Pascal Molli
John Breslin
Maria-Esther Vidal (Eds.)

SWCS'14

Third International Workshop on Semantic Web Collaborative Spaces, 2014

Workshop co-located with the 13th International Semantic Web Conference (ISWC 2014)

Riva del Garda, Trentino, Italy, October 19-23th, 2014

Proceedings

Editors' addresses:
LINA Lab, University of Nantes, France
pascal.molli@univ-nantes.fr;

National University of Ireland Galway, Ireland
john.breslin@nuigalway.ie;

Universidad Simón Bolívar
Department of Computer Science
Valle de Sartenejas
Caracas 1086, Venezuela
mvidal@ldc.usb.ve

Preface

This volume contains the papers accepted at the Third International Workshop on Semantic Web Collaborative Spaces, 2014, held on October 19th, 2014. All the papers included in this volume went through a peer-review process. Papers were evaluated in terms of *Technical Depth*, *Significance*, *Novelty*, *Relevance and Completeness of the References*, *Approach Evaluation*, and *Quality of the Presentation*. We accepted six out of seven submissions; five papers were accepted as long papers, and one paper was accepted as short paper. Our sincere thanks to the Program Committee members and external reviewers for their valuable input and for accepting to contribute to the review process.

Collaboration between data producers and consumers is a key challenge for facilitating the evolution of the Linking Open Data (LOD) cloud into a participative and updatable LOD cloud. Semantic Web Collaborative Spaces support the collaboration among Open Data producers and consumers to publish and maintain Linked Data, as well as, to improve quality. These collaborative spaces include social semantic frameworks such as crowdsourcing tools, semantic wikis, semantic social networks, semantic microblogs. Collaborative spaces have been developed for different domains, e.g., Health care, Life Sciences, and e-Government.

After two successful events, first in Lyon, France, joined to the 21st International World Wide Web Conference (WWW 2012), then in Montpellier, France, collocated with the Extended Semantic Web Conference (ESWC 2013), the Third International Workshop on Semantic Web Collaborative Spaces is collocated with the International Semantic Web Conference (ISWC 2014), at Trentino, Italy.

The Third International Workshop on Semantic Web Collaborative Spaces aimed at bringing together researchers from the database, artificial intelligence and semantic web areas, to discuss research issues and experiences in developing and deploying concepts, techniques and applications that address various issues related to collaborative spaces. This Third edition focused on collaborative data management, models to represent collaborative knowledge and reasoning, tools to interact with SWCS, and applications.

We set up an exciting program which included three sessions: Modeling Collaborative Communities, Applications of Semantic Collaboration, and Semantic Media Wiki Communities. We are grateful to ISWC organizers for their support in making this meeting successful.

October 2014 Pascal Molli, John Breslin, and Maria-Esther Vidal

Workshop Chairs and Organizing Committee

Prof. Dr. Pascal Moll, University of Nantes, France
Dr. John Breslin, National University of Ireland Galway, Ireland
Prof. Dr. Maria-Esther Vidal, Universidad Simón Bolívar, Venezuela

Program Committee

Maribel Acosta, AIFB, Karlsruhe Institute of Technology, Germany. (DE)
Uldis Bojars, University of Latvia, Latvia. (LV)
Anne Boyer, Lorraine University, France. (FR)
John Breslin, National University of Ireland Galway, Ireland. (IE)
Tobias Bürger, Capgemini SD&M, Germany. (DE)
Michel Buffa, University of Nice, France. (FR)
Amelie Cordier, LIRIS, Lyon University, France. (FR)
Alicia Diaz, La Plata University, Argentina. (AR)
Fabien Gandon, INRIA, Sophia Antipolis, Wimmics, France. (FR)
Magnus Knuth, Hasso Plattner Institute, University of Potsdam, Germany. (DE)
Markus Krötzsch, University of Oxford, United Kingdom. (UK)
Christoph Lange, University of Bonn, Fraunhofer IAIS, Germany. (DE)
Pascal Molli, University of Nantes, France. (FR)
Claudia Müller-Birn, Freie Universität, Berlin. Germany. (DE)
Grzegorz J. Nalepa, AGH University of Science and Technology, Poland. (PL)
Amedeo Napoli, CNRS, LORIA, France. (FR)
Harald Sack, Hasso Plattner Institute, University of Potsdam, Germany. (DE)
Hala Skaf-Molli, University of Nantes, France. (FR)
Sebastian Tramp, University of Leipzig, Germany. (DE)
Josef Urban, Radboud Universiteit, Nijmegen, Netherlands. (NL)
Maria-Esther Vidal, Universidad Simon Bolivar, Venezuela. (VE)

Sponsoring Institutions

ANR Kolflow Project (ANR-10-CORD-0021), University of Nantes
DID-USB http://www.did.usb.ve

Contents

Okkam Synapsis: a community-driven hub for sharing and reusing mappings across vocabularies

Stefano Bortoli[1], Paolo Bouquet[1 2], and Barbara Bazzanella[2]

[1] Okkam SRL
Via Segantini 23, I-38121 Trento, Italy
[2] University of Trento - DISI
Via Sommarive, 14 I-38123 Povo di Trento, Italy
bortoli@okkam.it, bouquet@disi.unitn.it, barbara.bazzanella@unitn.it

Abstract. In the past 10-15 years, a large amount of resources have been devoted to develop highly sophisticated and effective tools for automated and semi-automated schema-vocabulary-ontology matching and alignment. However, very little effort has been made to consolidate the outputs, in particular to share the resulting mappings with the community of researchers and practitioners, support a community-driven revision/evaluation of mappings and make them reusable. Yet, mappings are an extremely valuable asset, as they provide an *integration map* for the web of data and the "glue" for the Global Giant Graph envisaged by Tim Berners-Lee. Aiming at kicking-off a positive endeavor, we have developed *Synapsis*, a platform to support a community-driven lifecycle of contextual mappings across ontologies, vocabularies and schemas. Okkam Synapsis offers utilities to load, create, maintain, comment, subscribe, and define levels of agreement over user defined contextual mappings available also through REST services.

Acknowledgement. This work is partially supported by TAG CLOUD (Technologies lead to Adaptability and lifelong enGagement with culture throughout the CLOUD) FP7 EU Funded project, Grant agreement nr: 600924.

1 Introduction and Motivation

In the promising vision of the Semantic Web proposed by Tim Berners-Lee [2], the collaborative and distributed creation of semantically annotated documents would enable software agents to perform time-consuming activities on behalf of human users (see [1]). The community that gathered to corroborate and develop this ambitious vision achieved many relevant results with the definition of important standards such as OWL[19, 18, 14], RDF[15], and the important Linked Data publication principles [3, 4]. The combination of these principles with the more recent open data initiative across many countries is generating a considerable

amount of publicly available RDF and OWL data. In recent years, enterprises are attracted by the promise of using such big and rich data to develop new products and services for their customers (e.g. [7]). However, exploiting and mining data rises many challenges including the problems of entity matching, ontology matching, and making the data accessible and usable by non-expert users. In the past ten years many efforts were spent in the definition of sophisticated tools for automated ontology matching. These often provided very effective solutions in narrow domains, but a generic automatic reliable solution to the problem is still an open research problem [20]. Furthermore, in [26] it was recently discussed how often even experts have problems in finding agreement on defined ontology mappings. We argue that this is due to two main problems: 1) the intrinsic complexity and heterogeneity of existing ontologies, and 2) the inconsistency and fuzziness in usage of such ontologies due to contextually interpretable semantics. Namely, concepts and relations expressed in natural language are interpreted outside the original context of definition, and therefore prone to contextual interpretation. In fact, besides the effort of researchers in formal ontology [13, 11, 10] the process of ontology definition is driven by specific domain requirements and often ontology engineering practices are neglected [16].

Under these premises, we decided to take one of the ten challenges of ontology matching described in [23] and confirmed in [20], and propose a novel platform to support a collaborative ontology mappings definition and reuse [27]. The idea to take this challenge is rooted in the pragmatic need of resolving the problem of semantic heterogeneity affecting a knowledge-based solution of the entity matching in the context of the Semantic Web [5]. In particular, in this work we argue that collecting and maintaining ontology mappings as contextual bridge rules [6] to harmonize the semantic of entities' attributes can provide great benefits by enabling the application of knowledge-based solution to an entity matching problem [5]. Therefore, in our attempt to solve the entity matching problem in the linked data, we produced several thousands of mappings from existing ontologies, schemas and vocabularies towards a target ontology we named Identification Ontology[3] ([5] Chap. 5). Often these mappings were produced without considering the original, or intended, semantic of the properties, but rather relying on its actual function looking directly into the data. This approach, besides being practical and concrete, interprets the ontology mappings as contextual analogies as suggested in [21]. Namely, when producing mappings, rather than considering the similarity among original intended functional purpose of the properties (homology), we consider also its real function (analogy) so that the mapping relation holds primarily on the instances level. On the one hand, we are aware that this approach will create mappings that might not be absolutely coherent and correct across several contexts, but as long as they serve the purpose we can live with this limitation. On the other hand, we want to use a first core set of mappings to kick-off a positive endeavor for the definition of a platform to support a community-driven lifecycle of contextual mappings

[3] `http://models.okkam.org/identification_ontology.owl`

between ontologies, vocabularies and schemas that could serve the definition of new applications exploiting open linked data.

In this work we describe Okkam Synapsis, a web application conceived to support the linked data community in creating, sharing and reusing contextual ontology mappings to support the creation of novel services based on the linked data consumption. Okkam Synapsis offers utilities to load, create, maintain, comment, subscribe, and manage levels of agreement over user defined contextual mappings. Most importantly, endorsing the recommendations described in [26], we support different fine-grained typing models for the definition of the mappings (e.g. OWL and SKOS) and compute level of agreement according to different metrics to support filtering based on them. The mappings produced will be available also through REST services, providing several levels of selection to support diverse and unforeseen application scenarios. The purpose of the application is to enable the users of Okkam Synapsis to collaborate in the definition of mappings, commenting, rating, and subscribing them. Furthermore, we want to allow users to explicitly define the context of use of the defined mappings, so that others can take informed decision about reusing.

The underlying assumptions are:

- real linked data is in general too messy to rely on a unique set of mappings in different contexts of use
- linked data may change in time, therefore contextual mappings must be subject to specific lifecycle
- the number of existing vocabularies is growing, but reuse practices make the manual mapping process feasible (see Linked Open Vocabulary[4])
- perfect agreement about defined mappings is unlikely to happen [26], better let users to select what they need

The reminder of the paper is organized as follows: in Section 2 we overview the related works dealing with crowd-sourcing of ontology mappings, and other community-driven approaches; in Section 3 we describe in detail the platform, discussing functions and services. In Section 4 an overview of the architecture of the application is provided and finally in Section 7 we describe future work and outline some concluding remarks.

2 Related Work

According to the most recent survey we are aware of [20], there are not many tools supporting collaborative creation of ontology mappings. In [27] is described a system for community-driven ontology matching, embedding provenance, freshness and other metadata suitable for the selection of the mappings. Besides the a low-resolution screenshot presented in the paper, the system does not seem to be available anymore. In [17] Noy et al. describe a system for the collection of biomedial ontologies supporting the definition of mappings among them. Having

[4] `http://lov.okfn.org/dataset/lov/`

collected more than 30.000 mappings, the authors propose a systems for filtering and searching mappings. Furthermore, they argue about the concept of *mappings as bridges*, and outline the need of specifying the type of relations (e.g. equivalence). Currently the system is up and running, serving more than 370 biomedial ontologies and several million of concepts. In [22] is described CrowdMap, a solution for ontology matching based on crowd-sourcing. The ontology matching task are decomposed in micro-tasks and submitted to workers of crowd-sourcing platforms such as CrowdFlower and MTurk for manual evaluation. The results obtained were compared with the one of automatic tools showing the feasibility of the process. In [8] the authors discuss about the need of managing and reducing uncertainty related to crowdsourcing of ontology matching tasks, proposing different ways to create micro-tasks suitable to increase possible agreements. In [7] the authors describe Helix as a tool for creating ontology mapping as a pay-as-you-go task while consuming linked data. In [12] the Correndo and Alani describe OntoMediate, a project of the University of Southampton aiming at supporting, among other functions, the creation and sharing of ontology mappings. Unfortunately, the project is over and to the best of our knowledge there is no service available. Another trend in managing collective ontology matching is through gamification. In [16] and [24] are described Guess What?! and SpotTheLink proposing the solution of ontology matching tasks in form of games to give incentives and foster engagement to ease the cognitive effort of users and stimulate the creation mappings and links in the linked data cloud. Noticeably, to the best of our knowledge these systems are not currently available. In this context we do not consider papers presenting automatic solutions to the ontology matching for which we refer to the aforementioned survey [20].

In light of the analysis presented, to the best of our knowledge, the only system available providing the services comparable with the one of Okkam Synapsis is BioPortal [17]. However, given the vertical purpose of BioPortal and the limited collaborative features, we can safely affirm that there is room for a solution such as the one proposed in this paper.

3 User Interface and Features

The current version of Synapsis distinguishes between two kinds of users: administrators and end users. Administrators are users that have unrestricted access to all the user-level functions, including uploading a source ontology, creating new mappings for concepts and properties, deleting existing mappings, setting/changing the status of defined mappings, evaluating existing mappings and reusing/exporting mappings. End users have only access to social functions to express their level of agreement on previously created mappings and reusing them. They can endorse and comment existing mappings, follow mappings they are interested in, rate mappings and export mappings.

Figure 1 shows a snapshot of the User Interface of Okkam Synapsis which presents three main areas: the (target) ontology on the left, the mappings in the central part and the mapping filters on the right.

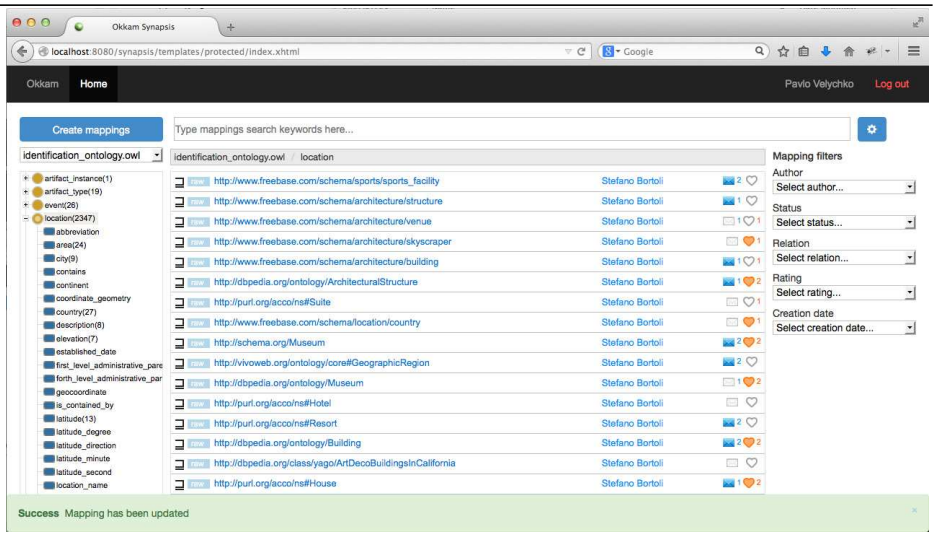


Fig. 1. Synapsis User Interface

After logging in, the user can select one of the ontologies/vocabularies currently present in the platform from the drop-down menu on the top-left corner of the page or import a new ontology selecting the Import function from the Function button. Following [17], we call the selected/uploaded ontology the Target Ontology[5], which is the ontology whose concepts/properties the user wants to map to target concepts/properties. After having selected it, the target ontology is loaded, processed and represented as an indented tree on the left side of the interface. The choice of using an indented tree is based on the study described in [9], where users evaluated this representation model as easier to use and more understandable than alternative models such as the graphs. With the primary objective of enabling users in defining mappings, we decided to flatten the ontology to a list of concepts and present the properties attached to them. In the current version, we rely on a simple RDF processor implemented relying on Apache Jena API[6]. The selection of a node of the source ontology triggers the loading of all the mappings defined for that concept or property in the central part of the window. Each mapping is composed by the following attributes:

- Resource URI: the URI of the resource mapped towards the element of the target ontology.
- Relation Type: the type of relation between the Resource URI and the Target URI. The user can select among an enumeration of relations types including

[5] According to this naming convention, a mapping can be seen as a relationship between two concepts/properties in different ontologies. Each mapping has a source concept/property, a target concept/property, and a mapping relationship.

[6] `https://jena.apache.org/`

OWL meta-relations such as *owl:EquivalentProperty*, *owl:EquivalentClass*, *owl:SubClass*, *owl:SubProperty* and SKOS meta-relations *skos:exact*, *skos:close*, *skos:broader*, *skos:narrower*. In this context, we neglect *skos:related* and *skos:unrelated* because we believe these types of relations are not interesting in our context. In order to help the user in choosing the right type of relation we refer to the guidelines proposed in [26] and still available at [25] as appendix A2.
- Status: a label among *Raw*, *Edited*, *Closed*, *Accepted*, declaring the status of a mapping. These labels are assigned by administrators of Okkam Synapsis keeping into consideration time and opinions expressed by the members of the community.
- Author: the author of the mapping.
- Description: A description of the resource mapped possibly coming from official documentation.
- License: a statement declaring the licensing model under which the mapping is made available to the community.
- Agreement Metrics: every user is enabled in stating whether she/he agrees or not with the proposed mapping. The level of agreement may be estimated using different metrics as suggested in [26].
- Number of Watchers: any mapping can be watched by a member of the community. Watching a mapping allows users to be notified about activities concerning the mapping.
- Number of Likes: any mapping can be *liked* by a member of the community. A like essentially implies an agreement and a subscription to possible events related to the mapping.
- Comments: members of the community are enabled in commenting and discussing about a mapping. We foresee cases where people may ask for clarifications and argue about the validity of the mapping.
- Contextual Tags: any mapping is annotated with a set of tags which identify fuzzy contexts of application of the mappings. These tags can be used to search and filter mappings.

In figure 1, one can see a graphical representation of all the mappings about the Location concept of the Identification Ontology. The first two graphical elements of the interface describe the relation and the status. The, after the URI of the mapping, one can see the author of the mapping, the whether the mapping was watched and by how many users. Finally, we show the number of people care about that specific mapping. Then, on the right side of the interface, a user can filter mappings according to these main dimensions, and typing on the top input field, can filter mappings relying on the namespaces or the local part of the mappings URIs.

Clicking on each mapping, the user can visualize all the details about the mapping in a specific detail page (as shown in Figure 2). This page allows to add comments, rate, subscribe and add possible contextual tags in a collaborative manner. Ratings are made on a 6-item scale including the following options: approved (i.e. the source and target concepts both mean the same thing), broader

(the target concept should be a broader term than the source concept), narrower (i.e. the target concept should be a more specific term than the source concept), related (i.e. the two concepts are not an exact match but they are closely related), not sure (i.e. there is a relationship between the two concepts but none of the above relations are appropriate or the term is used in a confusing or contradictory fashion), rejected (i.e. the two concepts are definitely not the same, nor do they have any other direct relationship with each other as listed above). The mapping detail page essentially aims to provide tools for the collaborative interaction for each single defined mapping. If a user subscribes a mapping, any notification will include a link to the specific mapping detail page.

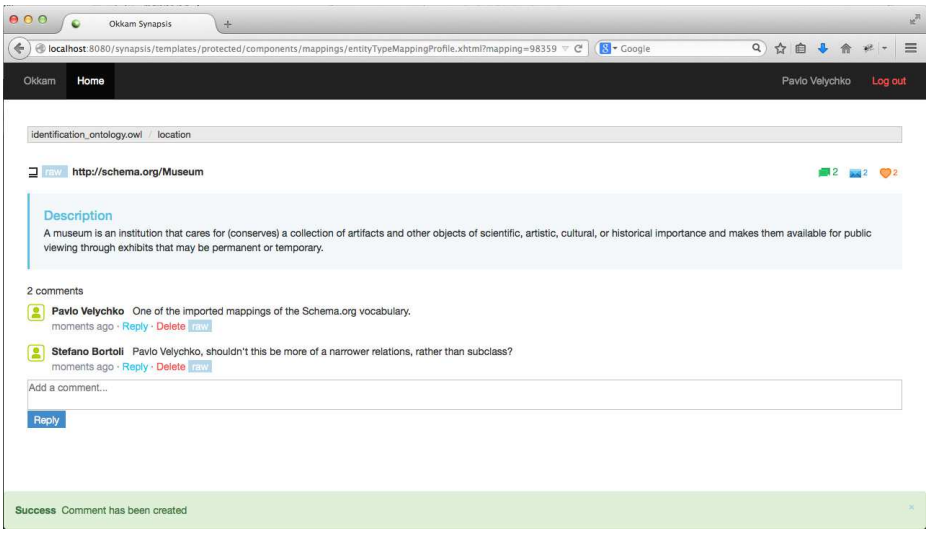


Fig. 2. Mapping page

Once selected the target ontology, the user is enabled in filtering mappings according to different features. On the right part of the page, the filter features are displayed, and the user is enabled in selecting them. Each selection triggers an action on the list of mappings, removing the filtered ones. Mappings can be filtered by author name, status, relation type, rating and creation date. It is also possible to select all the mappings that have comments.

4 Architecture and Data Model

The application is designed according to the traditional MVC design pattern, relying on J2EE JSF framework[7] for the Web interaction part. The mappings

[7] `http://docs.oracle.com/javaee/5/tutorial/doc/bnaph.html`

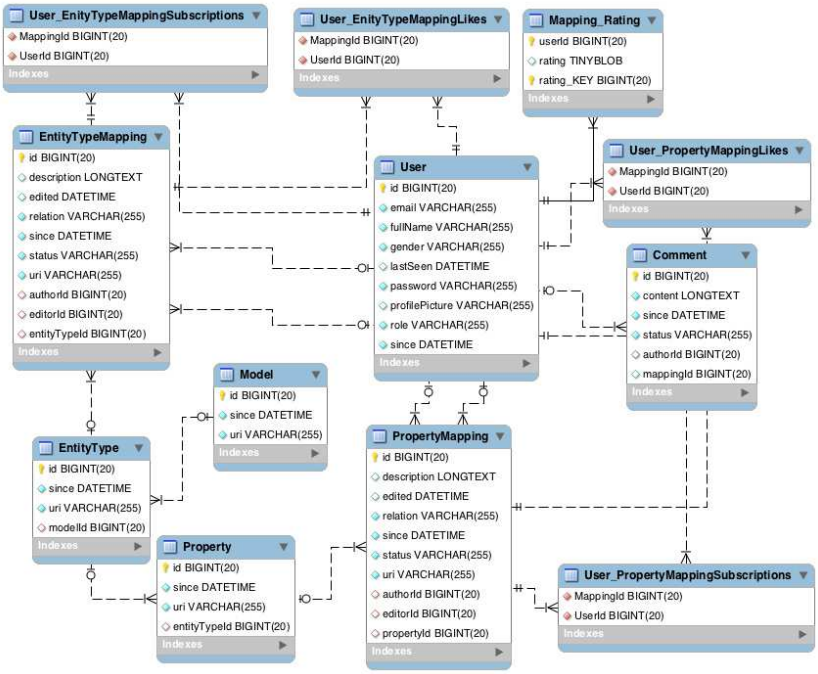


Fig. 4. Synapsis Data Model

6 Kick-off Mappings Dataset

Currently, Synapsis stores 22 mapping for equivalent class, and 205 mapping for subclasses of the entity type Person; 22 mappings for equivalent classes, and 2322 mappings for sub classes of the type Location; and finally we defined 20 mappings for equivalent classes and 2468 mappings for subclasses of the type Organization. These mappings were generated as contextual bridge rules to support semantic harmonization tasks in the knowledge-based solution described in [5]. In particular, the reader can find details about the process leading to the creation of such mappings from existing vocabularies towards the Identification Ontology[11] in Chapter 7 of [5]. We believe that this first core set of mappings can help to kick of a positive endeavor in the adoption of the Okkam Synapsis as a platform to create, share and manage mappings among vocabularies.

7 Conclusion and Future Work

In this paper we have presented a platform called Synapsis which provides a gateway to collaboratively-defined ontology mappings. Looking for a pragmatic

[11] `http://models.okkam.org/identification_ontology.owl` ([5] Chap. 5)

solution to the real world heterogeneity, complexity and inconsistencies, we decided enable users to define mappings as contextual bridge rules, and enable peers to comment and discuss about them. We believe that rating and estimation of level of agreement about mappings would allow to filter commonly shared mappings, and at the same marginalize odd ones. A beta version of the application is available at `http://api.okkam.org/synapsis`, and can be preliminarily tested and evaluated. In the next future, we plan to extend the support for the definition of mappings around applications, to support Linked Data application developer to select the set of mappings of interest and have them available for the application through the defined rest services.

References

1. G Antoniou and F. van Harmelen. *A Semantic Web Primer*. MIT Press, 2004.
2. T. Berners-Lee, J. A. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, May, 2001. http://www.sciam.com/2001/0501issue/0501berners-lee.html.
3. Tim Berners-Lee. Design Issues – Linked Data. Published online, May 2007. `http://www.w3.org/DesignIssues/LinkedData.html`.
4. C. Bizer, R. Cyganiak, and T. Heath. How to publish linked data on the web. online tutorial, July 2007.
5. Stefano Bortoli. *Knowledge Based Open Entity Matching*. PhD thesis, International Doctoral School in ICT of the University of Trento (Italy), 2013.
6. Paolo Bouquet, Fausto Giunchiglia, Frank Harmelen, Luciano Serafini, and Heiner Stuckenschmidt. C-owl: Contextualizing ontologies. In Dieter Fensel, Katia Sycara, and John Mylopoulos, editors, *The Semantic Web - ISWC 2003*, volume 2870 of *Lecture Notes in Computer Science*, pages 164–179. Springer Berlin Heidelberg, 2003.
7. Jason B. Ellis, Oktie Hassanzadeh, Kavitha Srinivas, and Michael J. Ward. Collective ontology alignment. In *OM*, pages 219–220, 2013.
8. Jrme Euzenat. Uncertainty in crowdsourcing ontology matching. In *OM*, pages 221–222, 2013.
9. Bo Fu, Natalya F. Noy, and Margaret-Anne Storey. Indented tree or graph? a usability study of ontology visualization techniques in the context of class mapping evaluation. In *The Semantic Web ISWC 2013*, pages 117–134. Springer Berlin Heidelberg, 2013.
10. A. Gangemi and V. Presutti. *Ontology Design Patterns*, pages 221–243. Springer Berlin Heidelberg, 2009.
11. Aldo Gangemi, Nicola Guarino, Claudio Masolo, Alessandro Oltramari, and Luc Schneider. Sweetening ontologies with dolce. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web*, EKAW '02, pages 166–181, London, UK, UK, 2002. Springer-Verlag.
12. Harith Alani Gianluca Correndo. Collaborative support for community data sharing. In *Proceedings of The 2nd Workshop on Collective Intelligence in Semantic Web and Social Networks*, 2008.
13. Nicola Guarino and Chris Welty. An overview of ontoclean. In Steffen Staab and Rudi Studer, editors, *The Handbook on Ontologies*, pages 151–172. Springer-Verlag, 2004.

14. Pascal Hitzler, Markus Kroetzsch, Bijan Parsia, Peter F. Patel-Schneider, and Sebastian Rudolph. *OWL 2 Web Ontology Language Primer (Second Edition)*. W3C, December 2012.
15. Frank Manola, Eric Miller, and Brian McBride. *RDF 1.1 Primer*. W3C, w3c working group note edition, June 2014.
16. Thomas Markotschi and Johanna Völker. Guess What?! human intelligence for mining linked data. In *Proceedings of the Workshop on Knowledge Injection into and Extraction from Linked Data (KIELD) at the International Conference on Knowledge Engineering and Knowledge Management (EKAW)*, 2010.
17. NatalyaF. Noy, Nicholas Griffith, and MarkA. Musen. Collecting community-based mappings in an ontology repository. In Amit Sheth, Steffen Staab, Mike Dean, Massimo Paolucci, Diana Maynard, Timothy Finin, and Krishnaprasad Thirunarayan, editors, *The Semantic Web - ISWC 2008*, volume 5318 of *Lecture Notes in Computer Science*, pages 371–386. Springer Berlin Heidelberg, 2008.
18. W3C OWL Working Group. OWL 2 Web Ontology Language: Document Overview, 27 October 2009. Available at `http://www.w3.org/TR/owl2-overview/`.
19. P.F. Patel-Schneider, P. Hayes, and I. Horrocks. Web Ontology Language (OWL) Abstract Syntax and Semantics. Technical report, W3C, February 2003. `http://www.w3.org/TR/owl-semantics/`.
20. Shvaiko Pavel and Jerome Euzenat. Ontology matching: State of the art and future challenges. *IEEE Trans. on Knowl. and Data Eng.*, 25(1):158–176, January 2013.
21. Elie Raad and Joerg Evermann. Is ontology alignment like analogy? – knowledge integration with lisa. In *Proceedings of Symposium On Applied Computing (SAC), Korea, Republic Of (2014)*, 2014.
22. Cristina Sarasua, Elena Simperl, and Natalya F. Noy. Crowdmap: Crowdsourcing ontology alignment with microtasks. In *Proceedings of the 11th International Conference on The Semantic Web - Volume Part I*, ISWC'12, pages 525–541, Berlin, Heidelberg, 2012. Springer-Verlag.
23. Pavel Shvaiko and Jérôme Euzenat. Ten challenges for ontology matching. In *Proceedings of the OTM 2008 Confederated International Conferences, CoopIS, DOA, GADA, IS, and ODBASE 2008. Part II on On the Move to Meaningful Internet Systems*, OTM '08, pages 1164–1182, Berlin, Heidelberg, 2008. Springer-Verlag.
24. Stefan Thaler, Elena Simperl, and Katharina Siorpaes. Spotthelink: Playful alignment of ontologies. In *Proceedings of the 2011 ACM Symposium on Applied Computing*, SAC '11, pages 1711–1712, New York, NY, USA, 2011. ACM.
25. Anna Tordai. *On Combining Alignment Techniques*. PhD thesis, Vreije Universiteit Amsterdam, 2012-12-03.
26. Anna Tordai, Jacco van Ossenbruggen, Guus Schreiber, and Bob Wielinga. Let's agree to disagree: On the evaluation of vocabulary alignment. In *Proceedings of the Sixth International Conference on Knowledge Capture*, K-CAP '11, pages 65–72, New York, NY, USA, 2011. ACM.
27. Anna V. Zhdanova and Pavel Shvaiko. Community-driven ontology matching. In *Proceedings of the 3rd European Conference on The Semantic Web: Research and Applications*, ESWC'06, pages 34–49, Berlin, Heidelberg, 2006. Springer-Verlag.

Collaborative Semantic Tables

Anna Goy, Diego Magro, Giovanna Petrone, Marino Segnan

Dipartimento di Informatica, Università di Torino
{goy,magro,giovanna,marino}@di.unito.it

Abstract. The scenario defined by current Web architectures and paradigms (such as Cloud Computing), poses challenges and opportunities to users. On the one hand, they have to manage huge amounts of digital resources handled by different applications in spite of their possibly related content; on the other hand, they are enabled to share knowledge and participate to content creation. The interaction of these two aspects provided a great impulse to collaborative resource management: in this paper we present T++, an environment that exploits semantic knowledge about digital resources in order to face these challenges, by providing an integrated and smart management of heterogeneous *information objects*.

Keywords: Collaborative Workspaces · Ontology-based Content Management · Personal Information Management · Linked Data · Semantic Technologies.

1 Introduction

The current ICT scenario, and in particular Web architectures and paradigms, are posing new challenges to Personal Information Management [4]. Many aspects of human-computer interaction have been influenced; in this paper, we focus on the most relevant for our approach: (a) Users have to deal with a huge number of heterogeneous resources, stored in different places, encoded in different formats, handled by different applications as belonging to different types (images, emails, bookmarks, documents, ...), despite their possibly related content. (b) Web 2.0 and, more recently, Cloud Computing, and in particular the Software-as-a-Service paradigm, have enhanced the possibility of user participation in content creation on the Web, as well as the possibility of resources and knowledge sharing. The interaction of these two aspects provided a great impulse to user collaboration in managing shared resources.

In this paper we present *Semantic Table Plus Plus* (Sem T++), an environment aimed at supporting users in collaborative resource management. In particular, in Sem T++, two types of semantic knowledge are modeled: (1) knowledge about *information objects*, i.e., information resources as such, and (2) knowledge about their *content*. The goal of this paper is to show how collaborative annotations, based on the formal semantic representation of these two types of knowledge, support users in the organization, retrieval and usage of shared digital resources.

The rest of the paper is organized as follows: in Section 2 we discuss the motivations of our approach and the main related work; in Section 3, we briefly present T++, and in Section 4 we describe its semantic enhancement, by presenting its architecture and the semantic model underlying it, and showing how the system supports users in collaboratively handling semantic descriptions of digital resources. Section 5 concludes the paper by discussing open issues and future developments.

2 Related Work

A survey and a discussion of existing Web-based applications supporting collaboration, including groupware and project management tools or suites, can be found in [10], a previous paper introducing T++. As far as the approach presented in this paper is concerned, one of the most relevant research areas is represented by Kaptelinin and Czerwinski's book *Beyond the Desktop Metaphor* [12], which contains an interesting presentation of the problems of the so-called *desktop metaphor*, and of the approaches trying to replace it. Within this framework, one of the most interesting models discussed in the mentioned book is Haystack [13], a flexible and personalized system enabling users to define and manage workspaces referred to specific tasks. Another interesting family of approaches are those grounded into Activity-Based Computing (e.g., [3], [19]), where the main concept around which the interaction is built is *user activity*. Also [9] and [16] propose a system supporting lightweight informal interactions, as well as larger and more structured collaborative projects, by relying on activity-based workspaces, handling collections of heterogeneous resources.

Strategies exploited to organize resources have been studied within the field of Personal Information Management; in particular, multi-facets classification of resources has been taken into account: resources can be tagged with meta-data representing different aspects (*facets*), leading to the creation of *folksonomies*, bottom-up classification models collaboratively and incrementally build by users [6]. Interesting improvements of such tagging systems have been designed by endowing them with semantic capabilities (e.g., [1]), in particular in the perspective of knowledge workers [14].

Another important research thread, aiming at coupling desktop-based user interfaces and Semantic Web, is represented by the so-called *Semantic Desktop* [17]. In particular, the NEPOMUK project (nepomuk.semanticdesktop.org) defined an open-source framework, based on a set of ontologies, for implementing semantic desktops, focusing on the integration of existing applications, in order to support collaboration among knowledge workers. Finally, [7] presents an interesting model connecting the Semantic Desktop to the Web of Data.

3 The Starting Point: Table Plus Plus

The *Table Plus Plus* (T++) project, described in [10] and [11], proposes an interaction model supporting users in collaboratively handling digital resources, based on the

metaphor of *tables*, populated by *objects*. T++ is characterized by the following main features.

Tables as thematic contexts. In T++, users can define shared workspaces devoted to the management of different activities. Such workspaces are called *tables* and support users in the separated, coherent and structured management of their activities. Users can define new tables, at the preferred granularity level; for instance, a table can be used to manage a work project, to handle children care, to plan a journey.

Workspace awareness. Workspace awareness is supported by three mechanisms: (a) On each table, a presence panel shows the list of table participants, highlighting who is currently sitting at the table; moreover, when a user is sitting at a table, she is (by default) "invisible" at other tables (*selective presence*). (b) Standard awareness techniques, such as icon highlighting, are used to notify users about table events (e.g., an object has been modified). (c) Notification messages, coming from outside T++ or from other tables, are filtered on the basis of the topic context represented by the active table (see [2] for a more detailed discussion of notification filtering).

Collaboration. An important aspect of T++ tables is that they are collaborative in nature, since they represent a shared view on resources and people; "tables represent *common places* where users can, synchronously or asynchronously, share information, actively work together on a document, a to-do list, a set of bookmarks, and so on" [10, p. 32]. The most peculiar aspect of T++ tables is the collaborative management of table resources: table participants, in fact, can (a) modify objects, delete them, or add new ones; (b) invite people to "sit at the table" (i.e., to become a table participant); (c) define meta-data, such as comments and annotations (see below).

Heterogeneous objects management and workspace–level annotations. Objects lying on tables can be resources of any type (documents, images, videos, to-do items, bookmarks, email conversations, and so on), but T++ provides an *abstract view* over such resources by handling them in a homogeneous way. Table objects, in fact, are considered as *content items* (identified by a URI) and can be uniformly annotated (by visibility labels, comments, and tags).

Within the T++ project, we developed a proof-of-concept prototype, consisting in a cloud application (a Java Web App deployed on the Google App Engine) accessible through a Web browser. The current version exploits Dropbox and Google Drive API to store files corresponding to table objects and Google Mail to handle email conversations. We are investigating the availability of open API provided by other common file sharing and online editing tools in order to improve interoperability and to enable users to configure the preferred tools to be exploited for object sharing and editing.

In [11] we also reported the results of a user evaluation of T++ in which we asked users to perform a sequence of pre-defined collaborative tasks (communication, resource sharing, and shared resources retrieval) using standard collaboration tools and using T++. The results showed that, with T++, performing the required tasks is faster and user satisfaction is higher.

4 The Semantic Enhancement of T++

4.1 Architecture

On the basis of T++, we designed an enhanced version, *Sem T++*, in which semantic knowledge plays a major role in supporting resource management on tables. Fig. 1 shows the relevant components of Sem T++ architecture:

- *Semantic Knowledge Manager*: it manages the semantic descriptions of table objects (stored in the Semantic KB) and invokes the Reasoner, when required. Moreover, it handles the connection with GeoNames (www.geonames.org), as described in Section 4.2. In the current prototype, it uses the OWL API library (owlapi.sourceforge.net) to interact with the ontologies and the Semantic KB, and the GeoNames Search Web Service (www.geonames.org/export) to query GeoNames. It interacts with the following knowledge bases (written in OWL: www.w3.org/TR/owl-features):
  - *Table Ontology*: it represents the (static) system semantic knowledge concerning *information objects*.
  - *Geographic Ontology*: it represents the (static) system semantic knowledge concerning geographic entities and features.
  - *Semantic KB*: it contains all the facts about the individuals involved in the semantic representation of table objects.
- *Reasoner*: it provides the system with "new" object features which can be exploited to support the user in table object management. The current proof-of-concept prototype uses Fact++ (owl.cs.manchester.ac.uk/tools/fact).
- *Object Manager*: it manages the "used objects" (i.e., objects on a table or included in objects on a table) and the references to elements used in the interaction with the user (e.g., available object types and object properties, corresponding to Table Ontology classes and relations). The Object Manager plays a mediation role between the Table Manager (and thus, indirectly, the UI) and the components which represents the system "intelligence", i.e. the Semantic Knowledge Manager and the Smart Object Analyzer.
- *Smart Object Analyzer*: it is a service that provides the Object Manager with the analysis of table objects, in order to discover information about them; for example, it looks for parts included in the analyzed object (e.g., images, links, etc.). In the current prototype, it exploits a Python Parser Service, able to analyze HTML documents[1].

[1] In the current prototype we focused on HTML documents since they are very common, quite easy to parse, and their semantic characterization introduces interesting aspects (e.g., the need of distinguishing between *information content* and *content of the HTML file*).

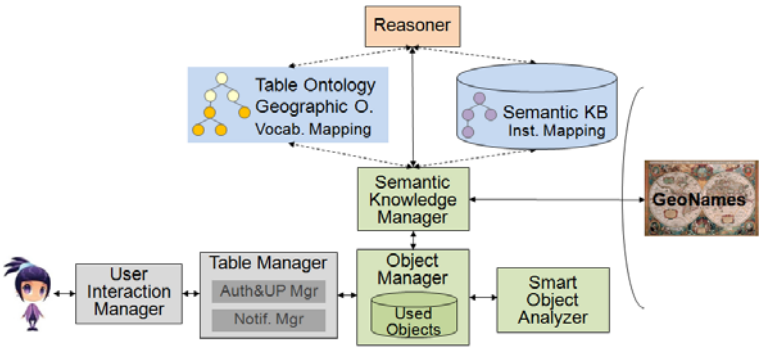


Fig. 1. Sem T++ architecture (relevant components).

4.2 Semantic Model

The core of our proposal is the Table Ontology, which models knowledge about information resources. It is grounded in the Knowledge Module of O-CREAM-v2 [15], a core reference ontology for the Customer Relationship Management domain developed within the framework provided by the foundational ontology DOLCE (Descriptive Ontology for Linguistic and Cognitive Engineering) [5] and some other ontologies extending it, among which the Ontology of Information Objects (OIO) [8]. The Table Ontology enables us to describe digital resources lying on tables as *information objects*, with properties and relations; for instance: a table object can have parts (e.g., images within a document), which are in turn information objects; it can be written in English; it can be stored in a PDF file, or it can be an HTML page; it has a content, which usually has a main topic and refers to a set of entities (i.e., it has several objects of discourse). Given such a representation, reasoning techniques can be applied, in order to infer interesting and useful knowledge; for example, if a document contains an image of Monte Bianco, probably the document talks about Monte Bianco.

The most relevant class in the Table Ontology is *InformationElement*: all table objects are instances of it. Moreover, we introduced some specific subclasses (e.g., *Document*, *Image*, *Video*, *Audio*, *EmailThread*, etc.), to provide a more precise characterization of the different types of objects that can lay on a table. In order to characterize such classes, we relied on: (1) a language taxonomy defined in O-CREAM-v2, representing natural, formal, computer, visual languages; (b) a set of properties (some of them inherited from O-CREAM-v2). A complete account of such properties is out of the scope of this paper; in the following we just mention the most important ones:

- *DOLCE* : *part*(*x*, *y*, *t*) – to represent relations such as the one between a document and an image or a hyperlink included in it.
- *specifiedIn*(*x*, *y*, *t*) – to represent relations such as the one between a document and the language it is written in (e.g., Italian).
- *hasTopic*(*x*, *y*, *t*) – to represent the relation between an information element (e.g., a document) and its main topic.

- *hasObjectOfDiscourse*(*x, y, t*) – to represent what a resource "talks about" (it is a subproperty of *OIO* : *about*).
- *identifies*(*x, y, t*) – to represent, for instance, the relation between a hyperlink and the resource it points to.

An important fragment of the proposed semantic model refers to particular properties which model *candidate relationships*. The idea is that the system, endowed with suitable axioms, can infer *candidate* features, mainly from included objects; for example, the Reasoner can infer that Monte Bianco is a *candidate* object of discourse of a document from the fact that the document itself includes an image of it. When the Reasoner infers such candidate relationships, the system asks the user for a confirmation: if (and only if) the user confirms, for instance, that Monte Bianco is actually an object of discourse of the document (*doc*), then a new relation *hasObjectOfDiscourse(doc, MonteBianco, t)* is added to the knowledge base. Analogous axioms are available for the *hasTopic* relation, to support the inference of *hasCandidateTopic* relationships.

Besides knowledge about information objects, all tables in Sem T++ are equipped with *geographic knowledge*. We decided to model geographic knowledge for two reasons: (a) Together with *time* (currently not modeled in Sem T++), *space* represents a cross-domain feature, which is – at least partially – represented by geographic knowledge[2]. (b) Geographic knowledge represents for us a testbed, i.e., an example of knowledge that characterizes the "content" of table objects: in the future, tables could be equipped with semantic knowledge about specific domains, by relying on the same mechanism we used for geography.

In order to provide tables with geographic knowledge, we exploited *GeoNames*, a huge, open geographical database containing over 10 million geographical entities. Moreover, we equipped the system with a Geographic Ontology and a *Vocabulary Mapping*, defining correspondences between Geography Ontology classes/properties and GeoNames *feature classes*/*codes*. GeoNames entities, in fact, are categorized into *feature classes* (e.g., *A*, corresponding to the generic concept of *country* or *region*) and further subcategorized into more specific *feature codes* (e.g., *A.ADM1*, corresponding to the concept of *primary administrative division of a country*). For each topic[3] mentioned on a table, the Semantic Knowledge Manager searches for corresponding GeoNames entities. If one or more results are found, the system currently asks the user to select the correct one, if any[4]; then a new individual is created in the Semantic KB, as instance of the Geographic Ontology class identified through GeoNames feature class/code (thanks to the *Vocabulary Mapping*). For example, an instance of the class *Mountain* is created for the topic *Mont Avic* (a mountain in Val d'Aosta, a northwestern Italian region). Moreover, a new mapping between such an instance and the corresponding GeoNames entity is created in the Semantic KB (see

[2] *Geographic knowledge* here means commonsense competence such as the ability to georeference places, or knowing that Monte Bianco is a mountain, and not scientific knowledge.

[3] In the current prototype, we consider only topics, although also objects of discourse could be taken into account.

[4] Mechanisms to support a (partially) automatic disambiguation are under study.

Instances Mapping in Fig. 1), thus making all information available in GeoNames (e.g., its location on a map) also available on Sem T++ tables.

The described model enables table participants to specify and combine different selection parameters in order to find objects on a table. For example, to get all email threads talking about *Monte Bianco trekking* (i.e., having it as main topic), the user can specify the following parameters: *topics={Monte_Bianco_trekking}, types={emailThread}*. Each parameter value corresponds to a user selection; object types are references to Table Ontology classes (*emailThread* in this example); values for other properties, e.g. *Monte_Bianco_trekking*, are references to individuals in the Semantic KB. Moreover, the user could provide more general queries, such as asking for all resources talking about mountains, thanks to the facts that topics (e.g., *Monte-Bianco*) are represented as instances of classes in the Geographic Ontology (e.g., *Mountain*). User queries are handled by the Semantic Knowledge Manager, which accesses the Semantic KB and invokes the Reasoner, in order to provide the objects matching the parameters.

4.3 Collaboratively Handling Semantic Descriptions of Digital Resources

One of the most challenging aspects of the presented semantic model is the creation and update of semantic representation of table objects. In fact, when a new object is created, or when an existing one is modified (e.g., when a table participant includes a new image or link in it), the corresponding semantic representation must be created or updated. In the following we will see how the collaborative management of semantic descriptions of table objects represents a step to face such a challenge, and, in particular, how this makes T++ tables actual *shared semantic spaces*.

Consider the new object case (the update case works in an analogous way): table participants can create new objects from scratch (e.g., when they start writing a new document), or they can add an existing resource (e.g., a bookmark pointing to a Web site) to the table. In both cases, the system builds a new semantic representation in four steps, in which different components play their role:

1. *Smart Object Analyzer*: some properties (e.g., mereological composition, types of the parts, formats) are automatically determined by the Smart Object Analyzer.
2. *Semantic Knowledge Manager/Reasoner*: other properties (e.g., candidate topics – see Section 4.2) are inferred by the Reasoner.
3. *User*: some of the inferred properties need a "user confirmation" (e.g., candidate topics); moreover, users can add properties (typically objects of discourse).
4. *Semantic Knowledge Manager/GeoNames*: the selected topic is linked to the corresponding geographic entity (through GeoNames), if any.

In order to describe the sketched process into more details, consider a usage scenario in which Maria participates in a table concerning the activities of a small ONG for environment safeguard, *Our Planet*, together with some other volunteers. Maria has to write an article for an online local newspaper, discussing the situation of a local old mule track in Champorcher (a small municipality in Val d'Aosta). Maria creates a new table object (an HTML document), writes some text in it, adds a picture of the envi-

possible to change the object type (an example scenario is described below) and to add/delete semantic properties, with the exception of the "contains" property, which refers to *DOLCE* : *part*(*x*, *y*, *t*): object parts (e.g., images included in a document) can be modified only by editing the object.

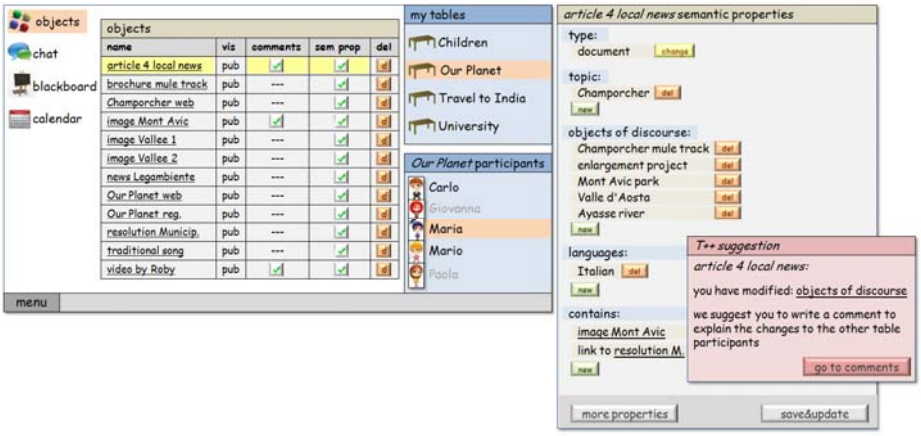


Fig. 3. Sem T++ user interface (mockup).

Mario finds that one of the objects of discourse (*Ayasse river*), up to him, is misleading: he thus decides to remove it and to add a new object of discourse (*Ourty alpine pasture*). When Mario clicks the "save&update" button, the semantic description is updated. Moreover, the table suggests Mario to write a comment in order to explain his decision to modify it (see Fig. 4, right-hand side); such a comment is attached to the modified object, but handled by the table, and is thus immediately accessible to other participants, as they sit at the *Our Planet* table.

A very similar case occurs when a user adds to a table an object by classifying it in a way which is then modified by another table user. For example, Maria could have added an image which, actually, is the scan of a document, and Mario could decide to change the object type from *Image* to *Document*. Again, Sem T++ suggests him to use the table–level annotation mechanism in order to share the reasons of the change.

This very simplified usage scenario shows the support provided by Sem T++ to the collaborative handling of semantic descriptions of table objects. Sem T++ has been designed having in mind a very "democratic" model of collaboration, in which people sitting at a table have the same privileges and thus all of them can modify objects, and in particular their semantic representations. However, Sem T++ also provides effective and easy–to–use table–level annotation mechanisms which represent a significative support to such a collaborative activity, together with the other standard mechanisms enabling discussion and communication among table participants, i.e., the *Blackboard* and the *Chat* (see [11] for details).

The coordination of collaborative activities taking place on Sem T++ tables deserves a final comment. As far as possible concurrent resource editing (e.g., document modification) is concerned, Sem T++ relies on existing applications, which typically

handle issues related to collaborative editing (such as Google Drive). As regards object management (adding, deleting, modifying table objects) and collaborative semantic annotation, the default configuration is the already mentioned simple policy allowing every table participant to freely modify, add, or remove all objects and annotations. Obviously, there are specific contexts (e.g., when tables are used to handle business activities, such as project management, for example) requiring more structured policies, modeling different user roles, privileges and specific workflows. We are working on the definition of mechanisms enabling users to configure such policies on each table.

5 Conclusions and Future Work

In this paper we presented Sem T++, an environment supporting users in collaborative resource management, by showing how formal semantic knowledge about *information objects* and their content can support an integrated, user-friendly management of heterogeneous shared resources. Sem T++ is a work in progress and many aspects, concerning the collaboration model and the user interface have not been discussed here. In particular, we are designing configuration mechanisms enabling users to define specific policies to coordinate collaborative activities, and specifically collaborative annotation management [18], on each table. Moreover, we are going to complete the integration of semantic modules into T++ prototype, in order to plan a user evaluation aimed at testing how the semantic model presented in this paper supports users in organizing and retrieving information objects on tables, but also the effort required to create and update semantic descriptions of table objects.

References

1. Abel, F., Henze, N., Krause, D., Kriesell, M.: Semantic Enhancement of Social Tagging Systems. In: Devedžić, V., Gašević, D. (eds.) Web 2.0 & Semantic Web, pp.25–56. Springer, Heidelberg (2010)
2. Ardissono, L., Bosio, G., Goy, A., Petrone, G.: Context-Aware Notification Management in an Integrated Collaborative Environment. In: UMAP 2009 workshop on Adaptation and Personalization for Web2.0, pp.23–39. CEUR-WS (2010)
3. Bardram, J. E.: From Desktop Task Management to Ubiquitous Activity-Based Computing. In: Kaptelinin, V., Czerwinski, M. (eds.) Beyond the Desktop Metaphor, pp.223-260. MIT Press, Cambridge, MA (2007)
4. Barreau, D.K., Nardi, B.: Finding and reminding: File organization from the desktop. ACM SIGCHI Bulletin 27 (3), 39-43 (1995)
5. Borgo, S., Masolo, C.: Foundational choices in dolce. In: Staab, S., Studer, R. (eds) Handbook on Ontologies, Second Edition, pp. 361–381. Springer, Heidelberg (2009)
6. Breslin, J. G., Passant, A., Decker, S.: The Social Semantic Web. Springer, Heidelberg (2009)
7. Drăgan, L., Delbru, R., Groza, T., Handschuh, S., Decker, S.: Linking Semantic Desktop Data to the Web of Data. In: Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A.,

Kagal, L., Noy, N., Blomqvist, E. (eds.) The Semantic Web – ISWC 2011. LNCS, vol. 7032, pp 33–48. Springer, Heidelberg (2011)
8. Gangemi, A., Borgo, S., Catenacci, C., Lehmann, J.: Task Taxonomies for Knowledge Content. Metokis Deliverable D07 (2005)
9. Geyer, W., Vogel, J., Cheng, L., Muller M. J.: Supporting Activity-Centric Collaboration through Peer-to-Peer Shared Objects. In: Group'03, pp.115–124. ACM Press, New York, NY (2003)
10. Goy, A., Petrone, G., Segnan, M.: Oh, no! Not Another Web-based Desktop!. In: 2nd Int. Conf. on Advanced Collaborative Networks, Systems and Applications, pp. 29–34. XPS Press, Wilmington, DE (2012)
11. Goy, A., Magro, D., Petrone, G., Segnan, M.: A Cloud-Based Environment for Collaborative Resources Management. Int. J. Cloud Applications and Computing 4(4), in press (2014)
12. Kaptelinin, V., Czerwinski, M. (eds.) Beyond the Desktop Metaphor. MIT Press, Cambridge, MA (2007)
13. Karger, D.R.: Haystack: Per-User Information Environments Based on Semistructured Data. In: Kaptelinin, V., Czerwinski, M. (eds.) Beyond the Desktop Metaphor, pp. 49–100. MIT Press, Cambridge, MA (2007)
14. Kim H., Breslin J.G., Decker S., Choi J., Kim H.: Personal Knowledge Management for knowledge workers using social semantic technologies. Int. J. of Intelligent Information and Database Systems 3(1), 28–43 (2009)
15. Magro, D., Goy, A.: A Core Reference Ontology for the Customer Relationship Domain. Applied Ontology 7(1), 1–48 (2012)
16. Muller, M. J., Geyer, W., Brownholtz, B., Wilcox, E., Millen, D. R. . One-Hundred Days in an Activity-Centric Collaboration Environment based on Shared Objects. In: CHI'04, pp.375–382. ACM Press, New York, NY (2004) Goy, A., Petrone, G., Segnan, M.: Oh, no! Not Another Web-based Desktop!. In: 2nd Int. Conf. on Advanced Collaborative Networks, Systems and Applications, pp. 29–34. XPS Press, Wilmington, DE (2012)
17. Sauermann, L., Bernardi, A., Dengel, A.: Overview and Outlook on the Semantic Desktop. In: 1st Ws on The Semantic Desktop at ISWC 2005, vol. 175. CEUR-WS (2005)
18. Su, A.Y., Yang, S.J., Hwang, W.Y., Zhang, J.: A web 2.0-based collaborative annotation system for enhancing knowledge sharing in collaborative learning environments. Computers and Education 55(2), 752–766 (2010)
19. Voida, S., Mynatt, E. D., Edwards, W. K.: Re-framing the Desktop Interface Around the Activities of Knowledge Work. In: UIST'08, pp. 211-220. ACM Press, New York, NY (2008)

Characterizing and Predicting Activity in Semantic MediaWiki Communities

Simon Walk[1] and Markus Strohmaier[2,3]

[1] Institute for Information Systems and Computer Media, Graz University of Technology, Graz, Austria
[2] GESIS - Leibniz Institute for the Social Sciences, Cologne, Germany
[3] Dept. of Computer Science, University of Koblenz-Landau, Koblenz, Germany

Abstract. Semantic MediaWikis represent shared and discretionary databases that allow a community of contributors to capture knowledge and to specify semantic features, such as properties for articles, relationships between articles, or concepts that filter articles for certain property values. Today, Semantic MediaWikis have received a lot of attention by a range of different groups that aim to organize an array of different subjects and domain knowledge. However, while some Semantic MediaWiki projects have been thriving, others have failed to reach critical mass. We have collected and analyzed a total of 79 publicly available Semantic MediaWiki instances to learn more about these projects and how they differ from each other. Further, we conducted an empirical analysis using critical mass theory on Semantic MediaWiki communities to investigate whether activity or the number of registered users (or a mixture of both) are important for achieving critical mass. In addition, we conduct experiments aiming to predict user activity and the number of registered users at certain points in time. Our work provides new insights into Semantic MediaWiki communities, how they evolve and first insights into how they can be studied using critical mass theory.

1 Introduction

Semantic MediaWikis are open repositories for structured data that can be edited by a community of users, who are interested in digitally modeling and representing domains. These Wikis have been used to capture knowledge from a wide variety of different domains, including for example beaches[4], games[5] or academic institutions[6].

Although Semantic MediaWikis have matured technologically, we still don't have a good understanding about the social processes behind them, e.g. why some Semantic MediaWiki communities are thriving and others are failing to reach critical mass. In this paper, we are using principles of critical mass theory

[4] http://beachapedia.org/
[5] http://nobbz.de/wiki/
[6] http://www.aifb.kit.edu/portal

to investigate activity and community growth in 79 publicly available Semantic MediaWikis with the goal of identifying and comparing factors that directly influence community growth and activity in said instances. In the context of online platforms, critical mass is often referred to as the amount or number of "something" (e.g., a feature or quality) that has to be reached for a system to become self-sustaining [8–10]. In terms of Semantic MediaWiki communities we want to know what this "something" is and if it is the same as it is for other systems and communities. In our empirical analysis we will look at *activity*, i.e. the accumulated number of changes contributed by the corresponding community to each Semantic MediaWiki at certain points in time. In addition, we will study the role of *community growth* via the number of accumulated unique users that have contributed to the Wikis at certain points in time. In particular, we are going to investigate whether activity or community growth (or a mixture of both) are important for achieving critical mass and predicting activity as well as community growth in Semantic MediaWikis at certain points in time. Answering these questions will fuel our understanding of how Semantic MediaWiki communities operate and evolve over time.

The remainder of this paper is structured as follows. In Section 2 we will present related work as well as work that has inspired the analysis conducted in this paper. A short characterization of the crawled Semantic MediaWiki instances and a description of the used methods for our analyses can be found in Section 3. The results and interpretations of our analyses are presented in Section 4. We conclude this paper in Section 5 and highlight future work.

2 Related Work

The work presented in this paper builds upon work in the areas of critical mass theory and collaborative ontology engineering.

2.1 Critical Mass Theory

In 1985, Oliver and colleagues [8–10] have discussed and analyzed the concept of critical mass theory by introducing so called production functions to characterize decisions made by groups or small collectives. Essentially, these production functions represent the link between individual benefits and benefits for the group.

They argue that when achieving critical mass of users, collective goods of groups are limited, thus interest can not be maintained longer than the limited (collective) resource allows for. In the case of online communities, the collective goods are not limited, theoretically allowing for an infinite increase in users. However, without users motivated in contributing, interest will decrease and critical mass will lose momentum and ultimately decelerate. In their work, three different types of production functions are identified: *Accelerating*, *decelerating* and *linear* functions (see Figure 2). The idea behind accelerating production functions is that each contribution is worth more than its preceding one. In a decelerating production function the opposite would be the case, resulting in

each succeeding contribution to be worth less than the preceding one. Until today it is still mostly unclear what these production functions look like for online communities and online production systems. Depending on the investigated or desired point of view, different aspects of these communities and online production systems can be used to calculate production functions. According to Solomon and Wash [15] it is still unclear which features of an online community characterize critical mass. One approximation they used was the activity and number of users for calculating and predicting critical mass in traditional WikiProjects. The authors argue that activity, for online production systems, after certain amounts of time is the best indicator of a self-sustaining system. In this work, we will adopt the same approach to characterizing critical mass for Semantic MediaWikis. Having an accelerating production function for the number of registered users and activity would indicate that users are interested in the collective good (e.g., the WikiProject) but also contribute to it (measured through activity). Achieving accelerating production functions for both of these factors critically promotes achieving critical mass. Once accelerating functions are reached, critical mass is likelier to follow, as interest (and pay-off) increases and user contributions rise, until the maximum potential of a system is reached.

The analysis of Oliver and colleagues [9] also highlights that different production functions can lead to very different outcomes in similar situations. For example, given an accelerating production function, users who contribute to a system are likely to find their potential contribution "profitable", as each subsequent contribution increases the value of their own contribution. Naturally, this increases the incentive to make larger contributions to begin with. Given a deceleration production function, users would not immediately see the benefit of large contributions, given that each subsequent contribution is increasing the overall value less, while more effort, in the form of larger contributions, is needed to turn a decelerating production function into an accelerating one.

Raban et al. [13] investigated factors that allow for a prediction of survival rates for IRC channels and characterized the production function of these chat channels as the best-fitting function for the curve that is generated when plotting the number of unique users versus the number of messages posted at certain (ascending) points in time.

Cheng and Bernstein [2] have analyzed concepts of activation thresholds, which resemble features that, when achieved, can help to reach and sustain self-sustainability. They created an online platform that allow groups to pitch ideas, which only will be activated if enough people commit to it.

Recently, Ribeiro [14] conducted an analysis of the daily number of active users that visit specific websites, fitting a dynamic model that allows to predict if a website has reached self-sustainability, defined through the shape of the curve of the daily number of active users over time. He uses two constants α and β, where α represents the constant rate of active members influencing inactive members to become active. β describes the rate of an active member spontaneously becoming inactive. Whenever $\frac{\beta}{\alpha} \geq 1$ a website is unsustainable and without intervention the daily number of active users will converge to zero. If $\frac{\beta}{\alpha} < 1$ and the number

of daily active users is initially higher than the asymptotic one, a website is categorized as self-sustaining.

2.2 Collaborative Ontology Engineering

The Semantic Web community has developed a number of tools aimed at supporting the collaborative development of ontologies. For example, Semantic MediaWikis [7] and some of its derivatives, such as OntoWiki and Moki [1, 4], add semantic, ontology modeling and collaborative features to traditional MediaWiki systems. In particular, OntoWiki represents a semantically enriched Wiki that supports collaborative ontology engineering, focussing on the acquisition of instance data and not the ontology or schema itself. MoKi is another collaborative tool that is implemented as an extension of a MediaWiki, which has already been deployed in a number of real world use cases.

Gil et al. [5,6] empirically analyzed different aspects of 230 different instances of Semantic MediaWikis, with a focus on the evolution of semantic features, such as properties and concepts. Among other things, they found out that in the investigated Semantic MediaWiki instances, categories were still much more popular than concepts. However, structured properties were used by all Wikis with a total of 50 instances exhibiting > 100 defined properties.

Protégé, and its extensions for collaborative development, such as WebProtégé [18] and iCAT [17], are prominent stand-alone tools that are used by a large community worldwide to develop ontologies in a variety of different projects.

To learn more about the nature of the engineering processes that occur when collaboratively developing an ontology, Pöschko, Walk and colleagues [12, 19] have created *PragmatiX*, a web-based tool to visualize and analyze a collaboratively engineered ontology.

Falconer et al. [3] investigated the change-logs of collaborative ontology-engineering projects, showing that users exhibit specific roles, which can be used to group and classify users, when contributing to the ontology. Walk et al. [20] applied Markov chains on the structured logs of changes of five collaborative ontology-engineering projects to extract sequential patterns. Pesquita and Couto [11] analyzed if the location and specific structural features can be used to determine if and where the next change is going to occur in a large biomedical ontology. Strohmaier et al. [16] investigated the hidden social dynamics when collaboratively developing an ontology providing new metrics to quantify various aspects to characterize collaborative engineering processes. Wang et al. [21] used association-rule mining to analyze user editing patterns in collaborative ontology-engineering projects.

3 Materials & Methods

We first characterize activity and community growth of our collected Semantic MediaWiki instances by applying principles of critical mass theory. We then continue our analysis and investigate if activity and community growth are good

predictors for determining the number of changes and users of Semantic MediaWikis at certain points in time. We comparing our results to what has been uncovered by Solomon and Wash [15] for WikiProjects, investigating if the number of users in the beginning stages of Semantic MediaWiki projects does play an important role for predicting activity and community growth. To study these effects in Semantic MediaWiki communities, we have crawled a total of 79 Semantic MediaWiki instances, which were all publicly available at the time of writing with the exception of *three* Wikis[7][8][9] that have already been taken offline.

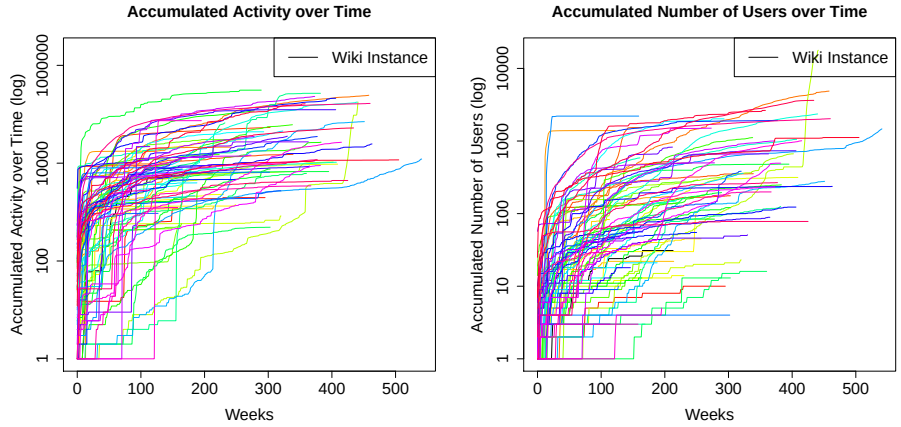
3.1 Semantic MediaWiki Datasets

The datasets used for the analyses in this paper are all randomly selected from different domains and vary in multiple aspects. Due to limitations in space we provide a summary of descriptive statistics for the entirety of our 79 Semantic MediaWikis[10] in Table 1. The number of users ranges from 1 to 85 users for our crawled Semantic MediaWiki instances with a mean of 6.7 unique users and a median of 2 users contributing to the different Wiki instances within the first month of its existence. Similar observations can be made for activity in Semantic MediaWiki communities. Initially we started our analysis with a little over 110 instances. However, due to restrictions necessary for our analyses we had to remove all Wikis with an observable lifespan of < 2 years, explaining the

[7] http://artfriendsgroup.com
[8] http://www.awaycity.com/wiki
[9] http://enlloc.net/hkp/w
[10] See http://www.simonwalk.at/wikis.html for a full list.

Table 1. Characteristics of the 79 datasets used for the prediction of activity and community growth. Community growth, represented as the number of users that have contributed at least 1 change, and activity, represented as the number of changes, are listed as average accumulated numbers over all Semantic MediaWiki instances after 1, 6, 12 and 24 months, as well as at the end of each project. Furthermore, we included the minimum (Min), median, maximum (Max) and standard deviation (SD) for each period. The differences between the Semantic MediaWikis are especially visible when looking at the standard deviation for activity and users during the first 2 years and at the end of our observation periods.

Number of	Timespan	Min	Mean	Median	Max	SD
Changes (Activity)	after 1 month	1	631.42	37	8,796	1,678.79
	after 6 months	1	2,840.91	904	63,547	7,622.06
	after 12 months	1	4,427.04	1,583	90,345	10,871.3
	after 24 months	1	10,595.18	4,694	159,502	21,264.96
	at end	459	41,338.49	11,534	310,933	68,225.41
Users (Number of Users)	after 1 month	1	6.7	2	85	13.09
	after 6 months	1	70.71	9	2,172	287.02
	after 12 months	1	102.35	23	2,203	296.25
	after 24 months	1	184.23	49	2,204	365.82
	at end	3	779.44	194	17,327	2,079.41
Duration	Weeks	113	291.44	295	541	110.08



(a) Accumulated Activity (b) Accumulated Users

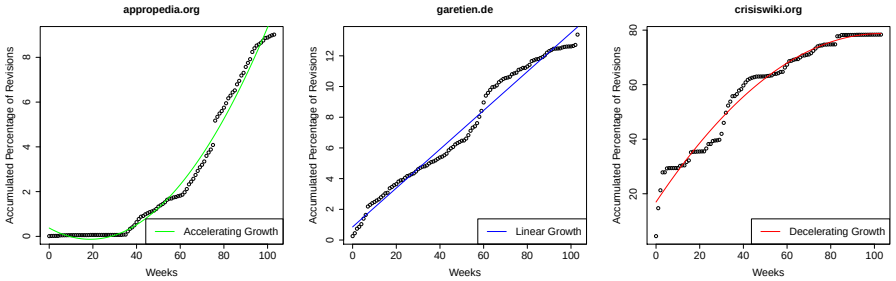
Fig. 1. Activity and Users per Semantic MediaWiki: The x-axes in both plots depict time in weeks, while the y-axes depict the accumulated amount of activity (represented as number of changes) and users during each corresponding week (log-scale). Each line represents one of the 79 Semantic MediaWiki instances. In both plots the differences in duration (x-axes), activity as well as number of users (y-axes) are visible.

minimum duration of 113 weeks. After removing all instances that did not meet the two year requirement we ended up with a total of 79 Semantic MediaWiki communities to investigate.

We have aggregated and accumulated activity and the number of users for each week from the inception of each Semantic MediaWiki until the date of the last observed change. The duration (observation period) of a Semantic MediaWiki instance starts with the first, and ends with the last change in our datasets. Figure 1(a) depicts this accumulated activity per week for every Semantic MediaWiki used in our analyses. Analogously, the accumulated number of users per week for every Wiki instance in our dataset is shown in Figure 1(b). The plots highlight the differences in observation lengths (x-axes), intensity of activity as well as number of users (y-axes, log-scale). Note that the number of users refers to all users that have contributed at least a single change. Anonymous users are represented by their ip address and are not filtered. These differences are also indicating that finding features that are suitable for fitting a general model to predict future information for Semantic MediaWiki communities is a difficult task.

3.2 Critical Mass Theory

We gathered the accumulated number of revisions and unique users after 1, 6, 12 and 24 months to determine the corresponding production functions for



(a) Accelerating Production Function

(b) Linear Production Function

(c) Decelerating Production Function

Fig. 2. Types of Production Functions: The x-axes in all three plots depict the time in weeks up to two years, while the y-axes depict the accumulated amount of activity during each corresponding week. The lines in each plot represent the best fitted linear or quadratic function for the observed data (circles).

each Semantic MediaWiki. As depicted in Figure 2, we plotted the accumulated number of users and activity versus elapsed weeks (one data point per week) and fitted a linear and squared function. As described in Solomon and Wash [15], if the squared function is not statistically significantly different from the linear function, the production function was classified as *linear*. If the difference is significant, depending on the priors of the second coefficient, representing the slope of the curve, we classified the production function as *accelerating* (positive coefficient) or *decelerating* (negative coefficient).

3.3 Activity & User Diversity Prediction

To determine if and to what extent features of Semantic MediaWiki communities are usable to determine the overall amount of activity and number of users after two years, we fit multiple regression models to the extracted activity and user data. To avoid any bias from differing overall timespans we use fixed time-intervals (1, 6 and 12 months) for extracting the input data for our regression models. Thus, we collected the accumulated amount of activity and users per week for each Semantic MediaWiki instance after 1, 6, 12 months to predict activity and the number of users after 24 months. Given that the extracted activity and number of users data from our 79 Semantic MediaWiki instances is over-dispersed, meaning that the variances are greater than the means (see Table 1), and the distribution of our extracted Semantic MediaWiki values resemble a negative binomial distribution, we can not use a standard logistic regression approach. Instead, we apply Negative Binomial Regression, which is used with count data that can not be smaller than 0 and follows a negative binomial distribution, on our datasets.

For each dependent variable, we are going to fit three negative binomial regression models, each using input data (activity and number of user) from

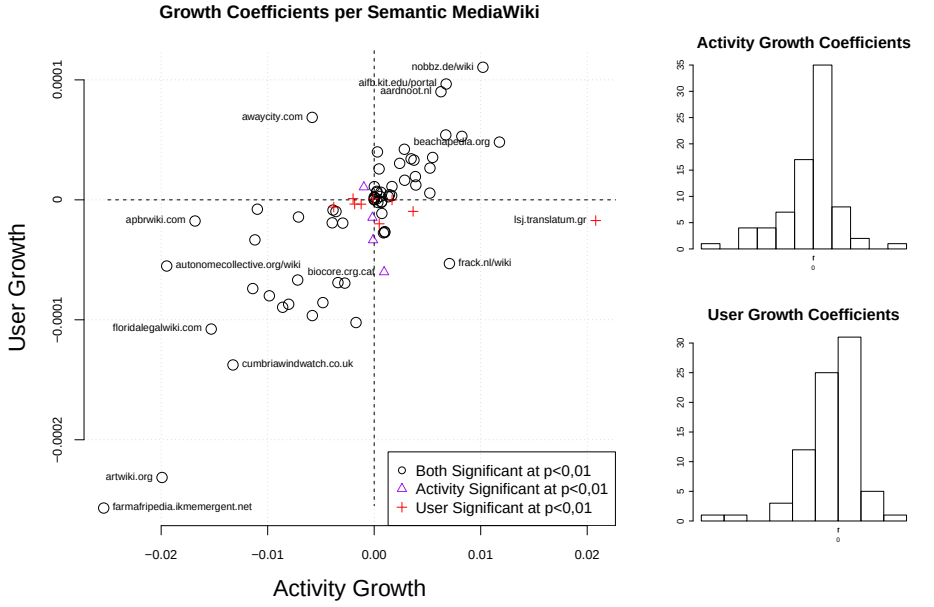
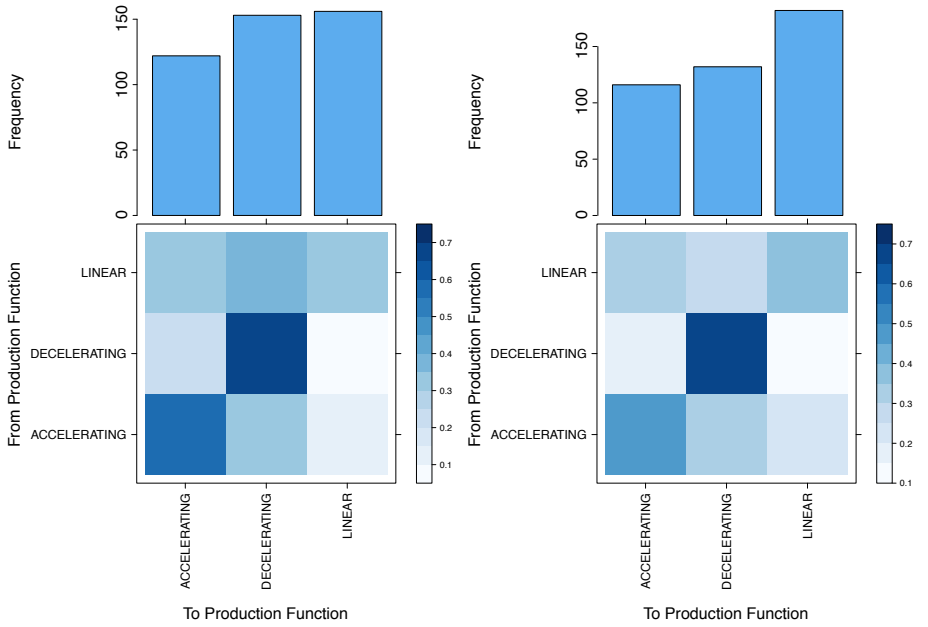


Fig. 4. The Growth Coefficient Distribution: This plot depicts the distribution of growth coefficients for our Semantic MediaWiki instances after after two years. The y-axis depicts the value of the number of users growth coefficients and the x-axis depicts the value of the activity growth coefficients. Each circle represents a Semantic MediaWiki instance with both production functions being significantly different (p-value < 0.01) from linear growth (e.g., http://aardnoot.nl/). Triangles (e.g., http://biocore.crg.cat/wiki) and crosses (e.g., http://lsj.translatum.gr) represent instances where only either the activity or user-based production function is significantly different from a linear function.

activity growth. As can be seen in the histograms, the values of the growth coefficients are equally scattered around positive and negative values. The larger the growth coefficient, the steeper the slope of the resulting production function. We calculated a Pearson correlation coefficient between (significant) user and activity-based growth coefficients of 0.75, indicating that critical mass for Semantic MediaWiki communities is constituted by an immanent correlation of the number of users and activity.

The median R^2 values for the fitted functions of activity and number of users at the different points in time range from 0.83 and 0.78 for the activity and user-based production functions in the first month to a R^2 of 0.95 for both after 2 years. These observed R^2 values represent a (rather) good fit, which also becomes more evident when looking at the sample fits in Figure 2 and the median R^2 values with input data from inception until year one.

To further characterize our investigated Semantic MediaWiki instances we have plotted the user diversity and activity growth coefficients extracted from



(a) Markov Chain Transition Matrix Activity-Based

(b) Markov Chain Transition Matrix User-Based

Fig. 5. Transition Probabilities of Production Functions: Figure 5 depicts the transition probabilities (darker means higher probability) between the different production function shapes of a fitted Markov chain model of first order for all Semantic MediaWikis. Each row in the depicted transition matrix corresponds to one type of production function (linear, decelerating or accelerating). The sum of each row is 1. The plot is always read from row to column, indicated by the axes label *From Production Function* and *To Production Function*. A histogram, highlighting the total occurrences of the different production functions, is depicted on top of the transition matrix.

the previously fitted production function models, using the accumulated number of users and changes from inception until the second year, for each Wiki individually. Figure 4 allows us to plot the different growth coefficients for all 79 Semantic MediaWiki instances, including information about the "intensity" of the observed slopes. Circles represent Semantic MediaWiki instances where both production functions were significantly different from a linear function. Triangles depict Semantic MediaWiki instances with significant activity-based production functions and linear user-based production functions. The crosses follow analogously to the triangles. This means that circles in the top right quadrant are Semantic MediaWiki communities that have an accelerating activity and user diversity production function. We can also see that Semantic MediaWikis have a tendency to exhibit the same production function for activity and user diversity, evident in the number of circles in the upper right and lower left quadrant

of Figure 4. To strengthen our observation we calculated a Pearson correlation coefficient of 0.75 for the different (significant) growth coefficient distributions. Thus, critical mass might be a mixture of the number of users and activity. We have trained a (first-order) Markov chain model, using the chronologically ordered sequences of extracted production functions after 1, 6, 12 and 24 months as input, to analyze whether Semantic MediaWiki communities frequently switch between production functions. For the user-based transition matrix (Figure 5(b)) accelerating and decelerating production functions tend to stay accelerating and decelerating. Linear production functions have a higher tendency to either switch to accelerating or stay linear, than become decelerating. The activity-based production functions (Figure 5(a)) exhibit very strong tendencies to stay at the same state (accelerating and decelerating). If a linear production function was determined for a Wiki, it is similarly likely to continue to exhibit a linear activity production function or switch to an accelerating production function, and is most likely to switch to a decelerating production function. In general, Semantic MediaWikis exhibit a high tendency to stick with their decelerating and accelerating production functions.

For managers of Semantic MediaWikis, this would mean that they would have to monitor both production functions and take action if already one of them is showing first signs of deceleration.

4.2 Factors that drive activity and user diversity

Given the observations made with critical mass theory in Section 4.1 we fitted 6 negative binomial regression models to predict the number of user and activity after two years, using the gathered input data from 1, 6 and 12 months. This method allows us to analyze if activity (and the number of user) after 2 years can best be explained by activity and/or the number of users of preceding points in time. The models are described in more detail in Tables 2 and 3. The goodness of fit for both models is described by the Akaike Information Criterion (AIC) and allows for relative comparisons between the different models. The closer the data that was used for fitting the models is to the target prediction time of two years, the better the model fits the data, evident in (minimally) decreasing AIC values.

When using negative binomial regression to predict the amount of activity after two years in Semantic MediaWikis communities the models show statistically significant effects for activity in all three models (1, 6 and 12 months) on the amount of activity after two years, when holding the number of users constant. When using the model fitted with data after 12 months to predict the activity in a Semantic MediaWiki community (see Table 2) with 500 and 600 users, with an activity of $10,000$ changes, we would expect to have $12,412$ and $12,342$ changes after two years respectively. The fitted model is clearly showing that more users, in the case of our observed Semantic MediaWiki communities, do not automatically mean an increase in activity after two years, which is in contradiction to our intuition after looking at the growth coefficients from the critical mass theory results.

Analogously, when holding activity on a constant level and predicting the number of unique users (or user diversity) after two years in Semantic MediaWikis (see Table 3), the amount of users already present after 1, 6 and 12 months is showing statistically significant effects on the number of users after two years. After 12 months we can determine statistical significance for activity and the (negative) interaction term as well. Similarly, when predicting the number of users in our Semantic MediaWiki communities after two years, using the fitted model after 12 months with $10,000$ and $11,000$ performed changes and 50 users, we would expect to have 99 and 101 users after two years. In contrast to the previous prediction we can observe the positive (and statistically significant for $p < 0.05$) influence of activity on the number of users after 2 years.

This actually means that, with a general model for Semantic MediaWiki communities, activity after two years can be predicted by looking at the activity after 1, 6 and 12 months. The number of users is not significant and, at least in our fitted model, has a negative impact on activity. This would mean (according

Table 2. Predicting Activity: The table depicts the configuration and results for the negative binomial regression model used to predict activity after two years. Input data for the models was the accumulated activity, unique users and an interaction term for both variables after 1, 6 and 12 months.

		Activity After 2 Years				
		Value	Std. Err (Coeff)	Std. Err.	θ	AIC
1 month	# Revisions	0.0004399**	0.000124	0.066	0.4977	1,582.4
	# Users	not sign.	not sign.			
	Revisions:Users	not sign.	not sign.			
6 months	# Revisions	0.00008919**	0.00002084	0.0743	0.5577	1,569.4
	# Users	not sign.	not sign.			
	Revisions:Users	not sign.	not sign.			
12 months	# Revisions	0.00009944**	0.00001445	0.0827	0.6145	1,558.6
	# Users	not sign.	not sign.			
	Revisions:Users	not sign.	not sign.			

$^{*}p < 0.05$; $^{**}p < 0.001$

Table 3. Predicting Users: The table depicts the configuration and results for the negative binomial regression model used to predict the number of users after two years. Analogously to the negative binomial regression model used to predict activity after 2 years, we have accumulated the number of users and activity for each Semantic MediaWiki after 1, 6 and 12 months and used this data (including an interaction term), as input for the listed regression models.

		Users After 2 Years				
		Value	Std. Err (Coeff)	Std. Err.	θ	AIC
1 month	# Revisions	not sign.	not sign.	0.0688	0.5159	947.56
	# Users	0.05386**	0.01839			
	Revisions:Users	not sign.	not sign.			
6 months	# Revisions	not sign.	not sign.	0.0892	0.6501	922.18
	# Users	0.009738**	0.001258			
	Revisions:Users	−0.0000004807**	0.0000001128			
12 months	# Revisions	0.00003025*	0.00001309	0.105	0.751	907.01
	# Users	0.006745**	0.001002			
	Revisions:Users	−0.0000001848*	0.00000008617			

$^{*}p < 0.05$; $^{**}p < 0.001$

to our model) that administrators and managers of Semantic MediaWikis should try to get as much content as possible, as soon as possible into their Wikis to ensure later activity. Critical mass for activity at later stages in a Semantic MediaWiki solely depends on activity in the beginning of a Wiki.

To predict the number of user after 2 years, the number of users after 1, 6 and 12 months are a significant factor. From month 1 to month 12 we can also observe a significance for the interaction term, which further increases in significance until activity becomes significant for the prediction at month 12. For increasing the number of users in a Semantic MediaWiki community, both, the number of users and activity (after a year) have to exhibit a positive (and significant) influence.

5 Conclusions & Future Work

The *main contribution* of this work is the characterization of activity and number of users using approaches of critical mass theory to gauge the viability of Semantic MediaWiki communities. We have studied 79 Semantic MediaWiki projects and their respective production functions over time. In addition, we have fitted negative binomial regression models to predict activity and the number of users after two years. Our approach is not specific to the projects under investigation but can be applied to other (Semantic) MediaWiki projects or collaborative online production systems at scale. In summary, we have found the following:

Semantic MediaWikis exhibit a wide range of evolving production functions: We have shown that the majority of observed Semantic MediaWikis start off with linearly growing activity and numbers of users. This changes within the first 6 to 12 months, which also apparently marks the timeframe where "something" determines if a Wiki will exhibit accelerating, decelerating or linear production functions after two years. At this point we leave it up to future work to further investigate, analyze and determine these influential factors.

Semantic MediaWikis suffer decaying information system lifecycles: The results obtained from the critical mass analysis, as well as the prediction experiment suggest that Semantic MediaWikis are prone to suffer from the vicious circles of decaying information systems. Meaning that Semantic MediaWiki instances that exhibit a decelerating production function (user and/or activity-based) are very likely to keep this decelerating production function, resulting in either less active users or lesser activity, which in turn triggers again less activity or less active users.

Successful Semantic MediaWiki communities start small: Our analysis suggests that the more content is produced by as few users as early as possible, the likelier it is for (our observed) Semantic MediaWikis to reach critical mass and exhibit the highest amount of activity after two years. This also means that the higher the number of users that contribute to a Wiki early on, the lower the amount of activity after two years is going to be. Surprisingly, after 12 months, the amount of activity becomes (positively) significant for the total number of users after 2 years. This indicates that after a certain amount of time (12 months), to attract more users, high activity in a Semantic MediaWiki has a positive effect.

One hypothesis to explain our observations could be that small groups around structured data projects are usually much more focused and devoted, as they need more background knowledge to contribute. However, this could imply that they do not necessarily need to reach critical mass for the number of users, but rather only in terms of activity, as their interest in creating a structured knowledge base already outweighs the efforts of contributing.

Summarizing, we believe that the work presented in this paper represents an important first step towards a better understanding of the factors that drive Semantic MediaWiki communities and their evolution. While our analysis has been initially performed on 79 Semantic MediaWikis and has been limited to user growth and activity, our method can be applied on a wider scale. Future work might focus on investigating additional instances, semantic properties, the evolution of the underlying knowledge base, different kinds of communities and types of Semantic MediaWikis with different motivations and interests, structural properties or additional dimensions of activity, such as passive usage logs (where *visits* are studied in addition to *edits*) or different kinds of activities and specific non-trivial phenomena, such as "edit wars", as well as other log data to expand our understanding of social and community dynamics in such systems.

References

1. S. Auer, S. Dietzold, and T. Riechert. OntoWiki–A Tool for Social, Semantic Collaboration. In *Proceedings of the 5th International Semantic Web Conference (ISWC 2006)*, volume LNCS 4273, Athens, GA, 2006. Springer.
2. Justin Cheng and Michael S Bernstein. Catalyst: Triggering collective action with thresholds. 2014.
3. Sean M. Falconer, Tania Tudorache, and Natalya Fridman Noy. An analysis of collaborative patterns in large-scale ontology development projects. In Mark A. Musen and scar Corcho, editors, *K-CAP*, pages 25–32. ACM, 2011.
4. Chiara Ghidini, Barbara Kump, Stefanie Lindstaedt, Nahid Mahbub, Viktoria Pammer, Marco Rospocher, and Luciano Serafini. MoKi: The Enterprise Modelling Wiki. In Lora Aroyo, Paolo Traverso, Fabio Ciravegna, Philipp Cimiano, Tom Heath, Eero Hyvönen, Riichiro Mizoguchi, Eyal Oren, Marta Sabou, and Elena Paslaru Bontas Simperl, editors, *Proceedings of the 6th European Semantic Web Conference on The Semantic Web: Research and Applications 2009*, pages 831–835, Berlin, Heidelberg, 2009. Springer.
5. Yolanda Gil, Angela Knight, Kevin Zhang, Larry Zhang, and Ricky J. Sethi. An initial analysis of semantic wikis. In Jihie Kim, Jeffrey Nichols, and Pedro A. Szekely, editors, *IUI Companion*, pages 109–110. ACM, 2013.
6. Yolanda Gil and Varun Ratnakar. Knowledge capture in the wild: a perspective from semantic wiki communities. In V. Richard Benjamins, Mathieu d'Aquin, and Andrew Gordon, editors, *K-CAP*, pages 49–56. ACM, 2013.
7. Markus Krötzsch, Denny Vrandecic, and Max Völkel. Semantic MediaWiki. In *Proceedings of the 5th International Semantic Web Conference 2006 (ISWC 2006)*, pages 935–942. Springer, 2006.
8. Gerald Marwell, Pamela E Oliver, and Ralph Prahl. Social networks and collective action: A theory of the critical mass, ill. *American Journal of Sociology*, 94(3):502–534, 1988.

9. Pamela Oliver, Gerald Marwell, and Ruy Teixeira. A theory of the critical mass. i. interdependence, group heterogeneity, and the production of collective action. *American journal of Sociology*, pages 522–556, 1985.
10. Pamela E Oliver and Gerald Marwell. The paradox of group size in collective action: A theory of the critical mass. ii. *American Sociological Review*, pages 1–8, 1988.
11. Catia Pesquita and Francisco M. Couto. Predicting the extension of biomedical ontologies. *PLoS Comput Biol*, 8(9):e1002630, 09 2012.
12. Jan Pöschko, Markus Strohmaier, Tania Tudorache, and Mark A. Musen. Pragmatic analysis of crowd-based knowledge production systems with iCAT Analytics: Visualizing changes to the ICD-11 ontology. In *Proceedings of the AAAI Spring Symposium 2012: Wisdom of the Crowd*, 2012. Accepted for publication.
13. Daphne R. Raban, Mihai Moldovan, and Quentin Jones. An empirical study of critical mass and online community survival. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work*, CSCW '10, pages 71–80, New York, NY, USA, 2010. ACM.
14. Bruno Ribeiro. Modeling and predicting the growth and death of membership-based websites. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, pages 653–664, Republic and Canton of Geneva, Switzerland, 2014. International World Wide Web Conferences Steering Committee.
15. Jacob Solomon and Rick Wash. Critical mass of what? exploring community growth in wikiprojects. 2014.
16. Markus Strohmaier, Simon Walk, Jan Pöschko, Daniel Lamprecht, Tania Tudorache, Csongor Nyulas, Mark A. Musen, and Natalya F. Noy. How ontologies are made: Studying the hidden social dynamics behind collaborative ontology engineering projects. *Web Semantics: Science, Services and Agents on the World Wide Web*, 20(0), 2013.
17. T. Tudorache, S. M. Falconer, C. I. Nyulas, N. F. Noy, and M. A. Musen. Will Semantic Web technologies work for the development of ICD-11? In *Proceedings of the 9th International Semantic Web Conference (ISWC 2010)*, ISWC (In-Use), Shanghai, China, 2010. Springer.
18. Tania Tudorache, Csongor Nyulas, Natalya F. Noy, and Mark A. Musen. WebProtégé: A Distributed Ontology Editor and Knowledge Acquisition Tool for the Web. *Semantic Web Journal*, 4(1/2013):89–99, 2013.
19. Simon Walk, Jan Pöschko, Markus Strohmaier, Keith Andrews, Tania Tudorache, Csongor Nyulas, Mark A. Musen, and Natalya F. Noy. PragmatiX: An Interactive Tool for Visualizing the Creation Process Behind Collaboratively Engineered Ontologies. *International Journal on Semantic Web and Information Systems*, 2013.
20. Simon Walk, Philipp Singer, Markus Strohmaier, Tania Tudorache, Mark A Musen, and Natalya F Noy. Discovering Beaten Paths in Collaborative Ontology-Engineering Projects using Markov Chains. *Journal of Biomedical Informatics*, January 2014.
21. Hao Wang, Tania Tudorache, Dejing Dou, Natalya F Noy, and Mark A Musen. Analysis of user editing patterns in ontology development projects. In *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*, pages 470–487. Springer, 2013.

User Profile Modelling in Online Communities

Miriam Fernandez[1], Arno Scharl[2], Kalina Bontcheva[3], Harith Alani[1]

[1]Knowledge Media Institute, The Open University, UK
{m.fernandez, h.alani}@open.ac.uk
[2] Department of New Media Technology, MODUL University Vienna, Austria
scharl@modul.ac.at
[3] Department of Computer Science, University of Sheffield, UK
k.bontcheva@dcs.shef.ac.uk

Abstract. With the rise of social networking sites user information is becoming increasingly complex and sophisticated. The needs, behaviours and preferences of users are dynamically changing, depending on their background knowledge, their current task, and many other parameters. Existing ontology models capture demographic information as well as the users' activities and interactions in online communities. These vocabularies represent the raw data, but actionable knowledge comes from filtering these data, selecting useful features, and mining the resulting information to uncover the most salient preferences, behaviours and needs of the users. In this paper we propose reusing and re-engineering ontological resources to provide a broader representation of users and the dynamics that emerge from the virtual social environments in which they participate.

Keywords: Semantic Web, Social Web, User Profile

1 Introduction

It is crucial for service providers to adequately understand the *needs*, *preferences* and *behaviours* of their users to ensure that their services are delivered to the right people at the right time. However, achieving such understanding of the user, based on a wide range of inter-dependent attributes and implicit information, is a complex research task. The user's current situation, past history and social environment need to be combined and integrated. Data about the time and activity of users should be linked with the users' past information to understand their current situation; previous activities and interactions should be taken into account to interpret and fully understand this situation; relations with other people and other user behaviour in similar contexts should be also considered and captured.

With the emergence of the Social Web and social networking sites such as Facebook, Twitter, Google+ and YouTube, a vast amount of personal information is created on a daily basis. The scale of this personal and social context data has a huge potential to improve the coverage of user modelling approaches and enhance the effectiveness of adaptive systems.

Multiple efforts have emerged from the Semantic Web (SW) community to target this problem. Vocabularies in standard representation formats, such as RDF and OWL, have been developed, to model users and their social context. Examples of these vocabularies include FOAF – Friend of a Friend [6] and extensions like the Relationship Vocabulary [17], SIOC [2;9], OPO – Online Presence Ontology [15], or MOAT – Meaning of a Tag [7]. While these ontologies do indeed capture user interactions within online communities, they do not model more dynamic user aspects such as behavioural evolution within the community. The aforementioned vocabularies represent the raw data, but actionable knowledge comes from filtering the vocabularies, selecting useful features, and mining the profile data to uncover the most salient preferences, behaviours and needs of the users.

In this paper we present a user profile model that goes beyond capturing raw data from user activities and interactions to capture the interpretation of these data within particular contexts. To generate this user profile model, we reuse existing ontological resources for modelling users and their social context, and extend this knowledge with well-known features extracted from current social media analysis methods [19, 32, 33, 34, 35]. By modelling and storing these features we enable inferences to be made over a richer layer of data, allowing the dynamic learning of user preferences, needs and behaviours.

To generate user profiles, we have followed the NeOn methodology [18], and its guidelines for reusing and re-engineering ontological resources. According to this methodology, three main steps should be followed: (1) select the most suitable ontological resources to be reused; (2) carry out the ontological resource re-engineering process to modify the selected ontological resources, and (3) assess if the modified/new ontology fulfils the ontology requirement specification. The ontology requirement specification states why the ontology is being built, what its intended uses are, and which requirements the ontology should fulfil.

The main use case for the presented user profile model is a collective awareness platform currently being built as part of *DecarboNet* (www.decarbonet.eu), a research project that aims to increase environmental awareness, trigger behavioural change and track the resulting information diffusion patterns across various social networks. The collective awareness platform of DecarboNet will consist of a knowledge co-creation environment embedded into an existing media analytics platform available at www.ecoresearch.net/climate [41], an upcoming social media application in the tradition of games with a purpose, and a portfolio of analytic services to identify patterns in both the structure of social networks as well as the content communicated between the nodes of these networks. The dynamic user and context models of DecarboNet will help to integrate the observable data flows across these components. They will enable analysts to capture the role of users in social innovation processes, assess their environmental knowledge and information seeking behaviour, and measure their engagement level within the DecarboNet community.

The remainder of this paper is structured as follows: Section 2 presents the ontology requirements specification. Section 3 outlines the available ontological resources. Section 4 describes the re-engineering process and the proposed extensions and modifications to the selected ontological resources. Section 5 discusses the results and concludes the paper.

2 Requirements Specification

The goal of systems that use personal data is to gain the capability to adapt aspects of their functionality or appearance to the preferences and needs of their users. To do so, the system must have an internal representation (i.e. a model/profile) of the user. Approaches that generate these user profiles generally distinguish among: (i) *modelling* – which information defines the user? (ii) *representation* – which formats and structures are used to represent the user profile? (iii) *acquisition and update* – how the previous identified information is acquired and evolves over time?

In this paper we focus on the problem of *user modelling*. Standard semantic formats such as RDF and OWL have been selected to represent our proposed user profile. Regarding the problem of user profile acquisition and update, we provide a brief overview of how the proposed user profile is currently being acquired and updated. Table 1 presents the Ontology Requirements Specification for our proposed user model following the NeOn methodology [18]. This methodology proposes the development of a filling card, and more particularly, a set of competency questions to assess whether the ontology fulfils the requirements. The resulting filling card for our user profile ontology is displayed below:

Table 1: Ontology Requirements Specification

Purpose: The purpose of building the user profile ontology is to provide a reference model for capturing the dynamics of user profiles in online communities
Scope: The scope of this ontology is the user in the context of online communities
Implementation Language: The ontology is implemented in OWL
Intended Users: The indented users of this ontology are adaptive systems or social media analysis modules. No human users are intended for this ontology
Intended Uses: - To dynamically infer, for a user, her exhibited behaviour within a particular online community and moment in time - To dynamically infer, for a user, her needs within a particular online community and moment in time - To dynamically infer, for a user, her preferences within a particular online community and moment in time - To infer, for a user, her personality from her previous actions across online communities
Ontology Requirements: (a) Non-Functional requirements: none (b) Functional requirements: defined by four main competency questions: - **CQ1**: What is the behaviour that user *u* adopts in the online community oc_x during the time period t1-t2? - **CQ2**: What are the needs of user *u* in the online community oc_x during the time period t1-t2? - **CQ3**: What are the preferences of user *u* in the online community oc_x during the time period t1-t2? - **CQ4**: What is the personality of user *u*?

As we can see in this filling card, the intended use of the ontology is to be able to infer, the *behaviour*, *needs*, *personality* and *preferences* of the user within a particular online community and moment in time. These concepts are discussed in detail in Section 4. User profiling based on the presented ontology and concepts aims to fulfil the Decarbonet requirements by enabling a structured analysis of behaviour patterns. Detecting user types (e.g. those likely to change their behaviours as a result of a specific intervention strategy) and updating dynamic user models on-the-fly will be used by the DecarboNet platform to guide data acquisition and filtering processes, form ad-hoc communities based on shared interests, devise effective engagement strategies, and provide tailored information services for citizens.

3 Ontology Selection

In this section we explore the ontologies developed so far to semantically model users particularly, in the context of online communities:

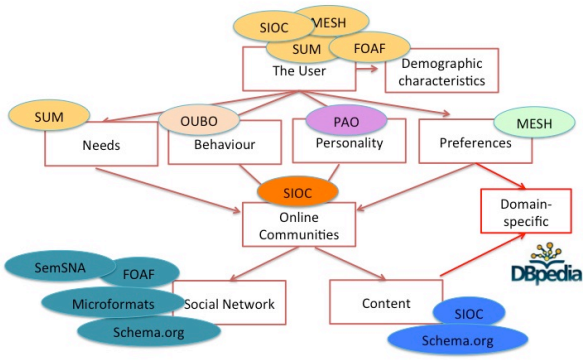


Fig. 1: Existing ontologies capturing the different user profile aspects in the context of online communities.

As we can see in **Fig. 1** our proposed user profile model aims to capture multiple aspects of the *user* and the online communities in which she participates. Among the aspects that the model aims to capture for the user we can highlight static elements, such as her demographic characteristics; but also more complex and dynamic elements, such as her needs, her behaviour, her personality and her topic (domain-specific) preferences. These aspects are inferred from the actions and interactions of the user within an online community. The online community provides information not only about the *social network* of the user (the people she interacts with) but also about the *content* she produces. Among the ontologies that aim to capture user information within the context of online communities we can highlight:

FOAF, the Friend Of A Friend vocabulary [6] describes people, their properties such as name, homepage, etc., and the social connections of different users by means of the foaf:knows relationship. This property allows people to be linked to one another across social web platforms.

The *Schema.org* vocabularies, [10] agreed among the major search engine providers (Google, Bing, Yahoo! and Yandex), are able to capture the knowledge about people and their social networks. They provide a collection of tags to define item types (Person, Place, Organisation, Review, Event, etc.) and social relations (knows, colleague, children, parent, sibling, relatedTo. etc.).

Microformats [11] provide vocabularies to describe people as well as their social connections. The hCard micro format [12] represents people and their attributes such as given name, family name, URL, email, etc. The XFN microformat [13] captures users' social networks by representing the relations between people via the 'rel' attribute (e.g., <a href="http://jeff.example.org" rel="friend met">).

Semantic Social Networks Analysis [14] (SemSNA) provides an understanding of the structure of networks, including richer representations of social links: cyclic path, directed path, betweenness, centrality, etc.

The *Online Presence Ontology* (OPO)[15] models the online presence of users. It proposes classes to describe the findability, noticeability or online status of users as well as their actions (working on a project, reading, listening, etc.)

The *SIOC* (Semantically Interlinked Online Communities) ontology [9], originally designed to capture the knowledge of discussion boards, models not only users and social interactions, as in previously mentioned works, but also content, and the reply-chain in which this content has originated. This ontology is based on, and reuses classes and relations from, several well-known ontologies such as the Friend Of A Friend (FOAF) vocabulary [6] and Dublin Core Metadata Terms (dcterms). [16]

Other works have also attempted to reuse some of these vocabularies to provide community-focused descriptions. One of these examples is the Facebook Open Graph Protocol, which can model and interlink users and objects within the Facebook social network. As opposed to this model, SIOC is not tailored to any particular social networking platform.

To the best our knowledge, SIOC is the most complete and generic ontology developed to date to capture the knowledge of online communities. It does not only capture knowledge about users and their social interactions, as Microformats or FOAF do, but it also captures knowledge about the content and the content generation process. Additionally, as opposed to the Facebook Open Graph Protocol, its purpose is more generic and has not been designed with a particular online community in mind, building a crucial base for data integration and unification across different online communities. Giving its popularity and adoption, SIOC is selected as the base of our proposed user model.

In addition to the previously presented ontologies, which capture demographic information about the users, as well as their actions and interactions within online communities; we have surveyed ontologies aiming to capture more complex user aspects. Capturing users' behaviour, personality, needs, or preferences can enable systems to provide better adaptations of their functionality or appearance.

While multiple ontologies can be found in the literature that aim to capture the domain or topic preferences of users for personalisation and recommendation [22, 36, 37], very few ontologies have been proposed to capture the behaviour or users, their needs or their personality.

Regarding behaviour modelling we can highlight the works of Ankolear et al. [38] and Rowe et al. [19]. Ankolear et al. [38] describe user roles in problem-solving

communities: bug fixer, bug reporter, contributor, developer, etc. While this work is focused on a specific type of online communities, Rowe et al. [19] propose a more generic model, the Open University Behaviour Ontology (OUBO); able to capture different user roles for online communities with different focuses.

Regarding user needs interpretation in the context of online communities, current research has focused on capturing user needs by: (i) applying well-known social theories such as Maslow's pyramid of needs or the self-determination theory to the world of online communities [27, 28, 29] or, (ii) explicitly asking users about their needs via questionnaires [30, 31]. The Semantic Web User Model (SWUM) [39] captures some of these user needs, as well as elements of the behaviour and personality of users.

Regarding the interpretation and understanding of users' personality in the context of online communities, current research has also focused on applying well-known social theories, such as the big-five personality traits [32, 33, 34]. The Personality Assessment Ontology (PAO) [40] captures this personality theory. Other ontologies like SWUM capture personality in the form of user characteristics such as "kind", "warm", "calm". While ontological models like SWUM capture needs and preferences of the users, they do not consider the dynamics of these user aspects. E.g., a user may exhibit different needs in different online communities or at different points in time. User's behaviours, needs and preferences are dynamic aspects and should be captured in context.

4 Dynamic User and Context Modelling

As we have seen in our previous section, existing vocabularies, either (i) capture raw data about the user and her social environment, but do not model more complex and dynamic aspects of the user (needs, personality, preferences, etc.) or (ii) they model more complex aspects of the user but they just capture a snapshot, from which the evolution over time, or in different communities, cannot be inferred. Our proposed user model aims to address these issues by reusing and extending some of the previously presented vocabularies.

4.1 Modelling User Actions and Interactions in Online Communities

To capture data about the user and her actions within online communities we have chosen SIOC as the base of our user profile model. SIOC makes use of the class sioc:*UserAccount*. This class reuses properties from other vocabularies, such as: *sioc:name*, which captures the name of the user, *dc:created*, which captures the time and date the user account was created, *sioc:creator_of,* which links the user to the content she generates or *foaf:knows,* which links the user with her social network.

To model the content creation process and the interaction of the user with other community members SIOC makes use of classes such as *sioc:Container*, *sioc:Thread* and *sioc:Post*. The class *sioc:Post* has the property *sioc:has_creator* that links the post with a particular user account. This class also has the property *sioc:hasParent*,

that links the Post with a particular Thread. The properties *sioc:reply_of* and *sioc:has_reply,* link the post to other posts, and the properties *sioc:content* and *sioc:created* capture the text of the content and the date/time it was posted. The class *sioc:Thread* is also linked to a particular *sioc:Container* (forum, blog, etc.) by the property *sioc:has_parent*. By reusing the SIOC classes and properties our model captures demographic information about the user as well as her actions within online communities (when the user posts a message, when she replies, etc.)

4.2 Modelling User Context

There are two types of context we wish to define to capture user dynamics and evolution: *location* and *time*. For the former we can use SIOC classes such as *sioc:Forum*, *sioc:Community*, etc., to represent the social virtual environment where the user, defined as an instance of *sioc:UserAccount*, is participating. To model time we reuse the class *oubo:TimeFrame* from the OUBO ontology [19]. The class *oubo:TimeFrame* defines a given time period in which users' features (see Section 4.3) are computed. We combine the temporal and location context aspects into a single context instance using the class *social-reality:C*. The class *social-reality:C* is reused from Hoekstra's work [20] and is used to represent a higher-level notion of context that can be used to include additional contextual information, apart from location and time.

4.3 Modelling User Behaviour

To capture and infer user behaviour, we propose an extension of the OUBO ontology [19]. This ontology uses SPIN rules to infer the role (*oubo:Role*) that a user (*sioc:UserAccount*) has in a given context (*social-reality:C*). To infer the role that a user assumes (Leader, Follower, Broadcaster, etc. [19, 35]) we need to capture fine-grained information about the user (user features). We propose to extend the OUBO ontology and to model user features under six different behavioural dimensions:

- *Popularity:* the popularity of a user measures whether the user is being liked, admired, or supported by many people.
- *Engagement*: the engagement of a user measures up to which level the user is committed to the community.
- *Initiation*: the initiation of a user measures how much the user instigates discussions and asks questions.
- *Contribution*: the contribution of a user measures the extent to which the user contributes or replies to threads initiated by other users.
- *Content Quality*: The content quality of a user measures her level of expertise and how useful her posted content is for the topic under discussion.
- *Focus Dispersion*: the focus dispersion of a user measures whether the user disperses his/her activity across many forums/sub communities/sub topics or concentrates his/her activity in a few forums/sub communities/sub topics.

User behavioural features can be computed using a variety of metrics. Table 2 presents some of the most common metrics used in the literature.

upo:UserDimension is associated with one or more *upo:Metric* by the relation *upo:hasMeasureFunction*.

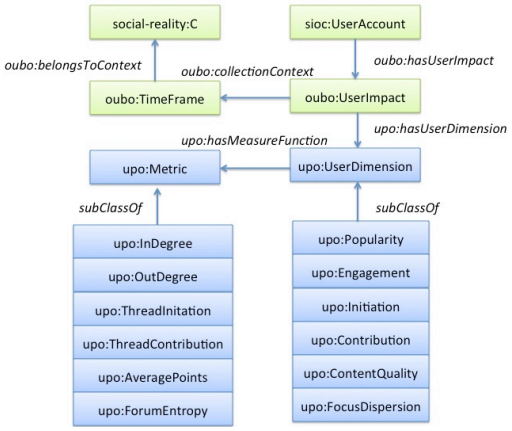


Fig. 2: Extensions proposed to capture the different behavioural dimensions

To infer the different roles that a user adopts over time we apply semantic rules encoded using SPIN (e.g., if popularity=high and contribution=high then role=leader). For more details of the role extraction process the reader is referred to [19]. Note that using the notion of context, features, and SPIN rules the proposed ontology fulfils **CQ1**, i.e., it can infer the behaviour (role) that user *u* adopts in an online community oc_x during a particular time period.

4.4 Modelling User Preferences

To capture and model user preferences semantically we build on our previous work and reuse parts of the MESH ontology [21]. This ontology has been used to model user preferences and has proven its effectiveness for personalisation and recommendation tasks [22]. Ontology concept-based preferences are more precise, and reduce the effect of the ambiguity caused by the use of keyword terms. For example, a preference stated as "*ProgrammingLanguage:java*" (this reads as the instance Java for the Programming Language class) lets the system understand unambiguously the preference of the user does not refer to the pacific island. Additionally, the multiple relations modelled in ontologies and their inference capabilities allow the inference of underlying user interest. For instance if a user is interested in *skiing*, *snowboarding* and *ice hockey* it can be inferred, with a certain degree of confidence, that the user is globally interested in *winter sports*.

To model user pferences we extend the class *sioc:UserAccount* with the properties *mesh:semanticInterest* and *mesh:itemRatings*. The property *mesh:semanticInterest* links the user with the class *mesh:SemanticInterest*. This class is modelled as a vector of *mesh:WeightedConcept* that represent the preferences of the user in terms of semantic concepts. *A mesh:WeightedConcept* class is represented by three main

properties *mesh:concept*, that captures the conceptual preference of the user, *mesh:weight*, that represents the preference score for that particular concept and *mesh:timestamp*, that represents the moment in time in which the user expressed interest for that particular concept.

To populate the preferences of our user profile model we make use of existing semantic annotators that are able to extract the subset of concepts expressed by the users in their posts. At the moment we make use of TextRazor to extract these concepts [23] but other systems, such as Alchemi API [24] or DBPediaSpotlight [25] could also be used. Note that TextRazor extracts concepts from DBPedia and FreeBase, to our knowledge, two of the most complete knowledge bases up to date Note that concepts with a confidence score lower than 3, in a scale from 0.5-10, are discarded. The preference level of the user for the concept is based on a sentiment analysis of the content. The SentiCircles sentiment analysis approach is used to compute the sentiment of the extracted concepts [26].

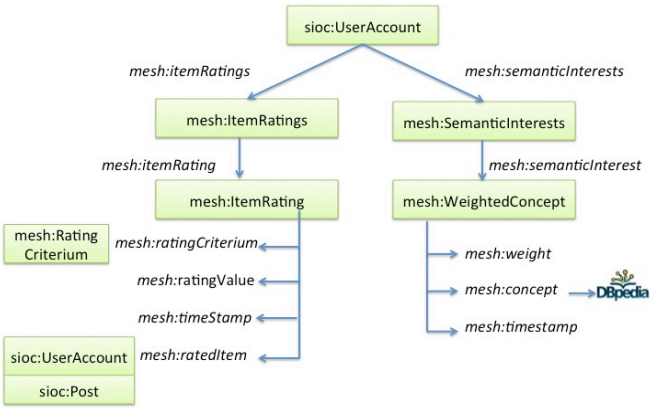


Fig. 3: Reused classes and properties of the MESH ontology to model preferences

In addition to the modelling of concept-based user preferences we also capture preferences in terms of ratings. In certain online communities users can provide ratings to express their preference for other users or for certain content. Preferences in terms of ratings are modelled with the class *mesh:ItemRatings*. This class is linked to a *sioc:UserAccount* via the property *mesh:itemRatings*. The class *mesh:ItemRatings* is a vector of *meshItemRating*. This class, which represents a rating score is modelled using four main properties: *mesh:ratingCriterium*, which represents the criterion/method used to rate the items (score, stars, etc.); *mesh:ratingValue*, which represents the value assigned by the user, *mesh:ratedItem*, which represents the item for which a preference has been established, and *mesh:timestamp*, which represents the moment in time in which the user rated that particular item.

As in the case of behaviour modelling, preferences are also dynamic, i.e., only certain user preferences should be consider in each particular context or sioc-reality:C. To dynamically select user preferences we build on our previous approach [22]. More specifically, the selection of applicable preferences in a particular context sioc-reality:C is based on two main principles:

- If a concept keeps occurring along time, this concept is selected within the current context as a long-term preference of the user.
- If a concept occurrence is very high on the recent short period, this concept can be selected in the current context as a short-term preference of the user.

In our extension (see Fig. 4) we define the classes *upo:LongTermPreferences* and *upo:ShortTermPreferences* to capture long and short term preferences in a particular context, *socialReality:C*; and the classes *upo:ConsistentFrequencySelection* and *upo:HighFrequencySelection* to model the methods used to capture long and short term preferences respectively. Note that by modelling and applying these methods the ontology fulfils **CQ2**, i.e., it is able to infer the needs of user *u* in the online community oc_x during a particular time period.

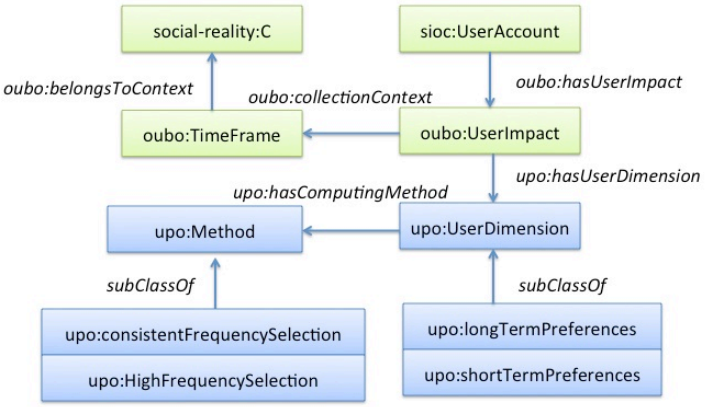


Fig. 4 Extension of the MESH ontology to capture dynamic preferences

4.5 Modelling User Needs

Our approach towards modelling and inferring user needs is based on the principle that needs are mirrored to certain online actions. For example, the action of commenting or replying to a post reflects the user's intention to help other users (*altruistic need*) as well as an aim to interact with other members of the community (*socialisation need*). In our model we aim to capture four main user needs that have been recurrently found in the literature [27, 28, 29, 30, 31]:

- *Information Need:* when a user initiates a discussion he or she is reflecting an information need. Users under this need are focused on solving their problems with the help of the community.
- *Socialisation Need*: when a user intentionally interacts with other users he reflects his need for socialisation.
- *Reputation need*: the reputation can be reflected on the number of points (ratings/likes/favourites) or replies the user receives from the community.
- *Altruistic Need:* the altruistic need, or need to help others, is reflected on the number of replies that a user provides to other people's questions. Users with a

high altruistic need share their knowledge with the community and spend their time and expertise to benefit others.

As in the case of user behaviour, several metrics can be used to represent needs (see Table 2). For example, the *Information Need* is linked with metrics such as: *Thread-initiation ratio* or *Self-reply ratio*. This last metric measures the number of replies given by user u_i in reply to his/her own threads. It is an indication of how strongly a user pursues obtaining an answer from the community. The *Socialisation Need* is reflected in metrics such as *out-degree*, which measures the proportion of users that the user has contacted/reply to. The *Reputation Need* is reflected in metrics such as the *average points*/likes/favourites, received by other users. The *Altruistic Need* is reflected in metrics such as *replies-count*, which measures the total number of replies written by a user within the community. Note that our proposed model can be extended to capture different user needs and different associated metrics. As shown in Fig. 5 the class *upo:UserNeed* and its corresponding subclasses are used to capture the defined user needs. Associated metrics are modelled under the class *upo:Metric*, such as *upo:SelfReplyRatio*. Note that we have decided not to reuse classes of the SWUM ontology, since this ontology captures needs in terms of features not measurable in the context of online communities, e.g., "sexual intimacy".

To infer whether a user (*sioc:UserAccount*) presents one particular need in a given context (*social-reality:C*), we apply semantic rules encoded in SPIN (e.g., if upo:Thread-initiation=high and upo:SelfReplyRatio=high then upo:InformationNeed=high). By following the same approach as for computing user behaviour [19] the ontology fulfils **CQ3**, i.e., it can infer the needs that user *u* adopts in an online community oc_x during a particular time period.

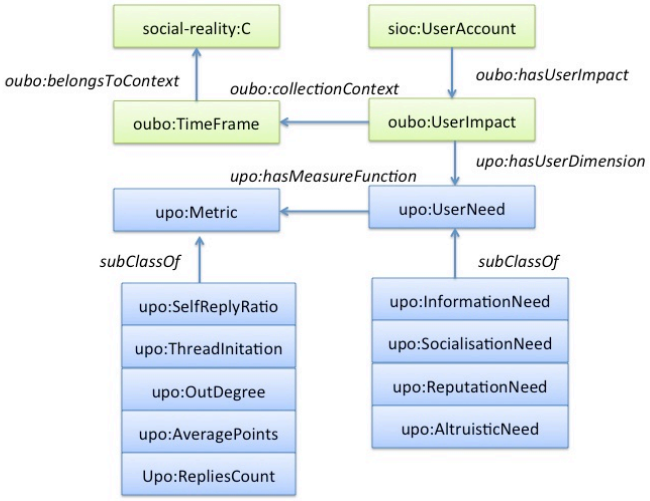


Fig. 5: Extensions to capture UserNeeds.

4.5 Modelling Personality

There is a body of research in online communities that has attempted to model and predict personality. These predictions are mainly based on the Big Five personality model [32, 33, 34], which defines personality in terms of five dimensions:

- *Openness to experience:* openness indicates the degree of intellectual curiosity.
- *Consciousness*: indicates a tendency to be organized and dependable.
- *Extroversion*: indicates sociability and the tendency to seek stimulation in the company of others.
- *Agreeableness*: indicates a tendency to be compassionate and cooperative.
- *Neuroticism*: indicates a tendency to experience unpleasant emotions easily.

To capture these personality dimensions, we reuse classes of the PAO ontology such as: *pao:Personality, pao:PersonalityDimension, pao:Agreebleness, pao:Conscientiusness, pao:Extraversion* and *pao:Neuroticisim*. Recent studies have shown that the previous personality dimensions are reflected, and can be predicted, with certain degree of accuracy, from the online actions of users within online communities [32, 33, 34]. Quercia et al. [32], for example, predict users' personality in Twitter by using features such as "following", "followers" and "listed counts". These metrics are modelled in our profile as *upo:OutDegree, upo:InDegree, upo:FocusDispersion*. Note that research has consistently shown that people's personality scores are stable over time [15]. Therefore, personality in our model is not considered in context. To infer the levels of personality dimensions for each user *u* we define SPIN rules that capture the prediction model defined by Quercia et al. [32], e.g., (if upo:OutDegree=high and opo:InDegree=high then pao:Extraversion=high). By defining these rules the proposed model fulfils **CQ4**, i.e., it can infer the personality of a particular user *u*.

5 Discussion and Conclusions

This paper presents a semantic approach to user profile modelling that goes beyond collecting raw data from user activities in online communities. This approach captures the interpretation of these data within particular contexts, enabling the inference of user needs, behaviour and preferences - over time and for different online communities. The generated ontology has been made available online [43].

To generate the proposed user profile model we have reused and extended existing ontological resources. Following the NeOn methodology [18], we have assessed the generated user profile model by using four competency questions (see Section 2). These questions ensure that, by using the information captured within the proposed user profile model, we can infer the needs, preferences and behaviours of users within particular online communities and time frames. Personality, on the other hand, it is the only aspect of the user that is not considered in the context model, since research has repeatedly shown that personality scores are stable over time [15].

The problem of user modelling and representation has been tackled by different research areas apart from the Semantic Web, including Information Retrieval [5], Recommender Systems [1], Adaptive Hypermedia [4] and Pervasive Computing [3].

Researchers working in these areas has captured demographic features such as gender, age, nationality, etc.), and user context (e.g. social interactions, tasks, platforms, etc.). The representation of these data has evolved from traditional keyword-based representations (i.e. weighted feature vectors and weighted n-grams) to semantically enriched representations such as folksonomies, taxonomies and ontologies. Explicit (e.g. manual editing of user profiles or requesting documents that exemplify the user interests) and implicit (e.g. click-through data, opened documents, and browsing history) learning techniques have been used to capture this information.

Using an ontology and semantics to tackle the problem of user modelling offers a number of advantages: (i) the ontology provides a generic, reusable and machine understandable model for representing the concepts and properties required for describing user activities and measuring their evolution; (ii) due to the reuse of well-known vocabularies, our proposed user profile facilitates the integration of data from multiple social networking platforms; (iii) most importantly, the use of an ontology supports inferring mechanisms that can be used to calculate or derive user behaviour, needs, and preferences.

Future work within the DecarboNet project will advance existing methods to digest and distil information about a user's personal characteristics, opinions, and behaviour, encoded in user-generated content available from dynamic and heterogeneous evidence sources. Users will be able to inspect the user model and gain interactive means explore contextualised information spaces through tailored content services. This integrated and dynamic approach based on data across multiple systems and communities will help to better understand the emergence of collective awareness.

Acknowledgement. The research presented in this paper has been conducted as part of the DecarboNet project (www.decarbonet.eu), Grant Agreement No. 610829.

References

1. Adomavicius, G. and Tuzhilin, A. (2005). Toward the Next Generation of Recommender Systems. IEEE Transactions on Knowledge and Data Engineering, 17: 734-749
2. Breslin, J.G. and Decker, S. (2007). The Future of Social Networks on the Internet: The Need for Semantics, IEEE Internet Computing, 11(6): 86-90
3. Baldauf, M., Dustdar, S. and Rosemberg, F. (2007). A Survey on Context-Aware Systems. International Journal of Ad Hoc and Ubiquitous Computing, 2(4): 263-277.
4. Brusilovsky, P. and Millan, E. (2007). User Models for Adaptive Hypermedia and Adaptive Educational Systems. The Adaptive Web - Methods and Strategies of Web Personalization. Eds. P. Brusilovsky et al. Berlin: Springer. 3-53.
5. Tamine-Lechani, L., Boughanem, M. and Daoud, M. (2010). Evaluation of Contextual Information Retrieval Effectiveness: Overview of Issues and Research. Knowledge and Information Systems, 24(1): 1-34.
6. FOAF | www.foaf-project.org
7. Meaning of a Tag (MOAT) | moat-project.org
8. Relationship Vocabulary | www.vocab.org/relationship
9. SIOC Project | sioc-project.org
10. Schema.org | www.schema.org
11. Microformats | www.microformats.org

12. hcard Microformat | www.microformats.org/wiki/hcard
13. XFN microformat | wwww.gmpg.org/xfn
14. Erétéo, G., et al. Semantic social network analysis: A concrete case. Handbook of Research on Methods and Techniques for Studying Virtual Communities: Paradigms and Phenomena, pages 122–156, 2011.
15. Online Presence Ontology (OPO) | www.online-presence.net/ontology.php
16. DCTerms | www.dublincore.org/documents/dcmi-terms
17. Relationship Vocabulary | www.vocab.org/relationship
18. Suárez-Figueroa, Mari Carmen, et al. Ontology engineering in a networked world. Springer, 2012.
19. Rowe, M. et al. Community analysis through semantic rules and role composition derivation. Journal of Web Semantics: Science, Services and Agents on the World Wide Web 18.1 (2013): 31-47.
20. Hoekstra, Representing social reality in OWL 2, in: 7th International Workshop on OWL: Experiences and Directions (OWLED 2010), 2010
21. MESH user ontology | www.mesh-ip.eu/upload/MESH_user_ontology.zip
22. Cantador, Iván, et al. A multi-purpose ontology-based approach for personalised content filtering and retrieval. Advances in Semantic Media Adaptation and Personalization. Springer Berlin Heidelberg, 2008. 25-51.
23. TextRazor | www.textrazor.com
24. AchemyAPI | www.alchemyapi.com
25. DBpediaSpotlight | spotlight.dbpedia.org
26. Saif, H., Fernandez, M., He, Y. and Alani, H. (2014) SentiCircles for Contextual and Conceptual Semantic Sentiment Analysis of Twitter, 11th ESWC, Crete, Greece
27. Arena, 2012 | www.arena-media.co.uk/blog/2012/05/maslows-hierarchy-of-needs
28. Scholz, T. 2007 | es.slideshare.net/trebor/motivating-people-to-participate, slide 24
29. Antonios, 2010 | www.johnantonios.com/2010/02/06/the-social-media-hierarchy-of-needs
30. Schaefer, C. Motivations and usage patterns on social network sites. (2008).
31. Smock, A.D., et al. Facebook as a toolkit: A uses and gratification approach to unbundling feature use. Computers in Human Behaviour 27, 6 (2011), 2322–2329
32. Quercia, D., Kosinski, M., Stillwell, D., & Crowcroft, J. Our Twitter profiles, our selves: Predicting personality with Twitter. (Socialcom2011).
33. Golbeck, J., et al. Predicting personality from twitter. (SocialCom 2011).
34. Sumner, C. et al. Predicting dark triad personality traits from twitter usage and a linguistic analysis of tweets. In Machine Learning and Applications (ICMLA2012).
35. Chan, J., et al. Decomposing discussion forums and boards using user roles. In ICWSM 2010.
36. Abel, F., et al. Analyzing user modeling on twitter for personalized news recommendations. In User Modeling, Adaption and Personalization (pp. 1-12). Springer Berlin Heidelberg.
37. Szomszor, et al. Semantic modelling of user interests based on cross-folksonomy analysis (pp. 632-648). Springer Berlin Heidelberg.
38. Ankolekar, A. et al. Supporting online problem-solving communities with the semantic web. Proceedings of the 15th international WWW conference. New York, USA.
39. SWUM ontology | www.swum-ontology.org
40. Donohue, B., & Otte, J. N. Fall 2013 Ontological Engineering Prof. Barry Smith The Five-Factor Model and Personality Assessment Ontology
41. Scharl, A., et al. From Web Intelligence to Knowledge Co-Creation-A Platform to Analyze and Support Stakeholder Communication. IEEE Internet Computing 17(5): 21-29
42. Nyea, C.D. et al. Testing the measurement equivalence of personality adjective items across cultures. Journal of Research in Personality, July 2008
43. Upo ontology: http://people.kmi.open.ac.uk/miriam/ontology/upo.zip

Generating Semantic Media Wiki Content from Domain Ontologies

Dominik Filipiak[1,2] and Agnieszka Ławrynowicz[1]

Institute of Computing Science, Poznan University of Technology, Poland
Business Information Systems Institute Ltd., Poland

Abstract. There is a growing interest in holistic ontology engineering approaches that involve multidisciplinary teams consisting of ontology engineers and domain experts. For the latter, who often lack ontology engineering expertise, tools such as Web forms, spreadsheet like templates or semantic wikis have been developed that hide or decrease the complexity of logical axiomatisation in expressive ontology languages. This paper describes a prototype solution for an automatic OWL ontology conversion to articles in Semantic Media Wiki system. Our implemented prototype converts a branch of an ontology rooted at the user defined class into Wiki articles and categories. The final result is defined by a template expressed in the Wiki markup language. We describe tests on two domain ontologies with different characteristics: DMOP and DMRO. The tests show that our solution can be used for fast bootstrapping of Semantic Media Wiki content from OWL files.

1 Introduction

There is a growing interest in holistic ontology engineering approaches [1]. Those approaches use various ontological as well as non-ontological resources (such as thesauri, lexica and relational DBs) [2, 3] . They also involve multidisciplinary teams consisting of ontology engineers as well as non-conversant in ontology construction domain experts. Whilst an active, direct involvement of the domain experts in the construction of quality domain ontologies within the teams appears beneficial, there are barriers to overcome for such involvement to be effective. Those are mostly related to the high complexity of logic-based ontology modeling languages such as OWL[1].

In order to remove the barriers, various tools have been developed that hide or decrease the complexity of logical axiomatisation in expressive ontology languages. Among these tools are Web forms [4], spreadsheet like templates [5] and semantic wikis [6–9]. Recent works have shown that domain experts may be effectively involved in using such tools for knowledge gathering stage of ontology development when the core structure of the ontology is already established [5]. Recent works have also shown that ontology modeling tools based on wikis can contribute to collaboration between ontology engineering experts and domain experts [10].

The aim of this work is to deliver a solution for transformation of OWL files to Semantic Media Wiki (SMW)[2][6] content. By transformation we mean an automatic

[1] http://www.w3.org/TR/owl2-overview/
[2] http://semantic-mediawiki.org

conversion of these files to Wiki articles and category pages. The final result is defined by a template written in the Wiki markup language. The purpose is to bootstrap a collaboration between ontology stakeholders & engineers and a wider community of researchers in constructing a domain ontology once a core structure of the ontology is established.

The rest of this paper is structured as follows. In Section 2 we discuss the work related to ours. In Section 3, we present a solution for transforming OWL files to Semantic Media Wiki content that is based on SPARQL and user defined Wiki templates. In Section 4 we present a simple evaluation of the implemented solution with two ontologies: DMOP and DMRO. Section 5 contains the discussion, and in Section 6 we conclude.

2 Related Work

In [11] an SMW extension consisting of a solution for transformation of ontological knowledge was presented. The focus was on transforming instance data (ABox), where the user could select a subset of instance assertions to import into SMW, and simple schema information. The more expressive ontology model was considered as an external source of knowledge, providing constraints into the domain model stored in the SMW installation. The paper discussed several use cases including coordinating a project team within a company and bootstrapping the contents and vocabulary of the semantic wiki of a conference system.

The Halo extension to SMW, contained in SMW+[3], was developed in order to facilitate the use of Semantic Wikis for large communities of users and thus consisted of a toolset for increasing the ease of use of SMW features. Among the tools, it provided an import and export functionality for OWL ontologies. Currently, the Halo extension is unmaintained.

MoKi [8] is a tool based on SMW that extends SMW by offering specific support for enterprise modelling. It provides support for domain experts in modeling business domains (domain ontologies) and simple processes (process models). MoKi offers a functionality to upload in MoKi an existing domain ontology modeled in OWL. This import functionality generates a MoKi page for each concept, property and individual from the ontology. The templates for these ontology entities are automatically filled based on the axioms modeled in the ontology such as, for example, is-a relation between concepts, domain and range of properties, and individuals being members of concepts.

The author of [12] presents a solution based on OWL Wiki Forms (a Semantic MediaWiki extensions that map Semantic Web ontologies to a Semantic Forms-based semantic wiki) and Fresnel (an ontology for specifying browsing interfaces for Semantic Web data). The solution consists of a mapping from any ontology to Fresnel style data and from Fresnel data to form-based semantic wikis. A technique for automatic generation of Fresnel lenses triples from given ontologies is presented, where the Fresnel lenses triples define a default target interface for data using those ontologies. It is also possible for the user to define custom Fresnel that can cascade over the default interface style, similarly as it is done in CSS.

[3] http://semantic-mediawiki.org/wiki/Semantic_MediaWiki_Plus

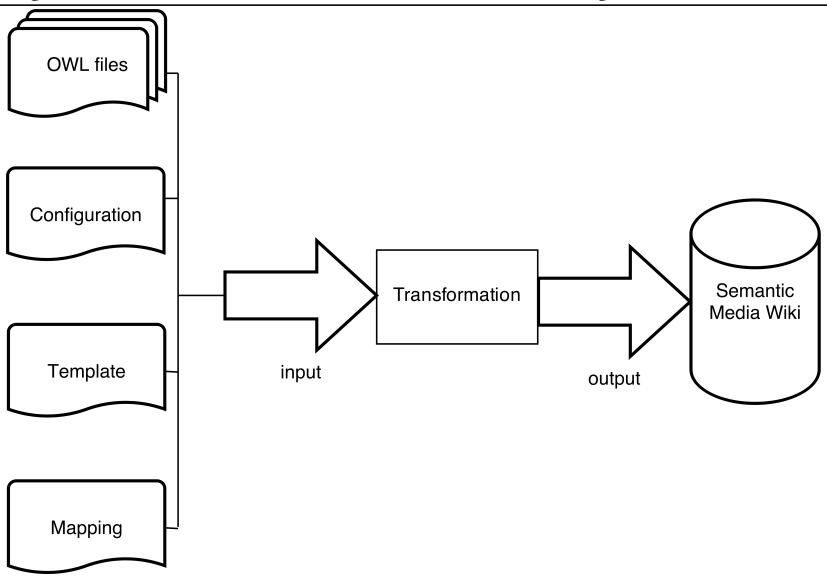


Fig. 1. Input and output of the proposed algorithm.

Our presented solution allows for a direct mapping from an ontology to user defined Wiki templates. In such way, it allows for a direct transformation from arbitrary (user selected) ontology entities to arbitrary Media Wiki form elements (taking into account the distinction between OWL entity types).

3 Approach

3.1 Overview

Figure 1 illustrates the general idea of our approach. The input to our transformation algorithm is a set of at least four files: an OWL file, a configuration file, a template file and a mapping file. There can be one or more OWL files. The configuration file contains information about Semantic Media Wiki adress, login and password. Moreover, it points out to the *root class* - it is an URI which defines which part of an ontology should be processed. The template file, based on Wiki markup syntax, defines how created pages should look like by specifying attributes URIs. Finally, the mapping file connects template and OWL files by assigning variables to entity URIs. Processing these files should result in an updated set of Semantic Media Wiki articles, including article pages, category pages and a template page.

3.2 Design Choices

The design issues, with which we had to deal, concerned, among others, selecting categories and attributes, and representing many attributes for a single entity. By an attribute

we mean a Wiki counterpart of a single OWL property describing a given entity, which is presented mostly in an *infobox* in our templates.

We can describe hierarchy in OWL files as a set of classes, which can be related with each other by child (subclass), parent (superclass) or equivalent class relation. However, Media Wiki articles can be shown as a directed acyclic graph, where each edge represents child/parent relation, according to its direction. Hence, we decided to treat equivalent (with additional constraints) classes as child classes. Entities described as *Named Individuals* are candidates for articles. Similarly, classes with *subClassOf*, *equivalentClass* or *rdf:type* attribute are categories stubs. Due to possibility of numerous attributes, we decided to extract only these which are declared by the user in a template file. Wiki markup syntax does not allow to make implicit declaration of a set of values for one attribute. It cannot predict the number of the values for a given attribute without any additional configuration or modification. Hence, we used Semantic Forms[4] with parser functions as a solution to this problem. The design choices for the transformation are listed in Table 1.

Table 1. Ontology to Semantic MediaWiki transformation choices.

Issue	**Solution**
Article categorization	Either equivalent classes (*owl:equivalentClass*) or parent classes (*rdfs:subClassOf*) are chosen for parent articles of a category
Selection of entities for articles	Resources classified as *owl:NamedIndividual* are chosen for ordinary articles
Selection of entities for category pages	Resources classified as *owl:equivalentClass* or *rdfs:subClassOf* with respect to a base category are chosen for category pages
Selection of properties	Properties for transformation are defined by the user in the template file and in the mapping file
Multiple values for one attribute	Semantic Forms with parser functions are used

3.3 Algorithm

Our approach is described by Algorithm 1. It is necessary to have all mentioned input files in order to define the configuration. To begin the transformation process all classes and individuals which are in a child relation with the root class (including transitivity) have to be found. It can be done by SPARQL queries. We present our solution for classes in Listing 1.1 and for individuals in Listing 1.2. The list of articles stubs is prepared from each acquired entity. Filling all stubs with content from classes and individuals is based on the template and the mapping. In order to do this the ontology has to be searched using SPARQL queries one more time. Article/Category title is based on entity URI. Since there is a requirement for each article to have a title, blank nodes are omitted. The last step consists of transferring the obtained data to Semantic Media Wiki.

[4] `http://www.mediawiki.org/wiki/Extension:Semantic_Forms`

Data: $OWLfiles$, $configuration$, $template$, $mapping$
Result: An updated set of Semantic Media Wiki articles
load $OWLfiles$;
$articles \leftarrow$ process $OWLfiles$;
for $article$ *in* $articles$ **do**
  if $article$ *is a template* **then**
    prepare $template$
  end
  if $article$ *is a category or an article* **then**
    set a title to $template$;
    fill $article$ with data corresponding to $template$ and $mapping$;
    add category footer to $article$;
  end
end
Connect to Semantic Media Wiki;
for $article$ *in* $articles$ **do**
  save $article$ to Semantic Media Wiki;
end

Algorithm 1: OWL files conversion to Semantic Media Wiki articles

Listing 1.1. SPARQL query for finding classes

```
SELECT DISTINCT ?subclass
WHERE {
  ?subclass ((owl:equivalentClass/owl:intersectionOf/rdf:
      ↪ rest*/rdf:first)|rdfs:subClassOf)+ <rootClass>.
}
```

Listing 1.2. SPARQL query for searching for individuals

```
SELECT DISTINCT ?individual
WHERE {
?individual rdf:type ?type.
?type ((owl:equivalentClass/owl:intersectionOf/rdf:rest*/
    ↪ rdf:first)|rdfs:subClassOf|rdf:type)+ <rootClass>.
FILTER (isURI(?individual) && !isBLANK(?individual)).
}
```

4 Evaluation

4.1 Materials

We prepared an application[5] written in Java, which implements the described algorithm. We used Apache Jena[6] to read and query OWL files. Tests of the transformation were performed with two different ontologies - *DMOP* and *DMRO*.

DMOP is an abbreviation for the Data Mining OPtimization Ontology [13, 14]. The ontology is focused on description of numerous data mining algorithms and their charateristics. The primary goal of DMOP is to support making decisions at each step of data mining process which determines the outcome of the process. DMOP is richly axiomatised–it uses almost all features of OWL 2 DL. Majority of DMOP entities are its 'own' entities defined in DMOP's namespaces. Moreover, it imports a part of DOLCE foundational ontology. DMOP has been successfuly used for meta-mining within the Intelligent Discovery Assistant (comprised of an AI planner and semantic meta-miner) that is deployed in the data mining environment RapidMiner.

DMRO, a Digital Multimedia Repositories Ontology [15], has different characteristics than DMOP. It was constructed as a lighweight ontology network using NeoN methodology [2, 3] and various ontology design patterns. The main file imports several ontology modules describing: multimedia resources, users, events, reviews, Web Usage Mining related concepts, and the domain topics. The modules re-use various ontologies and vocabularies such as Dublin Core[7], FOAF[8], RDF Review[9], OBO Relation Ontology[10], and OAI-ORE[11].

4.2 Results

We made simple tests consisting of transforming chosen branches of the ontologies to Semantic Media Wiki.

In case of DMOP, we transformed all classes with related entities, which are in a child relation with *DM-Algorithm* class (examples in Listings 1.3 and 1.4). The sample Wiki article about *C4.5* algorithm (named individual in DMOP) is shown in Figure 2. The article's title results from the URI of the individual, which is shown in Listing 1.3 (*rdf:about* attribute) and marked as A in Figure 2). Attributes are labeled as B, C, D, E in the same figure. Due to lack of information in the ontology files not all of infobox values are filled up (they are marked as B and D on Figure 2). Thanks to Semantic Forms, *has-quality* attribute (marked as F in the figure) has multiple values, what was implicitly declared in the template. Class membership is labeled by G.

[5] `https://github.com/mimol/owl2wiki`
[6] `https://jena.apache.org`
[7] `http://dublincore.org/documents/dcmi-terms/`
[8] `http://xmlns.com/foaf/spec/`
[9] `http://vocab.org/review/terms.html`
[10] `http://obofoundry.org/ro/`
[11] `http://www.openarchives.org/ore/`

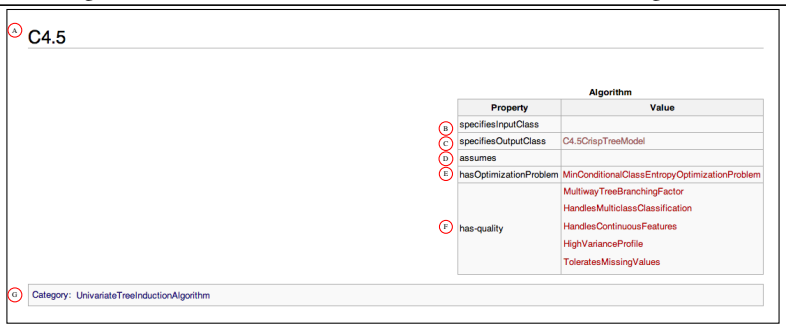


Fig. 2. The result of DMOP ontology transformation

Listing 1.3. Structure of a sample DMOP entity classified as an article

```
<owl:NamedIndividual rdf:about="&PD;C4.5">
 <DOLCE-Lite:has-quality rdf:resource="&DMOP;
     ↪ HighVarianceProfile"/>
 (...)
</owl:NamedIndividual>
```

Listing 1.4. Structure of a sample DMOP entity classified as a category

```
<owl:Class rdf:about="&DMOP;SomeClass">
 <rdfs:subClassOf rdf:resource="&DMOP;SomeSubClass"/>
</owl:Class>
```

We also successfully conducted another experiment with DRMO whose structure differs from DMOP's. Although category entities are similarly declared explicitely as OWL classes (as shown in Listing 1.6), the individuals are not declared as named individuals, but are declared as members of *owl:Thing* (Listing 1.5). Nevertheless, we were able to transform a branch rooted at *DMRO:Event*.

Listing 1.5. Structure of example DMRO entity classified as an article

```
<owl:Thing rdf:about="#Event2305">
 <rdf:type rdf:resource="&DMRO-Event;EventSection"/>
 (...)
</owl:Thing>
```

Listing 1.6. Structure of example DMRO entity classified as a category

```
<owl:Class rdf:about="&dul;Event">
 <rdfs:subClassOf rdf:resource="&dul;Entity"/>
</owl:Class>
```

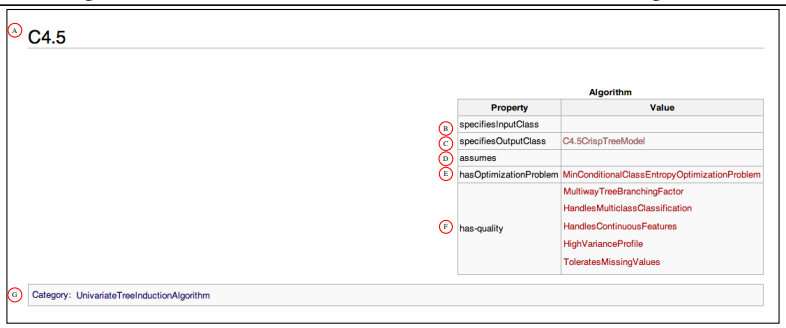


Fig. 2. The result of DMOP ontology transformation

Listing 1.3. Structure of a sample DMOP entity classified as an article

```
<owl:NamedIndividual rdf:about="&PD;C4.5">
 <DOLCE-Lite:has-quality rdf:resource="&DMOP;
     ↪ HighVarianceProfile"/>
 (...)
</owl:NamedIndividual>
```

Listing 1.4. Structure of a sample DMOP entity classified as a category

```
<owl:Class rdf:about="&DMOP;SomeClass">
 <rdfs:subClassOf rdf:resource="&DMOP;SomeSubClass"/>
</owl:Class>
```

We also successfully conducted another experiment with DRMO whose structure differs from DMOP's. Although category entities are similarly declared explicitely as OWL classes (as shown in Listing 1.6), the individuals are not declared as named individuals, but are declared as members of *owl:Thing* (Listing 1.5). Nevertheless, we were able to transform a branch rooted at *DMRO:Event*.

Listing 1.5. Structure of example DMRO entity classified as an article

```
<owl:Thing rdf:about="#Event2305">
 <rdf:type rdf:resource="&DMRO-Event;EventSection"/>
 (...)
</owl:Thing>
```

Listing 1.6. Structure of example DMRO entity classified as a category

```
<owl:Class rdf:about="&dul;Event">
 <rdfs:subClassOf rdf:resource="&dul;Entity"/>
</owl:Class>
```

5 Discussion

We tested our solution with ontologies having different characteristics. Since SPARQL engines by default are not supposed to perform reasoning, SPARQL may turn very structure-sensitive. Although, in the OWL serialization that we used the structure of category entity is based on *owl:Class* in both, DMOP and in DMRO (Listings 1.4 and 1.3, respectively), there are differences in individual selection. In DMOP, individuals are explicitly declared as a *owl:NamedIndividual*, while in DMRO they are not. In the latter case, the entities we classify as individuals are instances of *owl:Thing*. That is why it is important to consider possible cases and take them into account in SPARQL queries or transform OWL files to a canonical representation before SPARQL is applied to query them.
Alternatively, as a more standard solution, we could use an API for handling ontologies like OWL API or Jena ontology API. However, while designing our SPARQL-based solution we kept in mind that it can be further flexibly extended to transform remote (linked) data from SPARQL endpoints to Semantic Media Wiki content.

Our main motivation is the real need for Wiki based tools in the context of such portals as *DMO Foundry* (`http://www.dmo-foundry.org`) or *OpenML* (`http://openml.org`). Using the presented in this paper preliminary solution we have generated content that is a human-readable, structured and organised knowledge base. We envisage that it could be used by such domain users as researchers trying to find out which algorithm would match their expectations or even students during classes.

6 Conclusions

In this paper, we have presented a prototype solution for transformation of OWL ontologies to Semantic Media Wiki content. The solution is based on a mapping between selected ontology entities and user-defined Wiki templates, and on using SPARQL. We have successfuly applied our implemented prototype to two different ontologies: DMOP and DMRO.

Despite of describing a working prototype, we still consider this research as a work in progress. In future work, we plan to consider ideas for cascading templates (similarly to the work of [12]) and text mining (especially named entity recognition). We also plan to extend our solution by support for exporting knowledge from the Wiki to OWL ontologies. Finally, our plans are to use the solution in real world use cases like within data mining portals such as DMO Foundry or OpenML to provide Wiki based tools for community of researchers in data mining or in other disciplines. We plan to test the prototype in such settings, where the collaboration between ontology engineers and normal users will be investigated. We plan to conduct a case study to investigate how the collaboration could be improved based on our technical solution.

References

1. Denaux, R., Dolbear, C., Hart, G., Dimitrova, V., Cohn, A.G.: Supporting domain experts to construct conceptual ontologies: A holistic approach. Web Semantics: Science, Services and Agents on the World Wide Web **9**(2) (2011)

2. Gómez-Pérez, A., Suárez-Figueroa, M.: Scenarios for building ontology networks within the NeOn methodology. In: K-CAP. (2009) 183–184
3. Suárez-Figueroa, M.: NeOn Methodology for Building Ontology Networks: Specification, Scheduling and Reuse. PhD thesis, Universidad Politécnica de Madrid, Spain (2010)
4. Tudorache, T., Nyulas, C., Noy, N.F., Musen, M.A.: WebProtege: A collaborative ontology editor and knowledge acquisition tool for the web. Semant. web **4**(1) (January 2013) 89–99
5. Jupp, S., Horridge, M., Iannone, L., Klein, J., Owen, S., Schanstra, J., Wolstencroft, K., Stevens, R.: Populous: a tool for building OWL ontologies from templates. BMC Bioinformatics **13**(S-1) (2012) S5
6. Krötzsch, M., Vrandečić, D., Voelkel, M.: Semantic MediaWiki. In Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L., eds.: The Semantic Web - ISWC 2006. Volume 4273 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2006) 935–942
7. Auer, S., Dietzold, S., Riechert, T.: OntoWiki – a tool for social, semantic collaboration. In: Proceedings of the 5th International Conference on The Semantic Web. ISWC'06, Berlin, Heidelberg, Springer-Verlag (2006) 736–749
8. Ghidini, C., Kump, B., Lindstaedt, S., Mahbub, N., Pammer, V., Rospocher, M., Serafini, L.: MoKi: The enterprise modelling wiki. In Aroyo, L., Traverso, P., Ciravegna, F., Cimiano, P., Heath, T., Hyvnen, E., Mizoguchi, R., Oren, E., Sabou, M., Simperl, E., eds.: The Semantic Web: Research and Applications. Volume 5554 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2009) 831–835
9. Nalepa, G.: Loki – semantic wiki with logical knowledge representation. In Nguyen, N., ed.: Transactions on Computational Collective Intelligence III. Volume 6560 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2011) 96–114
10. Di Francescomarino, C., Ghidini, C., Rospocher, M.: Evaluating wiki collaborative features in ontology authoring. IEEE Transactions on Knowledge and Data Engineering **(to appear)** (2014)
11. Vrandečić, D., Krötzsch, M.: Reusing ontological background knowledge in semantic wikis. In Völkel, M., Schaffert, S., Decker, S., eds.: 1st Workshop on Semantic Wikis. Number 206 in CEUR Workshop Proceedings, Aachen (2006)
12. Rutledge, L.: From ontology to wiki generating cascadable default fresnel style from given ontologies for creating semantic wiki interfaces. In: Workshop on Semantic Web Collaborative Spaces (SWCS2013). (2013)
13. Hilario, M., Nguyen, P., Do, H., Woznica, A., Kalousis, A.: Ontology-based meta-mining of knowledge discovery workflows. In: Meta-Learning in Computational Intelligence. Volume 358 of Studies in Computational Intelligence. Springer (2011) 273–315
14. Keet, C.M., Lawrynowicz, A., d'Amato, C., Hilario, M.: Modeling issues, choices in the Data Mining OPtimization Ontology. In Rodriguez-Muro, M., Jupp, S., Srinivas, K., eds.: OWLED. Volume 1080 of CEUR Workshop Proceedings., CEUR-WS.org (2013)
15. Lawrynowicz, A., Palma, R.: Applications of ontology design patterns in the transformation of multimedia repositories. In Blomqvist, E., Gangemi, A., Hammar, K., Suárez-Figueroa, M.C., eds.: WOP. Volume 929 of CEUR Workshop Proceedings., CEUR-WS.org (2012)

SPARQL Query Result Explanation for Linked Data

Rakebul Hasan[1], Kemele M. Endris[1,2], and Fabien Gandon[1]

[1] INRIA Sophia Antipolis, Wimmics, France
[2] DISI, University of Trento, Italy
hasan.rakebul@inria.fr,keme686@gmail.com,fabien.gandon@inria.fr

Abstract. In this paper, we present an approach to explain SPARQL query results for Linked Data using *why-provenance*. We present a non-annotation-based algorithm to generate *why-provenance* and show its feasibility for Linked Data. We present an explanation-aware federated query processor prototype and show the presentation of our explanations. We present a user study to evaluate the impacts of our explanations. Our study shows that our query result explanations are helpful for end users to understand the result derivations and make trust judgments on the results.

1 Introduction

As a result of the W3C Linked Open Data Initiative, recently we have seen a rapid growth in publishing data sets on the Semantic Web, in form of RDF data with SPARQL query endpoints. This enables developers to query and integrate disparate Semantic Web data. As argued in [14, 16], it is essential to provide additional explanations about which source data were used in providing results, how the source data were combined, to enable users understand the result derivations, and validate or invalidate the results.

Within the Semantic Web community, explanations have been studied for Semantic Web applications and OWL entailments. Explanation for SPARQL query results has not been independently studied by the community. However, there have been several works on tracing the origin of query results – e.g. *why-provenance*. These attempts are based on the annotation approach (the eager approach) where the underlying data model, the query language, and the query processing engine are re-engineered to compute provenance during the query processing. This is undesirable for the Linked Data scenario as re-engineering the underlying data model, the query language, or the query processor is often not possible from the querying side. Furthermore, previous work on explanations for the Semantic Web does not study how explanations impact the end-users.

To address these problems, we provide SPARQL query result explanations. The main component in an explanation for a query result tuple is its *why-provenance*. We propose a non-annotation approach to generate *why-provenance* for SPARQL query results. We present an explanation-aware federated query

processor prototype to show the presentation of our explanations. Finally, we present a user study which evaluates the impacts of SPARQL query result explanations on the end-users.

The structure of the rest of this paper is as follows: in section 2, we present the related work. In section 3, we discuss SPARQL query result explanations, introduce the concept of *why-provenance*, and present our algorithm to generate *why-provenance*. In section 5, we present our explanation-aware federated query processor prototype. In section 6, we present a user study to evaluate the impacts of explanations. Finally, we conclude and discuss the future work in section 7.

2 Related Work

Previous work on explanation in the Semantic Web literature [7] addresses the problems of representing explanation metadata [13], and generating explanations for Semantic Web applications [10] and entailments [8]. SPARQL query result explanation has not be studied in the previous work. Query result provenance has been studied in the database community [2] and the Semantic Web community. The previous works on provenance for SPARQL query results are based on transforming the RDF data model and SPARQL query language to relational data model and relational database query language respectively [14, 4], or generation of provenance metadata during the query processing [16, 3]. However, in the Linked Data scenario, we do not have any control over the underlying data model or the query processor. Therefore, re-engineering the underlying data model or query processor is often not possible in the Linked Data scenario. Furthermore, the impacts of explanations on end-users has not been studied in the previous work on explanation in the Semantic Web literature. In the other fields, Lim *et al.* [9] studied the impacts of explanations on end-users for context-aware applications. Tintarev and Masthoff [15] studied the effectiveness of explanations for recommender systems.

3 Explaining SPARQL Query Results

We provide SPARQL query result provenance as query result explanations. More precisely, for a SPARQL query result tuple, we provide its *why-provenance* as its explanation. Buneman *et al.* [1] first introduced the notion of *why-provenance* for relational databases. *Why-provenance* captures all the different witnesses for a tuple in the query result. For a query Q and output tuple t, a *witness* is the sufficient subset of the database records which ensures that the tuple t is in the output. Each witness is a derivation for the output tuple. Theoharis *et al.* [14] later adapted *why-provenance* for RDF and SPARQL. Similar to the relational setting, *why-provenance* for RDF and SPARQL captures all the different derivations of a tuple in the query result. To illustrate, we use a simple example, containing RDF data about professors and the courses they teach, shown in Figure 1. We use identifiers for each triple for presentation purpose in this paper. Consider the SPARQL query *Q1* shown in Listing 1.1, which asks for all the

	Triples about professors
t1	:ProfA :dept :CS
t2	:ProfA :name "Prof. A"
t3	:ProfA :email "a@email.edu"
t4	:ProfA :course :CS101
t5	:ProfA :course :CS103
t6	:ProfA :course :CS201
t7	:ProfA :course :CS204
t8	:ProfB :dept :MATH
t9	:ProfB :name "Prof. B"
t10	:ProfB :email "b@email.edu"
t11	:ProfB :course :MATH101
t12	:ProfB :course :MATH201

	Triples about courses
t13	:CS101 :courseType :underGrad
t14	:CS103 :courseType :underGrad
t15	:MATH101 :courseType :underGrad
t16	:CS201 :courseType :grad
t17	:CS204 :courseType :grad
t18	:MATH201 :courseType :grad

Fig. 1. Example RDF triples.

professors who teach undergraduate level courses and their corresponding email addresses. The first triple pattern *?course :courseType :underGrad* in the query *Q1* selects the undergraduate level courses.

Listing 1.1. SPARQL query Q1

```
SELECT DISTINCT ?name ?email
WHERE
  { ?course :courseType :underGrad .
    ?prof :course ?course .
    ?prof :email ?email .
    ?prof :name ?name
  }
```

Result of Q1:

?name	?email
Prof. A	a@email.edu
Prof. B	b@email.edu

The second triple pattern *?prof :course ?course* selects the professors for those undergraduate level courses. The next two triple patterns *?prof :email ?email* and *?prof :name ?name* selects the email addresses and names of the corresponding professors matched by the two previous triple patterns. The result of the query *Q1* (under set semantics) executed on the RDF data containing the triples in Figure 1 is shown on the right in Listing 1.1. The *why-provenance* for the result tuple (Prof. A, a@email.edu) is {{t14, t5, t2, t3}, {t13, t4, t2, t3}}. Each inner set in *why-provenance* represents a derivation involving the triples in the inner set. This means that the result tuple (Prof. A, a@email.edu) can be derived in two different ways according to *Q1*. The first one by using the triples t14, t5, t2, and t3. The second one by using the triples t13, t4, t2, and t3. The *why-provenance* for the result tuple (Prof. B, b@email.edu) on the other hand has one derivation: {{t15, t11, t10, t9}}. Please not that we are using the triple identifiers only for presentation purpose. The original data model containing the triples shown in Figure 1 is not changed – *i.e.* we do not annotate the RDF triples. We use the RDF triples as they are in the original data source.

3.1 Algorithm for Generating Why-Provenance

In this section, we present our non-annotation approach to generate *why-provenance* for SPARQL query results. We currently do not support SPARQL queries with

sub-queries, FILTER (NOT) EXISTS, MINUS, property paths, and aggregates. The *GenerateWhyProvenace* procedure shown in Algorithm 1 generates *why-provenance* for an RDF model M, a SPARQL query Q, and a result tuple t. The RDF model M can be an RDF dataset or a SPARQL endpoint on which the SPARQL query Q is solved and the result tuple t is produced. At line 2

Algorithm 1 Why-provenance algorithm.

```
1: procedure GenerateWhyProvenace(M,Q,t)
2:     Q' ← ProvenanceQuery(Q, t)
3:     I ← Q'(M)
4:     E ← AlgebricExpression(Q)
5:     W ← DerivationsFromQuery(M, E, I)
6:     return W
7: end procedure
```

of Algorithm 1, we first re-write the original query to a provenance query by adding the tuple t as a solution binding using the SPARQL 1.1 VALUES construct, and projecting all the variables. The result set of the provenance query provides us all the variable bindings on the RDF data for the solution tuple t. Each tuple (row) in the result set of the provenance query represent a derivation for the solution tuple t. The main idea behind our algorithm is to extract *why-*

Algorithm 2 Procedure for creating the provenance query.

```
1: procedure ProvenanceQuery(Q,t)
2:     Q' ← AddValueBindings(Q', t)
3:     Q'' ← ProjectAllVariables(Q')
4:     return Q''
5: end procedure
```

provenance triples from the triple patterns in the original query by replacing the variables in the triple patterns by the corresponding values from each tuple (row) of result of the provenance query. At line 3 of Algorithm 1, we execute the re-written query. At line 4, we convert the original SPARQL query Q to SPARQL algebraic expression for ease of query parsing and manipulation. At line 5, the *DerivationsFromQuery* procedure extracts the derivations. Algorithm 2 shows the *ProvenanceQuery* procedure to re-write the original query to a provenance query. Line 2 adds the result tuple t as a solution binding using the SPARQL 1.1 VALUES construct. Line 3 modifies the query to projects all the variables in the query.

Algorithm 3 shows the *DerivationsFromQuery* procedure to extract the derivations given the RDF model M, the SPARQL algebraic expression E, and the provenance query results I. Lines 3–20 iterate through all the tuples of I, extracts provenance triples corresponding to each tuple, and stores them in a set of a sets D. We assume that basic a graph pattern in a SPARQL query is not repeated. We use a hash table, BP, to flag which basic graph pattern (BGP) is examined for a tuple in I to extract provenance triples. Lines 4–6 initialize the hash table by setting *False* for each BPG, meaning none of the basic graph

Algorithm 3 Procedure for extracting derivations from a query.

```
 1: procedure DerivationsFromQuery(M,E,I)
 2:     D ← ∅
 3:     for each tuple in I do
 4:         for each bgp in E do
 5:             BP[bgp] ← False
 6:         end for
 7:         T ← ∅
 8:         if hasUnion(E) or hasJoin(E) or hasLeftJoin(E) then
 9:             for each operator in E do
10:                 T1 ← TriplesForOperator(M, operator, tuple, BP)
11:                 if T1 ≠ ∅ then
12:                     T ← T ∪ T1
13:                 end if
14:             end for
15:         else
16:             bgp ← GetTheBGP(E)
17:             T ← TriplesFromBGP(M, bgp, tuple, BP)
18:         end if
19:         D ← D ∪ {T}
20:     end for
21:     return D
22: end procedure
```

patterns is examined for the current tuple in I at this point. If a query has just one BGP, we extract the provenance triples from that BGP (lines 15–18) for a tuple in I and store the provenance triples in set T. If a query has more than one BGP, *i.e.* if the algebraic expression has the union or the join or the left-join operator, we extract the provenance triples from the operand BGPs of each of the operators and store the provenance triples in set T (lines 7–14) for a tuple in I. We only extract provenance triples for a BGP once at this stage – using the hash table BP as flags for BGPs to keep trace of which BGP has been used so far to extract provenance triples. Finally line 19 does a union of the triples extracted for a tuple in I, stored in set T, as an element (shown by braces around T at line 19) with the set of sets D and assigns the result of the union to D. When we go out of the loop started at line 3, D contains all the derivations we extracted. We return the set of sets D at line 21. Each element in D is a set representing a derivation for the result tuple. Algorithm 4 shows the *TriplesForOperator* procedure which extracts provenance triples from the operands of an operator. Lines 3–4 get the left and the right BGPs for the operator *Op*. As we are only restricted to SPARQL queries without sub-queries, the operands are always BGPs. Lines 5–7 extract provenance triples from the left BGP L if provenance triples have not been extracted from L yet, and assigns them to the set P. Lines 8–11 extract provenance triples from the right BGP R, stored in the set T, if provenance triples have not been extracted from R yet, and assigns the union of P and T to P. At line 12, we return the set P which

Algorithm 4 Procedure for extracting triples from operands of an operator.

```
1: procedure TriplesForOperator(M,Op,Tup,BP)
2:     P ← ∅
3:     L ← GetLeftBGP(Op)
4:     R ← GetRightBGP(Op)
5:     if BP[L] = False then
6:         P ← TriplesFromBGP(M, L, Tup, BP)
7:     end if
8:     if BP[R] = False then
9:         T ← TriplesFromBGP(M, R, Tup, BP)
10:        P ← P ∪ T
11:    end if
12:    return P
13: end procedure
```

contains all the provenance triples extracted from the left and the right BGPs of the operator *Op*. The *TriplesFromBGP* procedure calls at line 6 and line 8 check if all the triples extracted from the BGPs exist in the RDF model M by sending SPARQL ASK queries with each extracted triples. This means that a BGP which was an operand of a SPARQL UNION or OPTIONAL operator would contribute to the provenance triples only if it matches against the RDF model M. Algorithm 5 shows the *TriplesFromBGP* procedure which does this. Lines

Algorithm 5 Procedure for extracting triples from a basic graph patter.

```
1: procedure TriplesFromBGP(M,BGP,Tup,BP)
2:     T ← ∅
3:     for each triplePattern in BGP do
4:         triple ← ReplaceVariablesByValues(triplePattern, Tup)
5:         if Ask(M, triple) = True then
6:             T ← T ∪ triple
7:         else
8:             BP[BGP] ← True
9:             return ∅
10:        end if
11:    end for
12:    BP[BGP] ← True
13:    return T
14: end procedure
```

3–11 iterate through the triple patterns in the BGP and extracts the triples. At line 4 we replace the variables of a triple pattern by the corresponding values in the tuple *Tup*, where *Tup* is a tuple from the result of the re-written provenance query. Lines 5–6 first check if the extracted triple is valid by sending an ASK query with this triple to the RDF model M, then if it's a valid triple we take the

triple and store it in the set T. If the triple is not valid (does not exist in M), we set the flag for the BGP to true and return an empty set (lines 7–9). At line 10, we go out of the loop started at line 3, and set the flag for the BGP to true. Finally at line 11 we return the set of extracted provenance triples.

4 Performance Evaluation of the Algorithm

We implement our algorithm using Jena-ARQ API[3]. We evaluate our algorithm using the DBPSB benchmark [11] queries on a Jena-TDB (version 1.0.0) triple store [12]. DBPSB includes 25 query templates which cover most commonly used SPARQL query features in the queries sent to DBpedia[4]. We generate our benchmark queries from these query templates. We allow Jena-TDB to use 16 GB of memory. We execute all the queries in a commodity server machine with a 4 core Intel Xeon 2.53 GHz CPU, 48 GB system RAM, and Linux 2.6.32 operating system. As the RDF dataset, we use the DBpedia 3.5.1 dataset with 100% scaling factor – provided by the DBPSB benchmark framework. To generate benchmark queries, we assign randomly selected RDF terms from the RDF dataset to the placeholders in the DBPSB query templates. We generate 1 query for each template resulting total 25 queries. Before executing the queries, we restart the triple store to clear the caches. Then we execute the 25 queries and along with the *why-provenance* algorithm for all the result tuples once in our warm-up phase. Then we execute each query and the *why-provenance* algorithm for all the result tuples of each query 5 times. We report the average execution time and average provenance generation time for all result tuples (PGT) for each query, both in milliseconds (ms). We specify a 300 second timeout for a query execution. Queries belonging to templates 2, 16, 20, and 21 did not finish executing within the 300 seconds time limit, and hence we do not report them.

4.1 Query Execution and Provenance Generation

Table 1 shows the number for results (#RES), query executing time (QET), provenance generation time for all result tuples (PGT), and provenance generation time per result tuple (PGTPR) for DBPSB queries. PGTs for queries with long execution times and large number of results (queries 6, 8, 10, 14, 22, 24, and 25) are very high. This is not surprising because for each result tuple of a query, we execute the original query with the result tuple as a variable-value binding. Database literature already discusses this issue [2]. Generally speaking, non-annotation approaches compute provenance only when it is needed, by examining the source data and the output data. This requires sophisticated computations involving the source data and the output data. This means each individual tuple in the output data has to be examined separately to compute their provenance, and hence time required for generating provenance for all the

[3] http://jena.apache.org/
[4] http://dbpedia.org

graph using Jena-ARQ. We borrow the idea of CONSTRUCT sub-queries from Corese-DQP [5]. We also implement the common federated query processing concepts of exclusive triple pattern groups and bound join proposed in [**?**].

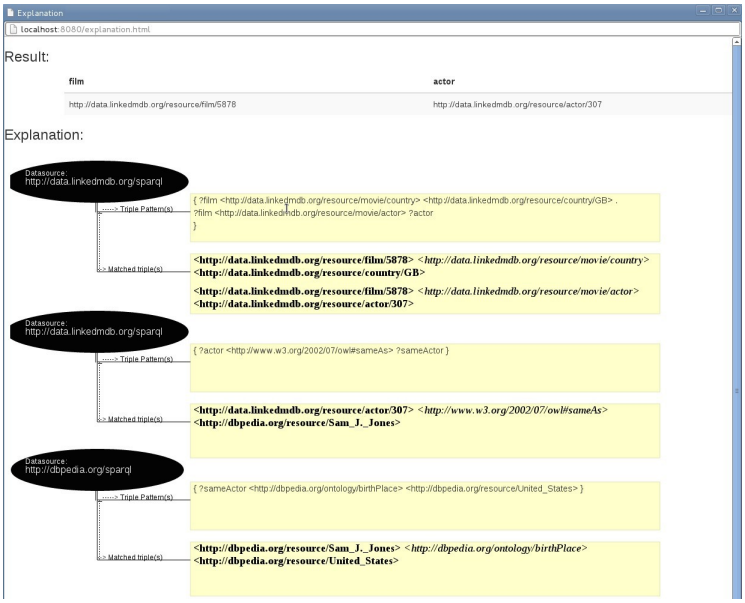


Fig. 2. Example of a query result explanation.

We provide a user interface to enable users to configure SPARQL endpoints as data sources, and submit queries. Furthermore, users can ask for explanation for each query result tuple from the user interface. We provide three types of information in an explanation. We show the *why-provenance* triples, which data source each triple in the *why-provenance* comes from, and which triple pattern of the original query each triple in the *why-provenance* matches. Figure 2 shows an example of a query result explanation. We generate the *why-provenance* triples using the algorithm we presented in section 3.1 on the local virtual RDF graph. We keep two additional indexes in the federated query processor to keep trace of which data source each triple comes from, and which triple pattern each triple matches. These two indexes allow us to provide the information on data sources and matched triple patterns in the explanations.

6 Evaluation of the Impacts of Explanations

We conducted a user study to investigate the impact of query result explanations. Our study is similar to the user study conducted by Lim *et al.* [9] to examine effectiveness of different types of explanations for context-aware intelligent systems. The questionnaire for our study consists of three sections: learning section, reasoning section, and survey section. Furthermore, we have two cases:

with explanation and without explanation. A participant is randomly assigned to the case of "with explanation" or "without explanation".

In the learning section, participants were given a high-level overview of our query processor and an example SPARQL query with a result tuple to help them learn how the federated query processor works. Participants for the "with explanation" case additionally received the explanation of the result tuple for the example query (as shown in Figure 2). In the reasoning section, participants were given the same SPARQL query as in the learning section, but a different result tuple along with the some triples contained in two data sources (DBpedia[5] and LinkedMDB[6]). Then we first ask the participants to select the relevant data sources for each triple pattern in the query. Next, we ask the participants to select the source triples (*why-provenance* triples) from the two data sources which contributed to the result tuple. Then we ask the participants to rate their confidence on their answer choices for the data source selection and the source triple selection questions. The choices for confidence rating were very low, low, medium, high, and very high. The questions in the reasoning section help us analyze how the users understand the result derivation process and if the explanation provided in the learning section have any impact on their understanding. In the survey section of our study, we ask the participants if explanations help users to understand the result derivation and to make trust judgments on the results. Furthermore, we ask them which types of information they think are helpful in an explanation for understanding and making trust judgments. The questions in the survey section help us understand how the participants feel about the system and its explanation features.

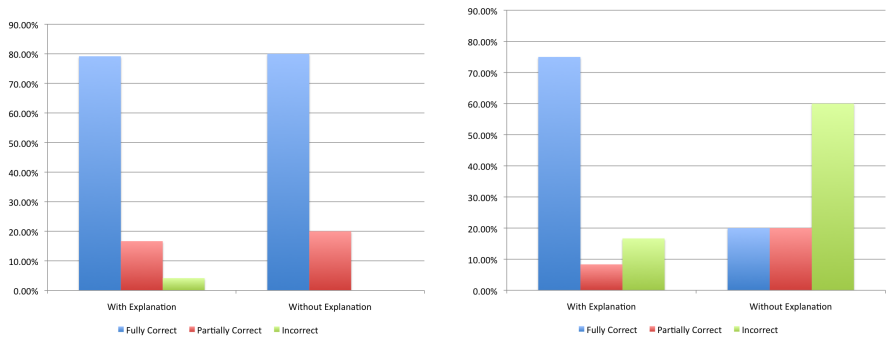
The query we used is a query to find the British movies with American actors. The result tuple includes URIs for a film and an actor. Part of the query is solved in LinkedMDB (finding the British movies) and part of it is solved in DBpedia (finding birth places of the actors). In the query result tuple, we intentionally do not provide natural language descriptions. Instead we provide URIs from LinkedMDB – which are numeric resource URIs – for the actor and the film. This is to make sure that participants are not using their background knowledge about movies and actors in their answers. For the data source selection and source triple selection questions, we provide small subsets of DBpedia triples (11 triples) and LinkedMDB triples (13 triples). We used Google Forms[7] for the questionnaires and Google App Engine[8] to randomize the selection of two cases – "with explanation" or "without explanation". We invited the member of our laboratory[9] (via our mailing list), the members of Semantic Web Interest Group[10] (via their mailing list), and the followers of Twitter hashtags #SemanticWeb, #RDF, and #SPARQL. 11 participants took part in the study. There

[5] `http://dbpedia.org/`
[6] `http://linkedmdb.org/`
[7] `http://www.google.com/google-d-s/createforms.html`
[8] `https://appengine.google.com/`
[9] `http://wimmics.inria.fr/`, `https://glc.i3s.unice.fr/`
[10] `http://www.w3.org/2001/sw/interest/`

were 6 participants for the "with explanation" case and 5 participants for the "without explanation" case. There were 8 male participants and 3 female participants. The ages of the participants range from 22 to 65. All the participants had knowledge of RDF and SPARQL. The questionnaire and the responses of the participants are available online[11].

6.1 Results of the Study

We analyze the ability of the participants to apply their understanding of the system by computing the number of fully correct, partially correct, and incorrect answers for the data source selection and the source triple selection questions in the reasoning section. If a participant selects all the correct choices for an answer, we consider it as fully correct. If a participant selects all the correct choices but also selects some extraneous choices, we consider the answer as partially correct. If a participant's choices for an answer do not contain all the correct choices, we consider it as incorrect. In addition, if a participant selected all choices given for the source triple selection question, we consider the answer as incorrect to avoid guessing. For the data source selection question, we had 4 questions for 4 triple patterns in the query. We count the number of participants who provided fully correct answers, partially correct answers, and incorrect answers for each of these 4 questions. Then we take the average of the counts for the fully correct answers, the average of the counts for the partially correct answers, and the average of the counts for the incorrect answers. These averages represent the average number of participants into the three answer categories categories – fully correct, partially correct, and incorrect – for the data source selection question as a whole. We compute these averages separately for both the "with explanation" and "without explanation" cases and compute the percentages of participants in the three answer categories for the two cases from these average. Figure 3(a) shows the percentage of participants with fully correct, partially



(a) Data source selection (b) Source triple selection

Fig. 3. Participants' response about data source selection and source triple selection.

[11] `http://ns.inria.fr/ratio4ta/sqe/`

correct, and incorrect answers when the explanation is given and when the explanation is not given for the data source selection question. The results are very similar for both "with explanation" and "without explanation" cases. Majority of the participants understood how data source selection works for our federated query processor system when the explanation was given ((79.17%) and also when the explanation was not given (80.0%). Therefore the impact of explanations for source selection understanding is not clear from our study. For the source triple selection question, we had two questions for the two data sources we used. We compute the percentages of participants in the fully correct, partially correct, and incorrect answer categories for the "with explanation" and "without explanation" cases using the same method as the data source selection question. Figure 3(b) shows the percentage of participants with fully correct, partially correct, and incorrect answers when the explanation is given and when the explanation is not given for the source triple selection question. More participants provided correct answers when the explanation was give (75% for "with explanation", 20% for "without explanation"). Furthermore, more participants provide incorrect answers when the explanation was not given (16.67% for "with explanation", 60% for "without explanation"). This clearly shows that participants who were given explanations understood better which triples contributed to the result from the two data sources. The final question in the reasoning section asks participants to rate their confidence level about the answers for the data source selection question and the source triple selection question. Figure 4 shows the confidence level of the participants about their answers. 50.0% of participants with explanation rate their confidence as very high whereas none of participants without explanation rate very high. 33.33% of participants with explanation rate their confidence as high whereas 80% of participants without explanation rate high. This shows that participants with explanation are more confident on their answers – as many of them answered "very high" or "high".

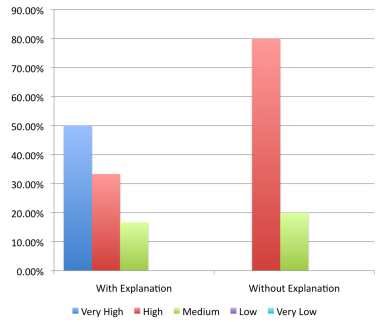
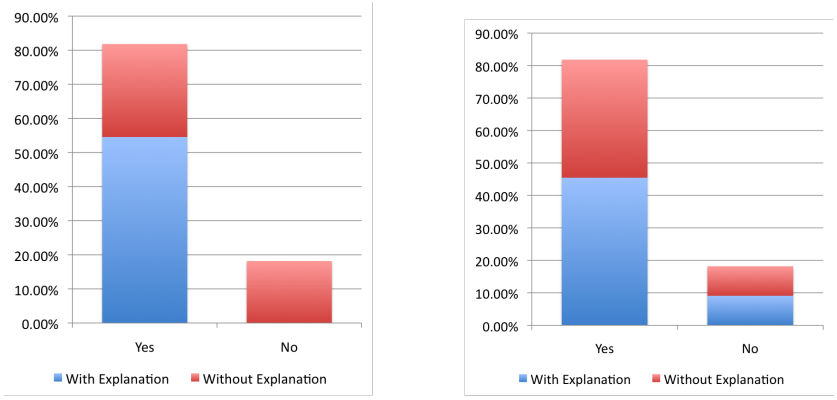


Fig. 4. Participants' confidence level about their answers.

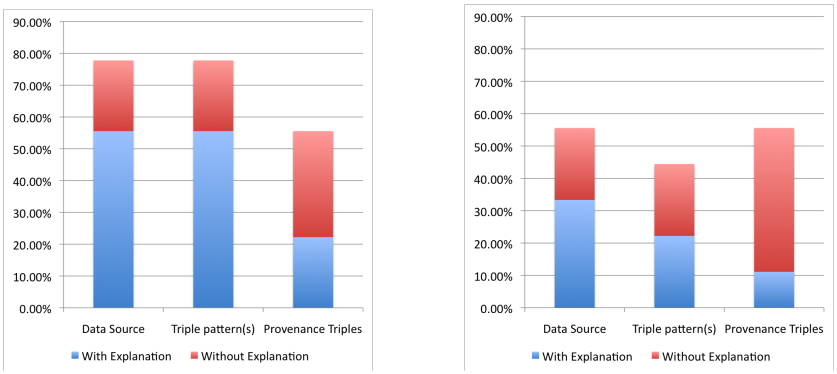
For the survey section, we ask the participants if explanations are helpful to understand the query result derivation, and if explanations are helpful to make trust judgments on the query result. If a participant answered "yes", he/she was also asked what kind of information he/she found helpful. Figure 5(a) shows the percentage of participants who answered explanations are helpful or unhelpful for understanding the query result derivation. Majority of the partic-



(a) Understanding (b) Making trust judgments

Fig. 5. Percentage of participants who answered explanations are helpful or unhelpful.

ipants (81.81) responded that explanations are helpful for understanding the query result derivation. Only 18.18% of the participants answered that explanations are unhelpful for understanding the query result derivation – none of these participants were given explanations. Figure 5(b) shows the percentage of participants who answered explanations are helpful or unhelpful to make trust judgments on the query result. Again, Majority of the participants (total 81.81%) responded that explanations are helpful to make trust judgments on the query result. Only 18.18% of the participants answered that explanations are unhelpful to make trust judgments on the query result. This shows that majority of the survey participants feel that explanations are helpful to understand query result derivations and to make trust judgments on query results. Figure 6(a)



(a) Understanding (b) Making trust judgments

Fig. 6. Participants who found different types of information in the explanation helpful.

shows the participants who found information on data source, triple pattern(s), and *why-provenance* triples helpful for understanding the query result derivation. Please note that only the answers from participants who answered "yes"

shown in Figure 5(a) are considered. Out of 9 participants who answered "yes", 77.78% responded that the data source related information was helpful, 77.78% responded that the triple pattern(s) related information was helpful, and 55.55% responded that the provenance triple related information was helpful. However, our analysis on source selection question responses (Figure 3(b)) shows that the explanation helped participants significantly improve their correctness on selecting the provenance triples. Therefore, it is hard to explain why only 22.22% with explanation responded that the provenance triple related information was helpful. One possible reason could be that when they were not given the explanation, they felt the need for explanation with provenance triple (hence 33.33% for without explanation). But when they were given the explanation, they were not aware that the provenance triple related information helped them to have a better understanding. Figure 6(b) shows the participants who found information on data source, triple pattern(s), and *why-provenance* triples helpful to make trust judgments. Again only the answers from participants who answered "yes" shown in Figure 5(b) are considered. Out of 9 participants who answered "yes", 55.55% responded that the data source related information was helpful, 44.44% responded that the triple pattern(s) related information was helpful, and 55.55% responded that the provenance triple related information was helpful. Again, it is interesting to notice that participants who were not given the explanation felt the need for provenance triples related information. This analysis shown in Figure 6 shows that participants found data source and triple pattern(s) related information helpful for understanding the query result derivation, but have less stronger feeling about provenance triples related information for understanding query result derivations. For making trust judgments, participants do not have as strong opinions, but majority of them feel that data source and provenance triple related information are helpful.

7 Conclusion and Future Work

In this paper, we present an approach to explain SPARQL query results for Linked Data. We present a non-annotation approach to generate *why-provenance* – the main component of an explanation – and show its feasibility for common Linked Data queries. We present an explanation-aware federated query processor prototype and show the presentation of our explanations. Finally, our user study to evaluate the impacts of explanations shows that our query result explanations are helpful for end users to understand the result derivations and make trust judgments on the results.

In the future work, we would like to extend our algorithm to generate *how-provenance*, which explain how a result tuple was derived with the details of the operations performed in the derivation. Furthermore, we would like to conduct the user study with more participants. Finally, we would like to represent our explanations in RDF using explanation vocabularies such as *Ratio4TA* [6].

Acknowledgments: This work is supported by the ANR CONTINT program under the Kolflow project (ANR-2010-CORD-021-02).

References

1. Buneman, P., Khanna, S., Tan, W.C.: Why and where: A characterization of data provenance. In: Proceedings of the 8th International Conference on Database Theory. pp. 316–330. ICDT '01, Springer-Verlag, London, UK, UK (2001)
2. Cheney, J., Chiticariu, L., Tan, W.C.: Provenance in databases: Why, how, and where. Found. Trends databases 1(4), 379–474 (Apr 2009)
3. Corby, O., Gaignard, A., Zucker, C., Montagnat, J.: Kgram versatile inference and query engine for the web of linked data. In: Web Intelligence and Intelligent Agent Technology (WI-IAT), 2012 IEEE/WIC/ACM International Conferences on. vol. 1, pp. 121–128 (Dec 2012)
4. Damásio, C.V., Analyti, A., Antoniou, G.: Provenance for sparql queries. In: Proc. of the 11th International Conference on The Semantic Web - Volume Part I. pp. 625–640. ISWC'12, Springer-Verlag, Berlin, Heidelberg (2012)
5. Gaignard, A.: Distributed knowledge sharing and production through collaborative e-Science platforms. Ph.D. thesis, Universit Nice Sophia Antipolis (2013)
6. Hasan, R.: Generating and summarizing explanations for linked data. In: Presutti, V., dAmato, C., Gandon, F., dAquin, M., Staab, S., Tordai, A. (eds.) The Semantic Web: Trends and Challenges, LNCS, vol. 8465, pp. 473–487. Springer (2014)
7. Hasan, R., Gandon, F.: A Brief Review of Explanation in the Semantic Web. Workshop on Explanation-aware Computing (ExaCt 2012), European Conference on Artificial Intelligence (ECAI 2012) (2012)
8. Horridge, M., Parsia, B., Sattler, U.: Laconic and precise justifications in OWL. In: Proc. of the 7th Int'l Conference on the Semantic Web. pp. 323–338. ISWC '08, Springer-Verlag (2008)
9. Lim, B.Y., Dey, A.K., Avrahami, D.: Why and why not explanations improve the intelligibility of context-aware intelligent systems. In: Proc. of the SIGCHI Conference on Human Factors in Computing Systems. pp. 2119–2128. CHI '09, ACM, New York, NY, USA (2009)
10. McGuinness, D., Furtado, V., Pinheiro da Silva, P., Ding, L., Glass, A., Chang, C.: Explaining semantic web applications. In: Semantic Web Engineering in the Knowledge Society (2008)
11. Morsey, M., Lehmann, J., Auer, S., Ngonga Ngomo, A.C.: Dbpedia SPARQL benchmark performance assessment with real queries on real data. In: Aroyo, L., et al. (eds.) The Semantic Web ISWC 2011, LNCS, vol. 7031, pp. 454–469. Springer Berlin Heidelberg (2011)
12. Owens, A., Seaborne, A., Gibbins, N., mc schraefel: Clustered TDB: A clustered triple store for Jena (November 2008)
13. Pinheiro da Silva, P., McGuinness, D., Fikes, R.: A proof markup language for semantic web services. Information Systems 31(4-5), 381–395 (2006)
14. Theoharis, Y., Fundulaki, I., Karvounarakis, G., Christophides, V.: On provenance of queries on semantic web data. IEEE Internet Computing 15(1), 31–39 (Jan 2011)
15. Tintarev, N., Masthoff, J.: Evaluating the effectiveness of explanations for recommender systems. User Modeling and User-Adapted Interaction 22(4-5), 399–439 (Oct 2012)
16. Wylot, M., Cudre-Mauroux, P., Groth, P.: Tripleprov: Efficient processing of lineage queries in a native rdf store. In: Proceedings of the 23rd International Conference on World Wide Web. pp. 455–466. WWW '14 (2014)