



Ludo: An Ontology to Create Linked Data Driven Serious Games

Oscar Rodríguez Rocha, Catherine Faron Zucker

► To cite this version:

Oscar Rodríguez Rocha, Catherine Faron Zucker. Ludo: An Ontology to Create Linked Data Driven Serious Games. ISWC 2015 - Workshop on LINKed EDucation, LINKED 2015, At Bethlehem, Pennsylvania, United States, Oct 2015, Bethlehem, Pennsylvania, United States. hal-01188202

HAL Id: hal-01188202

<https://hal.inria.fr/hal-01188202>

Submitted on 11 Jul 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Ludo: An Ontology to Create Linked Data Driven Serious Games

Oscar Rodríguez Rocha¹, Catherine Faron Zucker²

¹ INRIA Sophia Antipolis Méditerranée, Sophia Antipolis, France.
oscar.rodriguez-rocha@inria.fr

² Univ. Nice Sophia Antipolis, CNRS, I3S, UMR 7271, Sophia Antipolis, France
faron@unice.fr

Abstract. Given the increasing amount of structured data published on the Web, many possibilities are open for creating new types of games that use resources from the Web of Data. In particular, if we consider the subcategory of Serious Games in which the objective of the game is to educate the user through the interactive discovery of real-life concepts (associated to Semantic Web resources), the inclusion of a semantic representation of the user profile and his contextual information becomes an important element to recommend educational resources to the player. This paper proposes and describes an ontology that allows the description and representation of Serious Games with those characteristics. The potential of the ontology is shown through the use-case of a serious, Web-based and question-based board game prototype. This work was carried out in the context of the Semantic Educloud project.

1 Introduction

Although Serious Games have been around for a long time (as shown in [1] where the authors present a discussion on the historical origins of this term), their origin is often attributed to Sawyer and Rejeski, which in their white paper [2] published in 2002 proposed to use entertainment video games as learning tools to educate people about the potential outcomes and effects of public policy. We adopt the definition of Serious Games given by [3]: “games that do not have entertainment, enjoyment, or fun as their primary purpose”.

The increasing amount of structured data published on the Web according to the Linked Data principles opens many possibilities for creating new types of Serious Games. Thanks to the efforts of the scientific community and the W3C LOD project³, in the last few years more and more data have been published on the Web, resulting in a global data space called the Web of Data [4]. Its growth has been measured by the LOD cloud diagram⁴, which shows how in the first measurement in 2007 there were only 12 datasets compared to the 570 found in the last measurement (at the time of writing) published in 2014.

³ <http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>

⁴ <http://lod-cloud.net>

Among the different uses of these linked datasets, they can be exploited as knowledge bases to build games. In particular, if we consider the subcategory of Serious Games in which the objective of the game is to educate the user through the interactive discovery of real-life concepts, the inclusion of a semantic representation of the profile of the the game player and his contextual information become important elements to enhance the recommendation of educational resources. To allow this data integration and knowledge organization, this paper proposes an ontology that enables the description and representation of Serious Games with such characteristics. The potential of this ontology is demonstrated through the prototype of a serious, Web-based and question-based board game, which exploits the DBpedia dataset and vocabulary to propose the user questions with multiple answers. Our ultimate aim is to develop such a serious game within the Semantic Educloud project, in which the research work described in this paper is carried out.

The remainder of this paper is structured as follows: In section 2, we present the state of the art on Semantic Web based Serious Games. In section 3, we present and detail the ludo ontology we propose. In section 4 we describe a prototype of serious question-based game we have developed based on ludo. In section 5, we describe the Semantic Educloud project. Finally conclusions and future work are presented in Section 6.

2 Related Work

There are different categories of research works that integrate Serious Games and Semantic Web.

We can find some works that propose Serious Games for “knowledge acquisition”. That is the case of those that contribute to motivate and encourage end users to provide their knowledge to enable the creation or enrichment of an ontology. Ontogames [5–7] 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”. An example of an Ontogame can be found in [8], where the authors present *SeaFish*, “a game for collaborative image annotation and interlinking without text”. Other Serious Games were created to rank facts in knowledge bases as well as to find inconsistencies in a given knowledge base. WhoKnows? [9] is an online quiz that generates different kinds of questionnaires from DBpedia data sets. It uses the approach of “Games with a Purpose” [7] to detect inconsistencies in Linked Data and score properties to rank them for sophisticated semantic search scenarios. Similarly, RISQ! [10], is a Jeopardy!-like quiz game with questions automatically generated from LOD facts to gather ranking information for persons to provide a basis for the evaluation of semantic ranking heuristics. Unlike the works proposed in this category, our goal is the knowledge representation and reasoning in the domain of Serious Games to facilitate and improve a further resource recommendation process.

Some other works focus on creating Serious Games from models and ontologies. In [11] a preliminary metamodel is presented to facilitate the development of educational games; however the authors neither present a game prototype (that allows them to verify their model) nor an ontology for modeling games of this kind. A very inspiring work is presented in [12]: authors propose a game content model that can help game designers to document specifications of game design. As it will be detailed in this paper, the basis of our ontology is composed by the concepts described in this model. Furthermore, our work provides the means to capture the context and the profile of the player to enable recommendation of educational resources on top of them.

With regard to data representation in games and Serious Games, the authors in [13] present six prototypes of games that represent either structured data (from the Web of Data) or Open Data (from government, companies and organizations) within the game. Similarly, authors in [14], present a work that simulates the terrain of a Great War battle by using data from different Linked Data sources. The examples presented in these works on how to exploit structured data to create games have inspired us to create an ontology for this domain, as none of these works propose one.

3 An Ontology for Serious Games

The ludo ontology we propose in this paper has been designed (and is not limited to) to enable the modelization and further creation of Serious Games that use Linked Data datasets as a knowledge base to represent resources within the game. It also allows to model the player profile and his context within the game, making possible to generate more personalized recommendations of related resources in order to stimulate the learning process while playing. Figure 1 shows an overview of the ludo ontology and its main concepts. Classes and properties shown in yellow are further described in 3.1, while the ones in red, blue and green in subsections 3.2, 3.3 and 3.4 accordingly.

Ludo is published online in RDF format⁵, and its LOD-generated⁶ documentation is available as well⁷.

3.1 Formalization of the Serious Game

The ludo ontology is based on the work presented in [12], in which the authors describe a Game Content Model (GCM) that “represents a serious games ontology from an interactive content viewpoint”. In order to represent the rules, play and aesthetic information of a game, GCM comprises ten key concepts and sub-concepts. We have formalized them in OWL 2 as the basis of the ludo ontology.

⁵ <http://dbpedia-test-fr.inria.fr/ludo/ludo.rdf>

⁶ <http://www.essepuntato.it/lode>

⁷ <http://www.essepuntato.it/lode/http://dbpedia-test-fr.inria.fr/ludo/ludo.rdf>

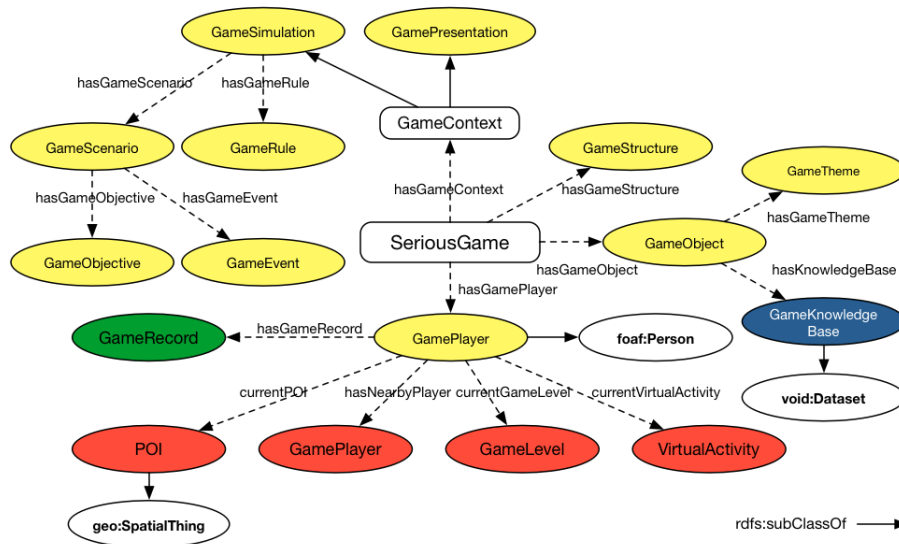


Fig. 1. Overview of the ludo ontology

GameStructure The architecture and flow of the game. It is subsequently divided into game sections that represent segments of the game. Each game section is composed of a **GameContext**, a **PedagogicEventIndicator** and an **EventTrigger**.

GamePresentation A virtual canvas with components of type **MediaComponent** or **GUIComponent** forming a game menu, a game notification or a cut scene to present information about the game and allow to navigate through the game structure.

GameSimulation The mechanism that recreates scenarios virtually, for game play. A simulation is controlled by a set of rules that define the interactivity, physical and temporal properties of the virtual world. It has **GameRules**, a **GameDimension**, a **GameTempo**, **GamePhysics** and a **FrontEndDisplay** used to display information about the simulation.

GameRule States the relationships between game objects and the game world and the effect of an interaction. GCM proposes that a rule can be either a **GameScoringRule** that defines what to be awarded to the game player when a **ScoringCondition** is met or a **GameInteractionRule** that dictates the **Outcome** of the interaction from two game objects.

GameScenario A situation which requires the game player to overcome a number of challenges in order to achieve the defined objects. A game scenario is described by a **GameEnvironment**, a set of **GameEvent**, a set of **VirtualCamera**, a **DifficultyIndicator** and a set of **GameObjective**.

- GameEvent** The happening associated to a game scenario. A `GameEvent` is composed of a set of `GameAct` and an `EventTrigger`.
- GameObjective** The goal associated to a game scenario. A `GameObjective` is described using a `GoalCondition`.
- GameObject** A virtual thing that populates the game world and can be designed to have a combination of abilities such as decision making, moving, acting and responding to surroundings and game player's input simulating their existence in the game world. A `GameObject` has as a set of `ObjectAttribute`, an `Appearance`, an `Intelligence`, it can perform `Action` and has a thematic description describing its aesthetic representation.
- GamePlayer** A user of the game who provides inputs as part of the gaming activity. A `GamePlayer` is represented as an entity with an `Avatar`, `GameAttributes`, an `Inventory`, `GameControl` and `GameRecord`. As it will be explained in Section 3.2, our proposal foresees extending this representation through the inclusion of concepts from the FOAF vocabulary⁸, in order to capture a more detailed representation of the player's profile and the introduction of some others to have a more detailed representation of the player's context within the game.
- GameTheme** The art requirements related to the game through expressive written text.

3.2 Formalization of the Player's Profile and Context

We extended the way in which the game player profile information is represented in GCM by using the FOAF vocabulary as it has been specifically designed to represent persons, their activities and relationships with others. The `GamePlayer` class is declared to be a subclass of `foaf:Person` and the following FOAF properties are used to describe a player:

- `foaf:givenName`, `foaf:familyName` and `foaf:nick` for his names,
- `foaf:age` for his age,
- `foaf:depiction` for the avatar image of the player.
- `foaf:knows` to indicate a reciprocated interaction with another player,

Considering the fact that while playing, the user generates a lot of information about the situation in which he is, and that this information can be very useful, especially for content recommendation scenarios, we decided to model it in the ludo ontology. Our modeling is strongly inspired by the way in which the PRISSMA [15] vocabulary handles contexts. The ludo ontology foresees four elements to represent the context of the player:

Virtual location The current virtual location of the player, refers to the correspondence between a virtual place (represented in the game) and a place from the real life. For example, if the player is in an augmented reality game

⁸ <http://www.foaf-project.org>

whose objective is to explore the museum of Louvre, we can define his current virtual location as Paris, geographic coordinates and area (radius). ludo introduces the concept of point of interest through the **POI** class, declared as a subclass of **geo:SpatialThing** from the *Geo*⁹ vocabulary. To describe locations, ludo reuses the **geo:lat** and **geo:long** properties of **geo:Point** and introduces property **radius** to associate a radius around a POI, property **poiCategory** to associate a category (like monument or museum) to a POI, and property **currentPOI** to associate a player with a POI.

Virtual nearby players ludo allows to represent the fact that, during the gameplay, the player gets in geographical proximity to one or more other players. For this, the **hasNearbyPlayer** property between two players (with **GamePlayer** for domain and range) allows to establish the proximity between players, considering also the geographical distance between them.

Game level The ludo ontology allows to represent game levels through the class **GameLevel** and to associate them with a number and a description by using the properties **hasGameLevelNumber** and **hasGameLevelDescription**. As it is important to consider the level of the game reached by the player, for instance to adapt the resources which will be recommended to him, ludo provides the property **currentGameLevel** that has a game player as domain and a **GameLevel** as range.

Virtual activity Models a representation of the current action being performed by the player, e.g., learning, reading, exploring. A game action can be modeled with the class **VirtualActivity**, it can be associated to a description through the **hasVirtualActivityDescription** property and associated to a player through the **currentVirtualActivity** property.

3.3 Game Knowledge Bases

The ludo ontology allows to model Serious Games that require to communicate with one or more datasets from the Web of Data, mainly to:

- retrieve useful information that can be represented within the game through virtual objects,
- store and access the configuration parameters and settings of the game,
- access to player profiles,
- generate recommendations of additional content or resources for the player.

By incorporating the VoID (“Vocabulary of Interlinked Datasets”) vocabulary¹⁰, the ludo ontology allows to describe such datasets, their characteristics and information to access them through the following classes:

void:Dataset Represents a dataset that is published and maintained by a single provider, available as RDF, and accessible through HTTP URIs or a SPARQL endpoint.

⁹ <http://www.w3.org/2003/01/geo/#vocabulary>

¹⁰ <http://www.w3.org/TR/void/>

void:Linkset Is a subclass of **void:Dataset**, used for storing triples to express the interlinking relationship between datasets.

In the ludo ontology, the **GameKnowledgeBase** class is defined as a subclass of **void:Dataset** and property **hasKnowledgeBase** is defined with class **GameObject** as domain and **GameKnowledgeBase** as range.

3.4 Monitoring of the Player Learning

In order to monitor the learning of a game player, we introduced in the ludo ontology (by extending the **GameRecord** class) some concepts to express events related to the learning of a specific player, in accordance with the statement-based¹¹ vocabulary¹² used by the TIN CAN API¹³. As a result, we are able to register the learning-related events of players and maintain compatibility with the TIN CAN API.

4 A Loaded Questions BoardGame

To show the potential and usefulness of the ludo ontology, we prototyped a Loaded Questions¹⁴ board game. Its principle is to present the player with different questions about a specific topic that was selected for him by analyzing his profile information at the beginning of the game. At the moment, we consider only the player's age.

Each question is presented to the player together with a set of possible answers, from which, only one is correct (as shown in Figure 2). If the player answers the question correctly, he accumulates the number of points indicated in the square in which he is; in case of a wrong answer no points are accumulated. At the beginning of the game and after answering each question, the player must roll a dice to obtain a number of squares to progress on the board.

The player wins when he reaches the goal square and loses when the number of attempts available to reach the goal is completed. If the user wins, he accesses the next level of the game. The higher the level, the lower the number of attempts given to the player to reach the goal.

4.1 Knowledge Representation

By using ludo, during the creation process of the game, it was possible to describe its rules and parameters. By declaring an instance of type **GamePresentation**, it was possible to define the position, value and representation properties of its components, declared of type **MediaComponent**: the dice, the boxes and the questions button. By declaring an instance of the **GameSimulation** class, it was

¹¹ <https://github.com/adlnet/xAPI-Spec/blob/master/xAPI.md#statement>

¹² <https://github.com/adlnet/xAPI-Spec/blob/master/xAPI.md#definitions>

¹³ <http://tincanapi.com>

¹⁴ [https://en.wikipedia.org/wiki/Loaded_Questions_\(game\)](https://en.wikipedia.org/wiki/Loaded_Questions_(game))

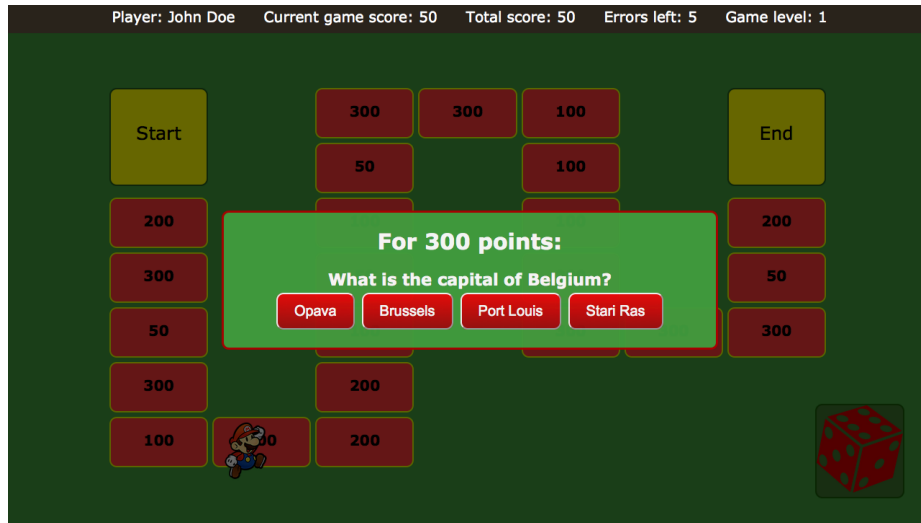


Fig. 2. A Loaded Questions board game based on ludo

possible to describe the game dimension of the game (2D), its game tempo and the front end displays that show the current number of points and the chances left when playing.

Here is an excerpt in turtle of the representation of some graphic elements of the game.

```

ludo:mainMenu rdf:type ludo:GamePresentation ;
  ludo:hasMediaComponent ludo:dice, ludo:box1, ludo:questionButton .
ludo:dice rdf:type ludo:graphic;
  ludo:hasPosition ludo:dicePosition ;
  ludo:hasURI <http://dbpedia-test-fr.inria.fr/ludo/semboardgame/img/dice.png> .
ludo:dicePosition rdf:type ludo:coordinates;
  ludo:y "-40.0"^^xsd:double ; ludo:x "-40.0"^^xsd:double .
ludo:box1 rdf:type ludo:graphic;
  ludo:hasPosition ludo:box1Position .
ludo:box1Position rdf:type owl:NamedIndividual ;
  ludo:x "30.0"^^xsd:double ; ludo:y "40.0"^^xsd:double .
ludo:questionButton rdf:type owl:NamedIndividual ;
  ludo:hasPosition ludo:questionButtonPosition .
ludo:questionButtonPosition rdf:type ludo:coordinates ;
  ludo:y "-40.0"^^xsd:double ; ludo:x "210.0"^^xsd:double .

```

The `GameRule` class was used to describe the game scoring rule of the game (the answer to the question is the correct one), and the respective outcome. The levels of the game were described with a difficulty indicator associated to a resource of type `GameScenario`.

Here is an example representation of the profile and the initial context of a player of the board game, with `ludo` the namespace prefix for the ludo ontology.

The initial game level contextual property of each player is set to “1” by default, while his current virtual activity is set to “playing”.

```
ludo:player2 a ludo:GamePlayer;
  foaf:age 15 ; foaf:depiction <player009.png>;
  foaf:family_name "Doe"; foaf:givenName "John"; foaf:nick "johny";
  ludo:currentGameLevel ludo:level1;
  ludo:currentVirtualActivity ludo:va1.
ludo:level1 rdf:type ludo:GameLevel;
  ludo:hasGameLevelNumber "1"^^xsd:nonNegativeInteger;
  ludo:hasGameLevelDescription "This is the initial level" .
ludo:va1 rdf:type ludo:VirtualActivity;
  ludo:VirtualActivityDescription "playing" .
```

Since the game relies on DBpedia as knowledge base, we used the property `void:sparqlEndpoint` to relate the game to its Endpoint URL:

```
ludo:dbpedia rdf:type ludo:GameKnowledgeBase;
  void:sparqlEndpoint <http://dbpedia.org/sparql> .
```

Considering the fact that for most of the Wikipedia pages, there is a corresponding DBpedia URI containing an RDF description of it, as a first step to the recommendation, we provide the game player with the chance to increase his knowledge on a specific topic, by presenting him the corresponding Wikipedia page of a DBpedia item, whose format is more adapted for reading (HTML).

4.2 Reasoning of the Game Engine

For the gameplay, the game engine executes SPARQL queries for two main purposes:

- to access the ludo-based RDF representation of the game. It includes the player profile and context, game parameters and settings.
- to access to the game knowledge base(s) — DBpedia in the case of our prototype game. The results of these SPARQL queries are used to formulate the game questions and possible answers on the fly.

When the game is launched, the game parameters and the player information are loaded. Then, the game engine is ready to query the domain knowledge base(s) and from the results obtained, it generates questions and answers that will be presented to the player as a game step. To achieve this, we adopted an approach based on [16]. First, a set of predefined dynamic queries from which to retrieve pairs of question-answer are created, each of them is associated with a topic and targets a specific player profile (for example a player with an age under 10 years). For instance, the following query enables to retrieve a list of countries in the world and their capitals.

```

SELECT DISTINCT ?s ?o ?p
WHERE {
  ?resource rdf:type dbo:Country.
  ?resource rdfs:label ?s.
  ?resource dbpprop:capital ?capital.
  dbo:capital rdfs:label ?p.
  ?capital rdfs:label ?o.
  FILTER ( lang(?p) = "en" )
  FILTER ( lang(?s) = "en" )
  FILTER ( lang(?o) = "en" )
} LIMIT 500

```

The results are multi sets of bindings with variables `?s` `?o` `?p` which are processed by the game engine, that uses these values in order to compose the question with the following template:

Question: What is the **capital(?p)** of **Belgium(?s)** ?

Answer: **Brussels(?o)**

Additional queries can be added to the game, resulting in more topics of questions, for example, we have also a query that gets the name of the best soccer scorers and the team to which they belong. When the game engine processes the results of a query with this template, it will also dynamically generate a set of *distractors*, that is, incorrect answers presented as possible answers to a question. Such incorrect answers are generated by first selecting all the question-(right)answer pairs. Then, to each pair, some other 3 false answers (taken randomly from other pairs) are added.

Finally, if the player selects a wrong answer, he is asked if he wants to be shown more information related to the resource in Wikipedia.

4.3 Implementation

The game prototype has been designed to be played from any web browser. The front end was created using HTML5 while the game logic has been programmed in Javascript by using the JQuery¹⁵ framework, which allows graphics effects and asynchronous calls (AJAX) to the backend developed in PHP using the ARC2¹⁶ library for working with RDF data through SPARQL queries. It is freely available online¹⁷.

5 The Semantic Educloud Project

Since 2013, the Educloud Project¹⁸ aims at developing a Serious Game to immerse college students as superheroes in a period of time (corresponding to his

¹⁵ <http://jquery.com>

¹⁶ <https://github.com/semsol/arc2>

¹⁷ <http://dbpedia-test-fr.inria.fr/ludo/semboardgame>

¹⁸ <http://www.gaya-technology.com/educloud/>

level of education) and in a city of his environment. The student discovers a virtually created real-life scenario and must solve puzzles that will confront him against the myths and beliefs of that time. While immersed in a 3D world, the student will acquire new skills and knowledge of Art History, visual arts, geography and technology. Different games will offer him the opportunity to time travel as he keeps solving the puzzles.

From a technical point of view, the goal is to create a true gaming experience, both in the style and in the gameplay, nevertheless able to maintain a rigorous educational requirement. This project relies on the Gayatech's skills in cloud data hosting, and Shiva 3D engine, and their experience in the gaming world.

In the context of this project, the Semantic Educloud project has been created in collaboration with Inria to improve the gameplay and recommendation of educational resources (proposed to Educloud learners) by using Semantic Web models and techniques to take advantage of the huge and increasing amount of Linked Data resources as well as the learner's profile and his contextual information. The ludo ontology presented in this paper represents the basis for the development of this project.

6 Conclusions and Future Work

In this paper, we have presented the ludo ontology to model Serious Games by taking into account the profile and context information of the game player. We reused existing vocabularies available in the Linked Data cloud: FOAF, to handle the game player's profile information; and VoID, to represent the datasets that the game can exploit as a knowledge base, either to represent resources as items inside it or to recommend them to the game player. Furthermore, we introduced an implementation of the simple vocabulary proposed by the TIN CAN API, to monitor the events related to the learning during the game of a player. Finally we created and integrated a vocabulary to represent the contextual information on a player inside the game.

Ludo has been used to create a web-based question-based serious board game that generates questions from the DBpedia semantic knowledge base. It enables to represent the game parameters, and players' profile as well as to monitor his learning process. Let us note that in this proposed game, we did not answer the well-known problem of data quality which occurs when automatically building questions from the LOD, since we focused on validating the ludo ontology. This should be addressed as future work.

In a near future, the ludo ontology will also be used in the context of the Semantic Educloud project, whose goal is to add semantic technologies in the Educloud project to develop a serious game benefitting from the resources available on the Linked Data cloud.

References

1. Djaouti, D., Alvarez, J., Jessel, J.P., Rampnoux, O.: Origins of serious games. In: *Serious Games and Edutainment Applications*. Springer London (2011) 25–43

2. Sawyer, B., Rejeski, D.: Serious games: Improving public policy through game-based learning and simulation. Woodrow Wilson International Center for Scholars (2002)
3. Michael, D.R., Chen, S.L.: Serious Games: Games That Educate, Train, and Inform. Muska & Lipman/Premier-Trade (2005)
4. Bizer, C., Heath, T., Berners-Lee, T.: Linked data - the story so far. *Int. J. Semantic Web Inf. Syst.* **5**(3) (2009) 1–22
5. Siorpaes, K., Hepp, M.: Ontogame: Towards overcoming the incentive bottleneck in ontology building. In: *On the Move to Meaningful Internet Systems 2007: OTM 2007 Workshops*. Volume 4806 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg (2007) 1222–1232
6. Siorpaes, K., Hepp, M.: Ontogame: Weaving the semantic web by online games. In: *The Semantic Web: Research and Applications*. Volume 5021 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg (2008) 751–766
7. Siorpaes, K., Hepp, M.: Games with a purpose for the semantic web. *IEEE Intelligent Systems* **23**(3) (2008) 50–60
8. Thaler, S., Siorpaes, K., Mear, D., Simperl, E., Goodman, C.: Seafish: A game for collaborative and visual image annotation and interlinking. In: *The Semantic Web: Research and Applications*. Volume 6644 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg (2011) 466–470
9. Waitelonis, J., Ludwig, N., Knuth, M., Sack, H.: Whoknows? evaluating linked data heuristics with a quiz that cleans up dbpedia. *Interactive Technology and Smart Education* **8**(4) (2011) 236–248
10. Wolf, L., Knuth, M., Osterhoff, J., Sack, H.: Risq! renowned individuals semantic quiz: A jeopardy like quiz game for ranking facts. In: *Proceedings of the 7th International Conference on Semantic Systems. I-Semantics '11, New York, NY, USA, ACM* (2011) 71–78
11. Longstreet, C., Cooper, K.: Developing a meta-model for serious games in higher education. In: *Advanced Learning Technologies (ICALT), 2012 IEEE 12th International Conference on*. (2012) 684–685
12. Tang, S., Hanneghan, M.: Game content model: An ontology for documenting serious game design. In: *Developments in E-systems Engineering (DeSE), 2011*. (2011) 431–436
13. Friberger, M.G., Togelius, J.: Generating game content from open data. In: *International Conference on the Foundations of Digital Games, FDG '12, Raleigh, NC, USA, 2012*. (2012) 290–291
14. Warren, R., Champion, E.: Linked open data driven game generation. In: *The Semantic Web – ISWC 2014*. Volume 8797 of *Lecture Notes in Computer Science*. Springer International Publishing (2014) 358–373
15. Costabello, L.: DC proposal: Prisma, towards mobile adaptive presentation of the web of data. In: *The Semantic Web - ISWC 2011 - 10th International Semantic Web Conference, Bonn, Germany, 2011, Proceedings, Part II*. (2011) 269–276
16. Foulonneau, M.: Generating educational assessment items from linked open data: The case of dbpedia. In: *The Semantic Web: ESWC 2011 Workshops*. Volume 7117 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg (2012) 16–27