OntoGame: Towards Overcoming the Incentive Bottleneck in Ontology Building

Katharina Siorpaes and Martin Hepp

Digital Enterprise Research Institute (DERI), University of Innsbruck, Innsbruck, Austria
katharina.siorpaes@deri.at, mhepp@computer.org

Abstract. Despite significant advancement in ontology learning, building ontologies remains a task that highly depends on human intelligence, both as a source of domain expertise and for producing a consensual conceptualization. This means that individuals need to contribute time, and sometimes other resources, to an ontology project. Now, we can observe a sharp contrast in user interest in two branches of Web activity: While the "Web 2.0" movement lives from an unprecedented amount of contributions from Web users, we witness a substantial lack of user involvement in ontology projects for the Semantic Web. We assume that one cause of the latter is a lack of proper incentive structures of ontology projects, i.e., settings in which the perceived benefits outweigh the efforts for people to contribute.

As a novel solution, we (1) propose to masquerade collaborative ontology engineering behind on-line, multi-player game scenarios, in order to create proper incentives for humans to help building ontologies for the Semantic Web. Doing so, we adopt the findings from the already famous "games with a purpose" by von Ahn, who has shown that presenting a useful task, which requires human intelligence, in the form of an on-line game can motivate a large amount of people to work heavily on this task, and this for free. Then, we (2) describe our OntoGame prototype, and (3) provide preliminary evidence that users are willing to invest a lot of time into those games, and, by doing so, unknowingly weave ontologies for the Semantic Web.

1 Introduction

One can observe only limited involvement of users in building ontologies, which is one of the reasons that may explain the shortage of current, high-quality domain ontologies. While the technical aspects of collaborative ontology engineering are already an established research topic, little attention has so far been dedicated to the incentive structures of ontology construction and usage, i.e. research on the motivations for people to contribute to an ontology or to adopt it. However, since building ontologies is a task that depends on human intelligence, both as a source of domain expertise and for producing a consensual conceptualization, it cannot be taken for granted that a sufficient amount of individuals engages in ontology projects just on the basis of altruism. Also, it is important to stress that ontology building is inherently a collaborative task, for two reasons: first, ontologies are supposed to be community contracts [1, 2]; second, the combination of required domain expertise and modeling skills is more likely to be found in a group than in a single individual.

In short, producing ontologies consumes resources; thus people with respective expertise must have a sufficient motivation to contribute. Unfortunately, the incentives for ontology building have not been a popular research topic so far. In this context, it can be observed that there is a significant difference in the incentive structures of formal ontologies vs. e.g. collaborative tagging. When tagging a resource, someone adding tags to data achieves two things at the same time: (1) improving the public vocabulary (i.e. the set of tags) and (2) improving his own access to the knowledge assets. Traditional ontology engineering, in contrast, detaches the effort from the benefits: by building an ontology alone, one does not improve one's own access to existing knowledge, while others may enjoy the added value of an ontology without having invested into its construction. Thus, it is not per se granted that those investing resources in the creation or improvement of an ontology will materialize sufficient benefits out of the usage of this ontology.

As a novel solution, we propose to masquerade user contributions to collaborative ontology engineering behind on-line, multi-player game scenarios in order to establish proper incentives for humans to help building ontologies for the Semantic Web. Doing so, we adopt the findings from the already famous "games with a purpose" by von Ahn, who has shown that presenting a useful task, which requires human intelligence, in the form of an on-line game can motivate a large amount of people to work heavily on this task, and this for free. Our goal is that users mastering the intellectual challenges of our games unknowingly weave ontologies for the Semantic Web.

1.1 The Motivational Divide: Web 2.0 is Fun, Ontology Engineering is Not

We can observe that "Web 2.0" applications enjoy great popularity and comprise strong user incentives [3]: Tagging, i.e., users describing objects with freely chosen keywords (tags) in order to retrieve content more easily, is immediately rewarding and provides some sort of community spirit. It is immediately rewarding, because each tagging action improves my personal access to the knowledge assets, and it provides community spirit, since seeing others agree with my own favorite tags gives some positive feeling of being in alignment. In the case of Wikipedia, even though arguably not a core "Web 2.0" application, we can see similar patterns. Wikipedia currently contains more than 1.7 million articles[1] with very high quality [4]. Kuznetsov [5] has traced this back to a multiplicity of social motivations for people to contribute to Wikipedia. In parallel, von Ahn's ESP game [6] has demonstrated how the high amount of hours invested in playing games on the Internet everyday can be exploited for useful purposes. The ESP game, which is about finding consensual tags for images, is now extremely popular: Von Ahn observed that some people are playing the game more than 40 hours per week. Within a few months after the initial deployment on October 25, 2003, the game collected more than 10 million consensual image labels, and this without paying a cent to the contributors.

[1] http://wikipedia.org/, retrieved on May 11, 2007

Our idea is to present various tasks of ontology engineering, that require human intelligence, in the form of multi-player game scenarios. Eventually, we have preliminary evidence that the proposed games are (1) sufficiently interesting for users and (2) that the contributions are a valuable input for building ontologies.

1.2 Related Work

Von Ahn and colleagues have already described a series of games with different purposes: The ESP game [7] aims at labeling images on the Web. The idea is that two players, who do not know each other, have to come up with identical tags describing the image. Peekaboom [8] is a related game for locating objects within images. Verbosity [9] is a game for collecting common sense facts. Finally, Phetch [10] is a computer game that collects explanatory descriptions of images in order to improve accessibility of the Web for visually impaired. Apart from Verbosity, we do not know of any other work on exploiting computer game scenarios for the conceptualization of domain knowledge, in particular none that is using a game for soliciting human intelligence in order to create and maintain ontologies.

1.3 Contribution and Overview

In this paper, we (1) derive from popular ontology engineering methodologies a set of tasks that require a substantial amount of human intelligence and thus user contributions, (2) suggest multiple game scenarios that represent these tasks, (3) describe our OntoGame prototype, and (4) give preliminary evidence that OntoGame is not only fun to play but that players are also able to produce useful and ontologically correct results. Finally, we summarize our work and describe pending and future extensions.

2 Multi-Player Games for Ontology Construction

The tasks in ontology construction have been analyzed in depth by work on ontology engineering methodologies, see e.g. [2]. For the purpose of this initial paper, we focus on a subset of tasks, which are classified according to the lightweight Uschold and King’s methodology [11]. In the following, we first outline the most important tasks. Then, we suggest various multi-player game scenarios that represent these tasks.

2.1 Tasks in Ontology Engineering and Ontology Usage

Uschold and King’s methodology describes four main activities: (1) *identification* of the purpose, (2) *building* the ontology, (3) *evaluation*, and (4) *documentation*. The second activity is divided into three sub-processes: (a) *ontology capture,* which comprises the identification of key concepts and relationships; (b) *ontology coding* as “committing to basic terms […] and writing the code”; (c) *integrating existing ontologies,* i.e. the reuse of or alignment with existing ontologies.

Within those four main activities, the following are steps that we think are particularly suited for representation as game scenarios.

Collecting named entities: Relevant conceptual elements of the domain of discourse must be identified and a unique key assigned.

Typing named entities according to the ontology meta-model: With typing named entities we mean the process of defining the type of conceptual element for each named entity, based on the distinctions from the applicable ontology meta-model. Many popular ontology meta-models support *classes*, *properties*, and *individuals* as core types. Classes "provide an abstraction mechanism for grouping resources with similar characteristics" [12]. Properties are a means for describing individuals of such classes in more detail; a popular distinction is between *object properties* relating an individual to another individual and *data type properties* relating an individual to a data value. Individuals can be viewed as instances of classes, i.e. entities with similar characteristics.

Adding taxonomic and non-taxonomic relations: A flat collection of ontological elements can be enriched by adding taxonomic and non-taxonomic relations. The most prominent form of this task is arranging the concepts into a subsumption hierarchy by introducing subClassOf relations.

Alignment between multiple ontologies: There may be multiple ontologies with a partial overlap in scope or content. In order to increase interoperability between data, such ontologies should be semantically related to each other at a conceptual level, which is known as ontology alignment.

Modularization: Depending on the domain of discourse, it is often useful to define subsets of concepts based on their ontological nature or target applications, since they may be more manageable.

Lexical enrichment: Ontology engineering methodologies tend to focus on formal means for specifying ontologies. However, in order to describe the *intended* semantics of ontology elements, informal means, like natural language labels or synonyms are also needed. However, relating a conceptual element to terms or synonym sets requires careful human judgement, since otherwise, inconsistencies between the informal part and the formal part of the ontology may result.

While not part of the actual ontology development, the **population of an ontology with instance data** (also known as annotation) is also included in here, because it is another task that can be addressed in a gaming scenario. In fact, one may assume that the absolute amount of human intelligence needed for annotating data will be much higher than that for building respective ontologies, and motivating many Web users to contribute to that is thus very promising.

2.2 Suggested Scenarios

As said, our core idea is to develop games that are fun to play and at the same time acquire human judgements relevant for specifying a conceptualization of a domain, so that there is a proper incentive for users to build and maintain ontologies. In Table 1, we present different gaming scenarios which address the tasks from the previous section.

Table 1. Suggested Scenarios

Scenario	**Task**	**Intellectual Challenge**
Classifying conceptual enitities according to types of an ontology meta-model: Two players are faced with the same Wikipedia article and need to agree on whether what the article describes is a class, an individual, or a relation.	Typing named entities	Assessing whether something is a significant (abstract or tangible) individual, an abstraction over multiple individuals (i.e. a class), or a type of relationship between two individuals.
Finding attributes for a class: Two players are faced with the description of a class and need to agree on an attribute for instances of this class.	Collecting and typing named entities (attributes)	Spotting and naming typical characteristics of instances of a given class.
Determining the range of an attribute: Two players are faced with an attribute and need to agree on the proper range for values, i.e. a class or datatype.	Typing named entities	Spotting the proper range of values, either as a class or a datatype.
Finding a super-class: Two players are faced with a class and need to come up with (and agree upon) a super-class of this class.	Adding taxonomic relations	Finding a proper, consensual abstraction.
Identifying typed relations between two classes: Two players are faced with two different ontology classes and need to enter matching names for typical non-taxonomic relations between those two.	Collecting and typing named entities (relations), adding non-taxonomic relations	Spotting and naming types of relations between instances of two classes.
Identifying the class for an instance: Two players are faced with an instance and need to come up with (and agree upon) the label for a class suitable for that instance.	Ontology population	Spotting a proper class for a given entity.
Annotation of a resource: Two players are faced with a Web resource and have to agree on a proper annotation using a given ontology as quickly as possible.	Ontology population	Grasping the semantic essence of a resource and expressing it using a given vocabulary.
Mapping two ontology elements: Two players are faced with two conceptual elements of the same type from two different ontologies and need to agree on the existence and type of semantic correspondence between the two.	Alignment	Judging upon the existence of a semantic correspondence between two conceptual elements and selecting the type of correspondence (e.g. sameClass, subClassOf,…)
Ontology modularization: Two players are presented with the name and description of a domain ontology and need to agree on which elementes from a given set of conceptual elements belong into this module.	Modularization	Assessing the domain relevance of a given conceptual element.

Most scenarios can be made more challenging and at the same time directed towards less obvious solutions by a “taboo word” list, representing solutions that are already known, which ensures that new valuable information is gained. Also, for free text entries and for choosing elements from an existing ontology, users should be supported by auto-complete and suggest functions in order to avoid friction by lexical variants or spelling mistakes.

2.3 Input Data for the Games

There are two different types of input for the game scenarios described in the previous section. We distinguish the following two types of data sources:

Core Data: With that we mean input data that is presented to the players. In most cases it is necessary to have content to start from; otherwise, the gaming fun is limited. We identify several external resources that can be used as such input data: First, Wikipedia, which can be regarded as a huge collection of conceptual entities identified by a URI, for which the conceptual stability has been recently demonstrated [13]. Second, lightweight ontologies extracted from folksonomies [14], and third ontologies and instance data from Semantic Web search engines like Swoogle [15, 16] and Watson[2]. Fourth, we can used popular upper-level ontologies like Proton [17].

Complementing Data: With that we mean additional data that can be helpful for improving the gaming fun or the usefulness of the results. For example, we can use various lexical resources for tolerating lexical variants, foreign language entries, or synonyms as correct answers. This simplifies the process of consensus finding between two users at a conceptual level and likely increases the gaming fun (e.g. that you do not fail a task just because of one user using British English and the other American English). To mitigate such problems, we propose to make use of the several dictionaries that are online accessible such as Leo Dictionary[3] and the Wordnet synsets [18].

3 OntoGame

In this section, we provide a description of OntoGame, outline the architecture of the implementation, and discuss the handling of cheating and mischievous users.

3.1 Overview

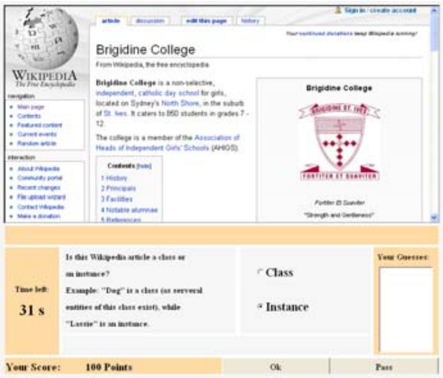
For the first prototype of our game approach, we use Wikipedia articles as conceptual entities, present them to the players, and have the users (1) judge the ontological nature and (2) find common abstractions for a given entry. The game is played by multiple players in parallel in teams of two players. The pairs are defined by random and are anonymous, i.e. players do not know each other and have no way of communicating with each other. Using the Wikipedia "random article" functionality[4], both players will see the initial part of a Wikipedia article (**Fig. 1**), which should be enough to grasp the intention of the article. The goal of a user playing our first OntoGame prototype is then two-fold: The first challenge is to guess whether the co-player judges the Wikipedia article as an **instance** or as a **class**, a distinction which is common in many popular ontology meta-models. If their answers do not match, both players are taken to the next article and no points are awarded. If they agree it to be an

[2] http://kmi-web05.open.ac.uk/WatsonWUI/

[3] http://dict.leo.org

[4] http://en.wikipedia.org/wiki/Special:Random

instance, the second challenge is to propose the label for a suitable **class** to which this instance belongs. If both players agree upon it being a class, the second challenge is to propose the label for a **super-class** (Fig. 2). In case they manage to reach consensus, they get points awarded and are taken to another article. The users are given 2 minutes time to agree on as many Wikipedia articles as possible.



Brigidine College

From Wikipedia, the free encyclopedia

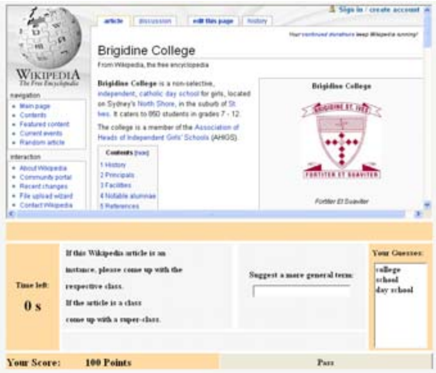
Brigidine College is a non-selective, independent, catholic day school for girls, located on Sydney's North Shore, in the suburb of St. Ives. It caters to 850 students in grades 7 - 12.

The college is a member of the Association of Heads of Independent Girls' Schools (AHIGS).

Is this Wikipedia article a class or an instance?

Example: "Dog" is a class (as serveral entities of this class exist), while "Lassie" is an instance.

Fig. 1. OntoGame: Phase I



Brigidine College

From Wikipedia, the free encyclopedia

Brigidine College is a non-selective, independent, catholic day school for girls, located on Sydney's North Shore, in the suburb of St. Ives. It caters to 850 students in grades 7 - 12.

The college is a member of the Association of Heads of Independent Girls' Schools (AHIGS).

If this Wikipedia article is an instance, please come up with the respective class.

If the article is a class come up with a super-class.

Suggest a more general term:

college
school
day school

Fig. 2. OntoGame: Phase II

As said, users can always decide to skip one article and proceed to the next one. If they choose to do so in phase II, points earned for mastering the first phase will remain. In other words, players can always choose to pass without losing the points they earned in phase I. In our opinion, this is an important feature, because (1) it is possible that poor or unsuitable articles are presented, (2) some articles might be too specialized to be understood by the current user, and (3) players may simply be unable to reach consensus. Instead of encouraging random guesses, we rather motivate users to proceed to the next random article.

3.2 Implementation

The architecture of OntoGame is based on Java and includes a game server, a Java applet serving as a client, and several Java servlets enabling communication between the game server and the applet using an object stream over an http tunnel. The game server implements the singleton pattern, which is used to restrict instantiation of a class to one object because in OntoGame exactly one object is needed to coordinate actions across the system including the games, discovering matches, etc. Four different servlets perform the following tasks: login, communication flows for phases I and II of the game, handling user input, matching and passing.

3.3 Cheating and Bad Input

Theoretically it is possible that users try to cheat or undermine the system intentionally. Von Ahn has already described some approaches that aim at minimizing the impact of such behavior. They have shown to be sufficient to make cheating unattractive, and to effectively minimize the impact on the results of the overall task. First, the players are paired anonymously and have no way to communicate directly with each other. Answers entered are visible for the other player only when agreement is reached. Thus, players cannot simply exchange instant messaging contacts or similar as guesses in order to prepare for later cheating. Second, IP address of partners must be different. Third, a massive global agreement (e.g. bots naming all concepts “XY”) can be detected in two ways: (1) if the answer time increases significantly compared to the average time and (2) letting players play several games with known solutions and compare whether the successful matches are a subset of the known solutions. If not, one can either just ignore the results from the game as not trustworthy, or ban the users temporarily.

4 Evaluation

In this section, we describe our evaluation methodology, the participants and test data, and present preliminary evidence on the contribution of our approach.

4.1 Methodology and Participants

We recruited nine individuals with different backgrounds and asked them to play several rounds of OntoGame. We requested them to do so at a pre-defined time in order to ensure that there are enough players to play OntoGame, as there is no single player mode implemented yet. The subjects were instructed to use English words in lower case only. All details of all the games played were recorded. After the experiment, we interviewed the participants about their experiences with the game and analyzed the recorded games. **Participants:** All individuals have experience using the Web, 5 out of 9 hold a degree (bachelor or master) in computer science. Two of them are working in research in the areas of ontologies and the Semantic Web. None had special training in building ontologies. The participants were in different rooms and did not communicate with each other during playing. The game was explained to the participants orally before playing it online.

Table 2. Participants

ID	Age and Gender	Background and Education
1	30, Male	Bachelor in computer science
2	26, Male	Master in computer science
3	24, Female	PhD student in computer science, ontologist
4	26, Male	Bachelor in computer science, ontologist
5	26, Female	Master in computer science
6	20, Female	Student of medicine
7	46, Female	PhD in history
8	17, Male	Student, regular Web user
9	19, Female	Student of economics, regular Web user

4.2 Results

In this section we present the results of our preliminary evaluation regarding the users’ experiences as well as the soundness of the output of the games.

Individuals and their experiences: The most important observation is that all of the players liked the game after playing it a couple of times and after understanding how it worked (Table 3). People that do not work in the area of ontology building, initially had difficulties to understand the meaning of “adding a more general term”. Most wanted to *tag* Wikipedia pages at first try, instead of adding super-class relationships. This shows that we need to improve the help section.

Table 3. Gaming fun and experiences reported by the participants

Player	Summary of the experiences (quotes)
1	Fun but demanding
2	Super, nice game
3	Fun to play, not boring because demanding
4	Fun to play
5	Fun to play
6	Hard to understand at first, had difficulties to grasp the meaning of super-class
7	Once I understood the game, it was really fun to play, addictive
8	Fun to play
9	Hard to grasp the meaning of “a more general term”, fun to play

Pointing out to the players that they have to be able to apply a natural language phrase like *“A is an example of the type of things B”* for instanceOf relations and *“Each example of type A is always also an example of type B”* for subClassOf relations helped a lot. We could also observe that with increasing experience, the score users achieved increased. **Output of the Games:** Even though this is early work and the scope of the evaluation was rather small, the preliminary results (Table 4) of the evaluation of OntoGame are promising. The nine individuals paired up in 10 different teams in order to play 26 rounds. By doing so, they processed 116 Wikipedia pages. The pairs were able to reach agreement on the type of conceptual element (class or instance) of 102 articles (88%). According to user comments, they found the distinction between class and instance straightforward. Additionally, many pages in this experiment that were chosen by the Wikipedia random page functionality described a person, which makes the distinction easy. Of the remaining 102 articles, in phase II more than 65% of the pages could be consensually assigned to a super-class (identified by a label at this stage only). The longer the individuals played (2-3 games), the more agreement they could reach with their partners. We have manually validated every consensual (1) class/instance choice and (2) super-class/instance-of relationship which was the output of OntoGame. We found that the class/instance choice of articles was in 99% of the cases correct. Furthermore, the super-class or instance-of relationships that players agreed on during playing, was appropriate in more than 92% of the cases. Table 4 summarizes the results.

Table 4. Results of the 116 rounds

Number of consensual class/instance choices	103 of 116	88.79%
Number of consensual super-class/instance-of relations	67 of 103	65.05%
Number of correct class/instance choices	102 of 103	99.03%
Number of correct super-class/instance-of relations	62 of 67	92.54%

5 Discussion and Outlook

We have suggested to apply the idea of “games with a purpose” by von Ahn to the tasks typically found in the construction of ontologies and in the annotation of resources. In particular, we defined suitable gaming scenarios and deployed a first prototype. From preliminary experiments with a first set of users, we can see that the resulting games are both fun to play and produce reasonable and useful ontological data. In particular, the high quality of the consensual conceptual choices is surprising. We are currently in the process of extending OntoGame in the following directions: (1) integration of lexical resources, which increases the gaming fun and simplifies consensus reaching, (2) adding new scenarios from Table 1, namely such for proposing attributes and for mapping between different ontologies retrieved from Swoogle or Watson, (3) integrating the PROTON ontology into the challenge, and (4) general improvements of the usability and user interface. In parallel, we are preparing an extended evaluation of our approach, both in a controlled environment and on a Web scale.

Acknowledgments. The authors would like to thank Michael Waltl and Andreas Klotz. This work has been funded by the Austrian BMVIT/FFG under the FIT-IT project myOntology (grant no. 812515/9284).

References

1. Hepp, M., D. Bachlechner, and K. Siorpaes. OntoWiki: Community-driven Ontology Engineering and Ontology Usage based on Wikis, International Symposium on Wikis (WikiSym 2005). 2005. San Diego, California, USA.
2. Gómez-Pérez, A., M. Fernández-López, and O. Corcho, Ontological Engineering. 2004: Springer.
3. Marlow, C., et al., Position Paper, Tagging, Taxonomy, Flickr, Article, ToRead, World Wide Web Conference (WWW2006). 2006, Edinburgh, Scotland.
4. Emigh, W. and S. Herring, Collaborative authoring on the web: A genre analysis of online encyclopedias, Hawai International Conference on System Sciences 2005: Hawai.
5. Kuznetsov, S., Motivations of Contributors to Wikipedia. ACM SIGCAS Computers and Society, 2006. **36**(2).
6. Von Ahn, L., Games with a Purpose. IEEE Computer, 2006. **29**(6): p. 92-94.
7. Von Ahn, L. and L. Dabbish, Labeling Images with a Computer Game, Conference on Computer/Human Interaction (CHI 2004). 2004, ACM.
8. Von Ahn, L., Peekaboom: A Game for Locating Objects in Images, SIGCHI Conference on Human Factors in Computing Systems. 2006, ACM: Montréal, Québec, Canada.
9. Von Ahn, L., M. Kedia, and M. Blum, Verbosity: a game for collecting common-sense facts, Conference on Human Factors in Computing Systems. 2006, Montréal, Canada.
10. Von Ahn, L., et al., Improving Accessibility of the Web with a Computer Game, SIGCHI Conference on Human Factors in Computing Systems. 2006, ACM.
11. Uschold, M. and M. King, Towards a Methodology for Building Ontologies, Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing. 1995: Montreal, Canada.
12. Dean, M. and G. Schreiber, OWL Reference, W3C Working draft. 2003.
13. Hepp, M., K. Siorpaes, and D. Bachlechner, Harvesting Wiki Consensus: Using Wikipedia Entries as Vocabulary for Knowledge Management. IEEE Internet Computing, Vol. 11, No. 5, pp. 54-65, 2007.
14. Van Damme, C., M. Hepp, and K. Siorpaes, FolksOntology: An Integrated Approach for Turning Folksonomies into Ontologies, Bridging the Gap between Semantic Web and Web 2.0 Workshop at the ESWC 2007. 2007, Innsbruck, Austria.
15. Ding, L., et al., Swoogle: A Search and Metadata Engine for the Semantic Web, 13th ACM Conference on Information and Knowledge Management. 2004.
16. Swoogle. Swoogle Semantic Web Search Engine. http://swoogle.umbc.edu/.
17. Terziev, I., A. Kiryakov, and D. Manov, BULO Guidance, SEKT Deliverable 1.8.1. 2005.
18. Fellbaum, C., Wordnet: an electronic lexical database. 1998: The MIT Press.