communications of the ACM, 45*(2), 42-47.

Jarrar, M., & Meersman, R. (2002). *Formal ontology engineering in DOGMA approach.* Paper presented at the International Conference on Ontologies, Databases and Applications of Semantics (ODBase 02).

Jovanović, J., & Gašević, D. (2005). Achieving knowledge interoperability: An XML/XSLT approach. *Expert Systems with Applications, 29*(3), 535-553.

Kardasis, P., & Loucopoulos, P. (2004). Expressing and organising business rules. *Information and Software Technology, 46*(11), 701-718.

Kotis, K., & Vouros, G. (2003). *Human centered ontology management with HCONE.* Paper presented at the IJCAI '03 Workshop on Ontologies and Distributed Systems.

Mao, Y. X., Wu, Z. H., Chen, H. J., & Zheng, X. Q. (2006). Concept map model for web ontology exploration. In *Frontiers of Www Research and Development - Apweb 2006, Proceedings* (Vol. 3841, pp. 544-555).

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM, 38*(11), 39-41.

Mylopoulos, J. (1998). Information modeling in the time of the revolution. *Information Systems, 23*(3-4), 127-155.

Nicola, A. D., Navigli, R., & Missikoff, M. (2005). *Building an eProcurement ontology with UPON methodology.* Paper presented at the 15th e-Challenges Conference, Ljubljana, Slovenia.

OMG. (2006). *Semantics of Business Vocabulary and business Rules (SBVR)*

Park, J., & Hunting, S. (2002). *XML Topic Maps: Creating and Using Topic Maps for the Web*. Boston, USA: Addison-Wesley.

Rao, J., Dimitrov, D., Hofmann, P., & Sadeh, N. (2006). *A Mixed Initiative Semantic Web Framework for Process Composition.* Paper presented at the 5th International Semantic Web Conference, Athens, USA.

SanJuan, E., & Ibekwe-SanJuan, F. (2006). Text mining without document context. *Information Processing & Management, 42*(6), 1532-1552.

Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., van de Velde, W., et al. (1999). *Knowledge Engineering and Management - The CommonKADS Methodology*. London, England: The MIT Press: Cambridge, Massachusetts.

Sure, Y. (2003). *Methodology, Tools & Case Studies for Ontology based Knowledge Management.* University of Karlsruhe.

Swartout, B., Ramesh, P., Knight, K., & Russ, T. (1997). *Towards distributed use of largescale ontologies.* Paper presented at the Symposium on Ontological Engineering of AAAI, Stanford, USA.

Uschold, M., & Grueninger, M. (1996). Ontologies: principles, methods and applications. *Knowledge Sharing and Review, 11*(2).

Uschold, M., & King, M. (1995). *Towards a methodology for building ontologies.* Paper presented at the Workshop on Basic Ontological Issues in Knowledge Sharing (IJCAI '95), Montreal, Canada.

Veale, T. (2006). An analogy-oriented type hierarchy for linguistic creativity. *Knowledge-Based Systems, 19*(7), 471-479.

Waterson, A., & Preece, A. (1999). Verifying ontological commitment in knowledge-based systems. *Knowledge-Based Systems, 12*(1-2), 45-54.