OntoGame: Weaving the Semantic Web by Online Games

Katharina Siorpaes[1] and Martin Hepp[1,2]

[1]SEBIS, Semantic Technology Institute (STI), University of Innsbruck, Austria
[2]Chair of General Management and E-Business, Bundeswehr University Munich, Germany
katharina.siorpaes@sti2.at, mhepp@computer.org

Abstract. Most of the challenges faced when building the Semantic Web require a substantial amount of human labor and intelligence. Despite significant advancement in ontology learning and human language technology, the tasks of ontology construction, semantic annotation, and establishing alignments between multiple ontologies remain highly dependent on human intelligence. This means that individuals need to contribute time and sometimes other resources. Unfortunately, we observe a serious lack of user involvement in the aforementioned tasks, which may be due to the absence of motivations for people who contribute. As a novel solution, we (1) propose to masquerade the core tasks of weaving the Semantic Web behind online, multi-player game scenarios, in order to create proper incentives for human users to get involved. 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 online game can motivate a large amount of people to work heavily on this task, and this for free. Then, we (2) describe our generic OntoGame platform, and (3) several gaming scenarios for various tasks plus our respective prototypes. Based on the analysis of user data and interviews with players, we provide preliminary evidence that users (4) enjoy the games and are willing to dedicate their time to those games, (5) are able to produce high-quality conceptual choices. Eventually we show how users entertaining themselves by online games can unknowingly help weave and maintain the Semantic Web.

1. Introduction

A pre-requisite for the Semantic Web to become a reality is the broad availability of ontologies and annotation data. However, the knowledge acquisition bottleneck [1] strikes the Semantic Web as it struck other endeavors in the past. Despite significant advancement in tools and semi-automatic approaches, we still need a significant amount of human labor and intelligence for the construction of ontologies, for the annotation of data in various modalities and formats, and for aligning the conceptual elements in multiple ontologies. Making the Semantic Web a reality requires an increase of available metadata by orders of magnitude as compared to the current state. However, we observe that it is hard to motivate people to dedicate their time to those three tasks. At the same time, the amount of Web content in complex modalities (like images, videos, sounds, or Flash applets) and services exposed on the Web is increasing; such is even harder to annotate without the aid of human intelligence.
Obviously, there are still many tasks that most humans can solve easily but state of the art computers cannot [2, 3]. A famous example for such tasks are CAPTCHAs [3]:

challenges related to image analysis that can be used to test whether the user is a human being or a computer agent. Those challenges are employed by many Web applications to block access by unwanted bots and scripts.

Similar to CAPTCHAs, most of the tasks for lifting the current Web to a semantic level remain dependent on human intelligence. Now – why would people want to invest time in building ontologies or annotating content? Clearly, we can observe a sharp contrast in user interest in two branches of Web activity – the "Web 2.0" movement lives from an unprecedented amount of contributions from Web users, while the work on the Semantic Web side is hampered by a substantial lack of user involvement in the aforementioned tasks. In our opinion, this is mainly because Web 2.0 environments provide direct rewards for user involvement, mostly in the form of improved access to Web content [4-6]: Users who tag objects in collaborative tagging systems immediately improve their own access to those objects, while at the same time improving the shared metadata. As for the Semantic Web, many important tasks come without a proper reward for the contributing humans: Building an ontology is a fairly abstract task and thus pretty much decoupled from immediate rewards. Also, heavyweight annotations often require a lot more time from a single skilled individual than this individual will ever save by means of the improved access.

This leaves us with two options for overcoming the lack of ontologies, annotations, and alignments: Either we make a leap in technology so that humans can be eliminated from those tasks. Or we fix the broken incentive scheme for the Semantic Web, i.e., create proper rewards for contributing humans. Luis van Ahn has demonstrated with his already famous games [2, 7-10] that one can exploit computer gaming scenarios for having people contribute human intelligence to actual problems. We adopt his approach for overcoming the key bottlenecks to building the Semantic Web: the lack of people actually dedicating intelligence and judgment for building and maintaining it.

1.1. Related Work

The most popular games with a purpose have been described by **Von Ahn** and colleagues, who have also coined the term *"human computation"*: The ESP game [8] aims at labeling images on the Web. Two players, who do not know each other, have to come up with identical tags describing an image. Peekaboom [7] is a related game for locating objects within images. Verbosity [10] is a game for collecting common sense facts. Phetch [9] is a computer game that collects explanatory descriptions of images in order to improve accessibility of the Web for the visually impaired. Only very recently, Law, von Ahn, and colleagues [11] also came up with a game called Tagatune for music and sound annotation based on tags. **Lieberman** and colleagues describe the game Common Consensus [12], which aims at collecting human goals in order to recognize goals from user actions and conclude a sequence of actions from these goals. Another approach to collecting common sense knowledge is the FACTory Game[1] published by **Cycorp**[2]: FACTory is a single-player online game that randomly chooses facts from the Cyc knowledge base [13] and presents them to the players. The player has to say whether the statement is true, false, doesn't make

---

[1] http://game.cyc.com
[2] http://www.cyc.com

sense, or whether the user does not know. The answers are scored depending on accordance with the majority of answers. Apart from Verbosity, Common Consensus, and FACTory, we do not know of any other work that uses computer game scenarios for the collection of knowledge, and none of those is directly linked to the Semantic Web.

1.2. Contribution and Overview

In this paper, we (1) propose to masquerade the core tasks of weaving the Semantic Web behind online, multi-player game scenarios, in order to create proper incentives for humans to get involved, (2) describe our generic OntoGame platform, and (3) multiple gaming scenarios for various task plus respective prototypes. Based on the analysis of user data and interviews with players, we provide preliminary evidence that users (4) enjoy the games and are willing to dedicate their time to those games, (5) are able to produce high-quality conceptual choices, and show (6) how they may unknowingly help weave the Semantic Web by doing so. Please check our project Web page at http://www.ontogame.org for the first fully-fledged public game and other prototypes. This paper extends our very first overview of experiments described in [14], in which we asked humans to judge whether a particular Wikipedia page primarily describes a set of objects (i.e. a class) or an individual (i.e. an instance).

2.   Multi-Player Games for Weaving the Semantic Web

In the following, we describe multi-player games for subtasks in ontology construction, ontology alignment, and ontology population (annotation).

2.1. Games for Ontology Construction

Ontology construction involves the following five tasks that are hard to delegate to computers:

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: The type of conceptual element according to the distinctions of the applicable ontology meta-model must be determined for each named entity. For example, many popular ontology meta-models support *classes*, *properties*, and *individuals* as core types.

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.

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

Lexical enrichment: Ontology engineering methodologies tend to focus on formal means for specifying ontologies. In order to describe the *intended* semantics of ontology elements, informal means, like natural language labels or synonyms are albeit also needed. However, relating a conceptual element to terms or synonym sets

requires careful human judgment, since otherwise, inconsistencies between the informal part and the formal part of the ontology may result.

In the following, we describe some game scenarios for those tasks.

Table 1. Games for Ontology Construction

Task	Input		Output
	Computational Side	**Human Side**	
Collecting and typing named entities	Users are presented with a class definition.	The players have to come up with and agree upon a label for an attribute its range.	Attributes and their ranges
Typing Named Entities	Users are shown a conceptual entity (e.g. a Wikipedia article).	The players have to agree whether the respective entity represents a class, a property, or an individual.	Meta-model classification of input entities
Adding taxonomic and non-taxonomic relations	Users are shown two classes.	The players have to judge whether one class subsumes the other or to come up and agree upon a label of a relationship between the classes.	Taxonomic relations and labels for other relationship types
Adding taxonomic relations	Users are shown a class.	Users have to come up and agree upon a label for a super-class, i.e. an abstraction.	Classes, taxonomic relations
Lexical Enrichment	Users are presented with one element from an ontology as well as a lexical resource (e.g. WordNet) including the possibility to browse the resource.	The players have to select an entity from the lexical resource, such as a synonym of the class label or a translation.	Links to terms
Modularization	Users are presented with a domain name (from a list of relevant domains) as well as a set of ontological elements.	The players have to define a subset of relevant ontological entities for that domain (and agree on this assignment).	Domain ontology modules

2.2. Games for Ontology Alignment

In an open environment such as the Web, it is likely that multiple, partly overlapping ontologies evolve and are being used. For improved access of the related information, the elements of overlapping ontologies must be aligned to each other; and since ontologies evolve due to conceptual dynamics in domain and advancement of our understanding of the world, such is a continuous effort rather than a one-time task. It's burdensome and never done. Euzenat and Shvaiko [15] distinguish four different techniques of ontology matching: (1) terminological techniques that rely on lexical resources within the ontology, (2) structural techniques that focus on the relations between entities, i.e. ontology elements, (3) extensional techniques comparing extensions of entities, and finally (4) semantic techniques that exploit formalized knowledge.

Despite significant advancement towards automatic matching of ontologies without human intervention, current systems are often not able to perform reliable automatic matching on real-world ontologies yet. The less formal the input ontologies are, the

less likely it is that a machine will ever be able to reliably determine the proper semantic relationships between elements from two different ontologies.

In this paper, we focus on semantic relationships between classes, individuals, relations, and data types. Between such entities, there are different possible types of correspondence, of which the most relevant set-theoretic relations are equivalence ($=$), more general ($\sqsupseteq$), disjointness ($\perp$), and subsumption ($\sqsubseteq$) as described in [15]. We think that the following tasks are particularly suited for the representation as game scenarios:

Equivalence of classes, relations, attributes: Indicating whether two classes or properties are equivalent, based on the label, a description, and additional lexical resources.

Subsumption between classes: Indicating whether a class is a sub-class of another class.

Table 2. Games for Ontology Matching

Scenario (Task)	Input		Output
	Computational Side	**Human Side**	
Matching classes	Players are faced with the two concepts c1 from ontology A and c2 from ontology B and a set of possible mapping relations.	Players have to select and agree on the most appropriate relation between the concepts.	Alignments
Matching classes	Players are presented with concepts c1 from ontology A and the complete subsumption hierarchy of ontology B plus the set mapping relations.	Players have to select the most specific corresponding class in ontology B, the appropriate relation between the concepts, and agree on both choices.	Alignments

The tasks in ontology matching were outlined in the previous section. In literature, equivalence ($=$), subsumption ($\sqsubseteq$), and disjointness ($\perp$) are described as the most important matching relations. Thus, we do not only want to know from our players whether two classes are the same but we want to know the kind of relation that exists between them. In our games (Table 2) we let players choose from a set of possible relations. Furthermore, one has to keep in mind that our goal is to attract as many users as possible to play in order to create a wealth of data, even if only lightweight. Therefore, we decided to make use of SKOS [16] relations: SKOS (Simple Knowledge Organization System) core is a lightweight meta-model that describes just the minimal set of classes and properties that are necessary to express knowledge in simple structures. We have preliminary evidence that players are able to understand the meaning of SKOS relations, such as broader or narrower, more easily than the precise meaning of subClassOf [17]. Thus, we use the following relations for matching ontologies: (1) equivalent ($=$), (2) broader: a concept that is more general in meaning, (3) narrower: a concept that is semantically narrower in some sense, (4) related: a concept with which there is an associative semantic relationship, (5) partly overlapping with: there is an overlap in meaning between these concepts, (6) strict subClassOf; this relation is intended only for expert games, (7) Not related: disjointness ($\perp$).

2.3. Games for Semantic Annotation

Generally, all annotation scenarios require (1) a resource, e.g. a Wikipedia article or a media object, and (2) an ontology, e.g. the Proton ontology. The players are then asked to annotate the resource using the given ontology (Table 3). For each consensual aspect, both players will earn points. In many cases it will be necessary to hide the ontology behind a graphical user interface or natural language patterns in order to increase the game fun as well as the comprehensibility of the task. Candidate resources that are vastly available on the Web are textual resources, images, videos, sounds, software, and Web services. The (semi-) automatic annotation of multimedia content is especially challenging for a machine; however, this is a task that can often be easily done by a human actor. Thus, we see an especially large potential in turning multimedia content annotation into games. Additionally, games that involve music, pictures, or videos are more enjoyable for players. Another potential application area, which will not be addressed in this paper, is the annotation of Web services.

For annotation games, we depend on the availability of sufficiently detailed (domain) ontologies, which can be a bottleneck as of today. This is why we aim at interweaving games for annotation with games for ontology construction.

Table 3. Games for Semantic Annotation

Scenario (Task)	Input		Output
	Computational Side	Human Side	
Annotation	Players are shown a resource, which can be text or multimedia content, and a suitable (domain) ontology.	Players have to select and agree on the appropriate annotation of the resource.	Semantic Annotations

3. OntoGame: A Generic Game Infrastructure

In order to keep up interest, the set of available games should be changed or the games being updated frequently. Also, the resulting data from past games should be stored in a generic format so that we can run statistical analyses when deriving ontologies, annotations, or mappings from consensual games. Note that the games do not directly return the correct modeling; moreover, we will use an appropriate threshold of consensual, matching rounds that must confirm a particular modeling choice before it is assumed to be correct.

The heart of our OntoGame is a generic game infrastructure that allows to plug-in various scenarios with minimal modifications. All user inputs and results are stored in RDF for simple analysis and reuse. The user interface is designed in a way that in can be easily adjusted to a new scenario.

3.1. System Description

Each OntoGame is an online, multi-user game where players play in teams of two: these teams are selected randomly and anonymously. The players have no means to communicate with or identify the counterpart. This is important in order to avoid cheating or false input, which will be discussed in detail in a later section.

In all game scenarios, users are faced with a task, e.g. matching two classes or finding a suitable abstraction of a Wikipedia article in a given ontology. The players have to reach consensus on their choice in order to earn credits. After each choice, both players get feedback about what their partner's choice was, regardless of whether they reached consensus or not.

Before using the system, each user has to register. It is desirable to have users login with the same username every time they play because of two reasons. First, **competition:** users can build a reputation in the system and work on their rank, which constitutes an additional incentive to play games [6]. Second, **reliability:** if users have a history of good, meaningful game rounds, their judgment is more reliable than that of others. This can be exploited when deriving formal content and when to spot cheating. Upon pressing a "play!" button, the user is randomly paired with another player and the game starts. In case there is not an even number of users on-line, a single-player mode is started; this alternative remains invisible to the user, though. In single-player mode, users play pre-recorded challenges as if playing with a real partner.

Players can skip a step (Fig. 2) and abort the current challenge in the games; the team will then proceed with a new challenge. At the moment, this feature follows the principle of consensus as well: only when both players decide to skip, they are taken to the next round. Skipping is an important feature, because it is possible that poor or incomprehensible challenges are given, for which players may simply be unable to produce consensual solutions. Instead of encouraging random guesses, we rather motivate users to proceed to a new challenge.

3.2. Implementation

The OntoGame platform (Fig. 1) is a client-server infrastructure based on Java. The game server runs on a Apache Tomcat 5.5[3] server together with the RDF repository Sesame[4] [18] and servlets. The game server connects to the repository via a database connector and runs the servlets. The servlets connect to the client via an object stream over an HTTP tunnel. The controller runs the graphical user interface. 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 the game, handling user input, matching, and skipping.
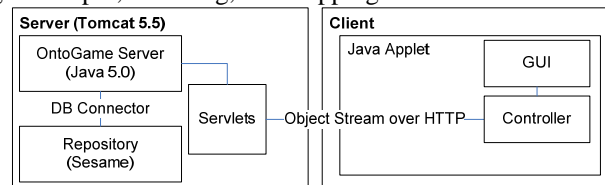

Fig. 1. OntoGame Platform

---

[3] http://tomcat.apache.org/
[4] http://www.openrdf.org/

3.3. Cheating

One may argue that cheating and other forms of destructive user behavior endanger the quality of the game output. However, von Ahn has already shown that the impact of cheating can be minimized yet by several simple mechanisms. We follow his suggestions and use the following techniques: First, the players are paired anonymously and have no way to communicate with each other. Second, we check whether the IP addresses of partners are different, so one cannot simply run the game on the same machine multiple times and hope for being teamed up with oneself. Third, simple cheating strategies like always choosing the first option or enter pre-agreed words as text input can be detected rather easily by having them play one challenge for which the correct result is known. If the consensual solution to such a challenge is different from the set of known solution, user input from both players will be ignored when deriving formal content. Also, one can monitor the response times and assume bots when they are significantly lower than the average.

We are also considering more sophisticated reputation mechanisms for future releases.

4. Four Cool OntoGame Scenarios

In the following, we describe four game scenarios for weaving the Semantic Web that we consider most promising and that address real-world problems, such as searching videos or product search in e-Bay. The first two ones are already released to the general public. The two others are design studies for which the implementation is underway.

4.1. Turning Wikipedia into a Huge Domain Ontology with Proton Grounding

In this game, we show the first paragraph from a randomly selected Wikipedia page. By Wikipedia convention, this is almost always a reliable excerpt of the page content. Then, we ask the user to select whether this Wikipedia entry rather describes a set of objects (i.e., a class) or a significant single object (i.e., an individual), see Fig. 2[5]). If both players agree on that choice, they proceed to the next level. In this level, they have to agree upon the most specific class of the Proton ontology [19] of which the Wikipedia entry is a subclass or instance (see Fig. 3). The use of Proton is mainly motivated by two factors. First, we needed a general-purpose ontology that would make sense as an upper-level ontology above all Wikipedia entries. This ontology should already contain sufficient specializations so that the difference in the level of abstraction as compared to Wikipedia URIs was appropriate for average users. In the future, we will also consider upper ontologies such as DOLCE[6] or SUMO[7].

The deeper the teams manage to go into the hierarchy, the more Wikipedia articles they play, and the more Proton abstractions they find within 2 minutes, the more points they are awarded. For the moment, we do not make use of the Wikipedia

---

[5] Larger screenshots are available at http://www.ontogame.org.

[6] http://www.loa-cnr.it/DOLCE.html

[7] http://www.ontologyportal.org/

category system due to its diverging and unstructured nature, but may use this in the future for suggesting suitable Proton choices.

The motivation for this game is that the URIs of the more than 1.8 Million Wikipedia entries are reliable identifiers for countless useful conceptual entities [20]. For example, Wikipedia contains more than 220,000 URIs for types of products and services and is thus eight times larger than eCl@ss or UNSPSC, the two largest categories for products and services. If we are able to ground those 1.8 Million conceptual elements properly in the Proton ontology, we will create the largest general interest ontology for annotating Web resources – 1.8 Million identifiers for anything from artists to high schools, from products to organizations. This game is online for playing by the general public at http://www.ontogame.org.

Fig. 2. Ontologizing Wikipedia: Step 1

Fig. 3. Ontologizing Wikipedia: Step 2

Example

Alice and Bob play the game: they both see an excerpt of the Wikipedia article about Lupicinus[8]. They first have to agree on whether the most important ontological role of Lupicinus is to be a class or an instance. Alice and Bob agree on instance (because it is an instance of Person), get 20 points and are taken to the next step. Here they are shown the first level of the Proton ontology, which divides things into *abstracts, happenings,* or *objects*. Alice and Bob both agree on *object*, get 10 points and are taken into the object branch of Proton. Here they agree on *agent* and are awarded 20 points and are taken even deeper in the Proton hierarchy. Our players both choose *person* in the next level, get 30 points, and finally agree on the Proton class *man*, receiving 40 points. The round ends here and they are taken to the next randomly chosen article. This continues until the time of 2 minutes is over.

4.2. Annotating YouTube Videos

The objective of this scenario (see Fig. 4 and 5) is to annotate YouTube[9] videos. It is inspired by Jim Hendler's comment at last year's ISWC's panel discussion that search in YouTube (and videos in general) was a key application of semantic search.

In order to annotate YouTube videos in games, we specified a simple domain ontology that describes the content of videos. The relevant standard for the description of multimedia is MPEG-7[10]. We also took IMDB (Internet Movie

---

[8] http://en.wikipedia.org/wiki/Lupicinus

[9] http://www.youtube.com

[10] http://www.chiariglione.org/MPEG/standards/mpeg-7/mpeg-7.htm

DataBase)[11] into consideration as input. IMDB has a huge user base and we are interested in what users are searching for when they search for videos. Therefore, we had a close look at the search interface of IMDB in order to model a very simple video content ontology. Our approach to annotating YouTube videos is to start with a very lightweight conceptual model, which will be extended once the game will have generated a wealth of data. From the ontology, we derived a set of challenges that are posed to users. This game is online for playing by the general public at http://www.ontogame.org. We are currently integrating the ontology produced by the first scenario in this game. We are also considering how we could exploit the existing YouTube tags.

Describing Video Content
Both players are shown the first ten seconds of a randomly chosen YouTube video with the option to view further parts of the video. Then they are presented with challenges: each time the players agree on an answer based on a predefined ontology, they are taken to the next level. Again, the number of points players can earn increases with the number of mastered levels. Also, the total amount of time available is two minutes.

Fig. 4. Annotating YouTube: Level 1

Fig. 5. Annotating YouTube: Level 2

The set of challenges for each video is:
1. The video is: Non Fiction or Fiction.
2. The video's color is: black/white or color.
3. The video's genre can be best described as: {list of 27 genres ranging from action over drama western as used by IMDB}
4. Generally, the video is about: {set of topics; the players can take multiple guesses}
5. The language of the video is: {list of languages including option "no language"}
6. The location of the video is: {set of countries and locations; the players can take multiple guesses}
7. The time period the video plays in is: {users enter the earliest and latest covered year or decade}
8. The video was produced: by a private person or by a company.

Example
The first video presented to Alice and Bob is a video where Tim Berners-Lee is speaking about the Semantic Web. They agree on that the video is non-fiction (+10 pts.). On the next level, they quickly agree that the video is color (+10 pts.). Next,

---

[11] http://www.imdb.com

they consensually choose "Scientific Talk" as genre (+30pts.). On the following level, Alice selects that the video is about Tim Berners-Lee while Bob selects "Web" (none of them can see the other's guesses). Next, they both enter "Semantic Web" and get 40 points. After specifying the language of the video as English, they can not reach agreement on the location and thus decide to skip and go to the next video. (This continues until the time of two minutes is up.)

4.3. Mapping UNSPSC and eCl@ss

UNSPSC and eCl@ss are the two most important categorization standards for products and services, and establishing mappings between them for achieving data interoperability is one of the long-lasting target applications of semantic technology [21]. In this game scenario (Fig. 6), we have humans weave a net of semantic alignments between classes in both standards. Players are faced with a randomly chosen class from UNSPSC, as well as a set of possible relations, and the eCl@ss tree. In each step, the players have to agree on a class from eCl@ss and the kind of relation between the UNSPSC class the eCl@ss one. Before choosing a branch in eCl@ss, players can open the branch and see sub-classes in order to get a better understanding of the branch they are choosing. Choosing multiple classes is allowed. As described in an earlier section, we use the matching relations "same as", "narrower than", and "partly overlapping with".

Fig. 6. Mapping UNSPSC and eCl@ss

Fig. 7. Annotating eBay with eCl@ss

4.4. Annotating eBay with eClassOWL

The objective of this game (see Fig. 7) is to annotate offerings in eBay auctions with the product categories and product properties in eClassOWL [22]. For this purpose, we randomly select eBay auctions and present them to the players. The players are provided with a tree view of the eClassOWL ontology. Similar to the scenario where Wikipedia articles where annotated with the Proton ontology, the players have to choose a class from eClassOWL and reach agreement on this choice. In most cases, classes on a high level of eClassOWL will have several sub-classes, where the first step is repeated: the deeper the players manage to get in the hierarchy, the more points they are awarded. In many cases, it will increase gaming fun and quality of results when the system comes up with a suggestion for a branch of eClassOWL that is likely to fit. Therefore, we will investigate in how far we can (1) use matching algorithms in the background and (2) make use of the eBay category system. Also, the attributes of eClassOWL may be considered in future games.

5. Evaluation

While the last three scenarios are currently prototypes with still limited scalability, scenario 1 has been made available to the general public recently. In the following, we summarize our evaluation of the approach based in this scenario. First, we checked the data produced by the game for ontological correctness. Second, we conducted interviews among all participants who played the game in order to find out about the fun factor of the game.

5.1. Methodology

We invited 33 individuals in 5 groups with different backgrounds and asked them to play OntoGame for a duration of between 10 and 20 minutes. We asked each group to play at the same time to ensure that there were enough players to play OntoGame. Each individual of one group was asked to play separately in order to evaluate the single player mode and to verify the results of already played games. Only very few had experience with building ontologies due to their professional background (research). During most experiments, the participants were in different rooms and did not communicate with each other during playing. In two cases, the groups were in one room. However, we supervised the experiments and made sure they did not communicate with each other. The game was explained to the participants briefly before playing it online.

All of the games were logged. After the game, we interviewed participants about their experiences with the game and analyzed the output of the recorded games. They were asked the following questions:

1. Were the rules of the game hard to understand?
2. How do you rate the challenge of the game? ("OK", "too easy", "too hard")
3. Was it fun to play the game?
4. What did you especially like/dislike about the game?
5. Would you play it again?
6. General feedback.

5.2. Results

The results of our preliminary evaluation are encouraging, as summarized below.
Quality of Results: 27 individuals actually played the game. 170 Wikipedia articles were played by different players in 825 games, i.e. some pages were played multiple times. Players decided to skip directly and proceed to the next article in about 11% of the games. We took the remaining 733 games as a basis for our evaluation and analyzed (1) how many were correct regarding the choice class vs. instance and (2) regarding the abstraction in Proton, and (3) how many and (4) which mistakes were made (Table 4). For this purpose, we manually analyzed the data generated by the games.
Excluding those challenges that were skipped immediately (n=92, 11%), our players were able to agree on both the ontological nature and a Proton class in almost 80% of the cases (n=586). Of these tasks that were completed consensually more than 99% (n=581) were semantically correct. Of the challenges for which the player agreed on

class vs. instance only (n=147 of 733), 99% of choices (n=142) were correct. In a nutshell, we can see that if consensus is reached, it largely represents correct choices.

Table 4. Summary of Results

	Criterion	Number	Percent
1	**General**		
1.1	Number of Wikipedia pages that were played at least once	170	-
1.2	Total number of challenges played	825	-
1.2	Challenges that were not skipped and actually played	733	88.85%
2	**Consensus**		
2.1	Challenges in which only the first task was completed consensually	147 of 733	20.05%
2.2	Challenges in which both tasks where completed consensually	586 of 733	79.95%
2.3	Challenges in which both tasks were completed consensually, and the consensus was at the leaf level of Proton	405 of 733	55.25%
3	**Conceptual Quality of the Consensual Solutions**		
3.1	Amount and ratio of challenges in 2.1 of which the consensual choice for only task 1 was correct	142 of 147	96.60%
3.2	Amount and ratio of challenges in 2.2 of which the consensual choice for tasks 1 AND 2 were correct	581 of 586	99.15%
3.3	Amount and ratio of challenges in 2.3 of which the consensual choice for tasks 1 AND 2 were correct	404 of 405	99.75%
4	**Mistakes**		
4.1	Total of wrong choices	10	-
4.2	Wrong judgment of ontological nature	5	-
4.3	Wrong abstraction	5	-

Only a marginal amount of the consensual choices games were conceptually wrong. The following mistakes were made:

Class vs. of instance: In one case, players classified an article as a class while it was an instance (a person). Four Wikipedia articles were categorized as instances while they were classes. We are aware that the judgment whether the dominant ontological role of a conceptual entity is a class or an instance is sometimes subjective.

Wrong abstraction in Proton: In the remaining cases, the teams chose wrong abstractions in Proton, i.e. a park was classified as *abstract* while it is a *location* or a bank classified as a *service* while it is an *organization*.

While this tentative assessment is encouraging, it is currently a very preliminary evaluation. In particular, the extremely high conceptual reliability may have been caused by a substantial amount of single-player games which used recorded game-scenarios. Since the amount of recorded game-scenarios was initially small, the share of correct solutions based on us researchers playing the game may have been higher than in a large-scale deployment. However, a more comprehensive analysis is already in preparation and in principle confirms the first assessment.

Fun Factor: We received very positive feedback from the participants: surprisingly, those without any background in computer science enjoyed playing the game especially. In earlier experiments many participants experienced problems to grasp the distinction between class and instance caused troubles. Therefore, we changed the descriptions in the game to make it more understandable. Almost all participants confirmed that the rules of the game were easy to grasp.

More than 80% found the game challenging enough, all of them described the time pressure and the variety of concepts in the Proton branches as challenging. Four

participants found it too easy. Six participants mentioned that in the beginning the game was too hard when one does not know the Proton ontology. Furthermore, they indicated that abstract Wikipedia articles were hard to classify. However, they also indicated that they enjoyed learning the Proton ontology and hence increased their playing pace. 21 players liked the game and said it was enjoyable to play. Six found it neither especially exciting nor especially boring. Two said that they found it boring. 19 stated that they would play the game again. Seven participants mentioned that they liked making sense of a rather short excerpt of the Wikipedia article. The majority described the second step of the game, i.e. matching the article to a Proton class, as the most fun part of the game.

Almost all participants enjoyed playing with a human counterpart and liked the consensus component of the game. Two mentioned that they would have liked to know who they were playing with. We are therefore working on a functionality that gives an additional reward to the players in form of information that is revealed about their partner (e.g. gender or nationality). Fifteen participants perceived the ranking of players displayed in the beginning of each game as a motivation to further improve their abilities and thus status in the system.

6.   Conclusion and Outlook

In this, paper we proposed to masquerade the core tasks of weaving the Semantic Web behind online, multi-player game scenarios, in order to create proper incentives for humans to get involved. We presented game scenarios that in combination have the potential to increase the amount of ontologies, annotations, and alignment data in the Semantic Web substantially. If only 1,000 individuals in the world will play our games 1 hour per day for three months, this will mean 90,000 hours of volunteer work; something that would otherwise cost about a million euro at an hourly rate of 11 euro – and few experts in the Semantic Web will work for 11 euro per hour.

Based on the analysis of user data and interviews with players, we provide preliminary evidence that users enjoy the games and are willing to dedicate their time to those games and are able to produce high-quality conceptual choices.

Each of the scenarios addresses a real-world problem: Annotating Wikipedia does not only help to learn Proton and learn new topics from randomly selected Wikipedia pages, but it will allow help extend Proton to make it one of the biggest domain ontologies in the world. Annotating video content will make the vast amount of content for entertainment and education available at video portals such as YouTube accessible to search at the semantic level. Using games for creating alignments between eCl@ss and UNSPSC has the potential to mitigate one of the most substantial data interoperability problems in the product data domain. Annotating eBay offerings with references to eClassOWL will help make the vision of Semantic Web-based e-commerce a reality. Please play OntoGame @ www.ontogame.org, and help weave the Semantic Web!

References

[1]   Wagner, C., "Breaking the Knowledge Acquisition Bottleneck Through Conversational Knowledge Management", Information Resources Management Journal, 19(1): pp. 70 -83, 2006.

[2]   Von Ahn, L., "Games with a Purpose", IEEE Computer, 29(6): pp. 92-94, 2006.

[3]   Von Ahn, L., M. Blum, N. Hopper, and J. Langford, "CAPTCHA: Using Hard AI Problems for Security", International Conference on the Theory and Applications of Cryptographic Techniques, Springer LNCS, Warsaw, Poland, 2003.

[4]   Hotho, A., R. Jaeschke, C. Schmitz, and G. Stumme, "BibSonomy: A Social Bookmark and Publication Sharing System", Conceptual Structures Tool Interoperability Workshop at the Conference on Conceptual Structures, Aalborg University Press, 2006.

[5]   Mika, P., "Ontologies are us: A unified model of social networks and semantics", International Semantic Web Conference (ISWC2005), Springer LNCS, Greece, 2005.

[6]   Marlow, C. M. Naaman, D. Boyd, and M. Davis, "Tagging, Taxonomy, Flickr, Article, ToRead", World Wide Web Conference (WWW06), ACM, Edinburgh, Scotland, 2006.

[7]   Von Ahn, L., "Peekaboom: A Game for Locating Objects in Images", Conference on Human Factors in Computing Systems (CHI 2006), ACM, Montreal, Canada, 2006.

[8]   Von Ahn, L. and L. Dabbish, "Labeling Images with a Computer Game", Conference on Human Factors in Computing Systems (CHI 2004), ACM, 2004.

[9]   Von Ahn, L., S. Ginosar, M. Kedia, R. Liu, and M. Blum, "Improving Accessibility of the Web with a Computer Game", Conference on Human Factors in Computing Systems (CHI 2006), ACM, Montreal, Canada, 2006.

[10]   Von Ahn, L., M. Kedia, and M. Blum, "Verbosity: a game for collecting common sense facts", Conference on Human Factors in Computing Systems (CHI 2006), ACM, Montreal, Canada, 2006.

[11]   Law, E., L. von Ahn, R. Dannenberg, and M. Crawford, "Tagatune", International Conference on Music Information Retrieval (ISMIR 2007), Vienna, Austria, 2007.

[12]   Lieberman, H., D. Smith and A. Teeters, "Common Consensus: A Web-based Game for Collecting Commonsense Goals", Workshop on Common Sense for Intelligent Interfaces, ACM Conference on Intelligent User Interfaces (IUI-07), Honolulu, 2007.

[13]   Lenat, D.B. and R.V. Guha, "Building Large Knowledge-based Systems: Representation and Inference in the Cyc Project", Addison-Wesley, Boston, 1990.

[14]   Siorpaes, K. and M. Hepp, "OntoGame: Towards Overcoming the Incentive Bottleneck in Ontology Building", International IFIP Workshop On Semantic Web & Web Semantics (SWWS '07), OTM conferences, Springer LNCS, Vilamoura, Portugal, 2007.

[15]   Euzenat, J. and P. Shvaiko, "Ontology Matching", Springer 2007.

[16]   W3C, "Simple Knowledge Organisation System (SKOS)", http://w3.org/2004/02/skos/.

[17]   Hepp, M., "Possible Ontologies: How Reality Constrains the Development of Relevant Ontologies", IEEE Internet Computing, 11(7): pp. 96-102, 2007.

[18]   Broekstra, J., A. Kampman, and F. Van Harmelen, "Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema", International Semantic Web Conference (ISWC2002), Springer LNCS, 2002.

[19]   SEKT Consortium, "PROTON Ontology", http://proton.semanticweb.org.

[20]   Hepp, M., K. Siorpaes, and D. Bachlechner, "Harvesting Wiki Consensus: Using Wikipedia Entries as Vocabulary for Knowledge Management", IEEE Internet Computing, 11(5): pp. 54-65, 2007.

[21]   Schulten, E., H. Akkermans, N. Guarino, G. Botquin, N. Lopes, M. Doerr, and N. Sadeh, "The E-Commerce Product Classification Challenge", IEEE Intelligent Systems, 16(4): pp. 86-89, 2001.

[22]   Hepp, M., "eCl@ssOWL", http://www.heppnetz.de/eclassowl/