Paul Groth Natasha Noy (Eds.)

ISWC-DC 2014

Doctoral Consortium at ISWC 2014

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

Riva del Garda, Italy, October 20, 2014

Supplementary Proceedings

Preface

This volume contains the papers presented at the Doctoral Consortium of the International Semantic Web Conference 2014 held October 20, 2014 in Rio del Garde, Italy. There were 41 submissions. Each submission was reviewed by 2 program committee members. The committee decided to accept 16 papers. Six of these papers were included in the main conference proceedings published by Springer. Ten papers are included in this volume. We gratefully acknowledge the support of iMinds (http://www.iminds.be) as the Doctoral Consortium sponsor.

October 2014 Paul Groth, Natasha Noy

Program Committee

Harith Alani, KMi, The Open University
Lora Aroyo, VU University Amsterdam
Abraham Bernstein, University of Zurich
Oscar Corcho, Universidad Politécnica de Madrid
Philippe Cudré-Mauroux, University of Fribourg
Fabien Gandon, Inria
Pascal Hitzler, Kno.e.sis Center, Wright State University
Lalana Kagal, MIT Diana Maynard, University of Sheffield
Enrico Motta, KMi, The Open University
Terry Payne, University of Liverpool
Guus Schreiber, VU University Amsterdam
Elena Simperl, University of Southampton

Contents

Interpreting environmental computational spreadsheets

Martine de Vos

Computer Science, Network Institute, VU University Amsterdam, the Netherlands
Martine.de.Vos@vu.nl

Abstract. Environmental computational spreadsheets are important tools in supporting decision making. However, as the underlying concepts and relations are not made explicit, the transparency and re-use of these spreadsheets is severely limited. The goal of this project is to provide a semi-automatic methodology for constructing the underlying knowledge level model of environmental computational spreadsheets. We develop and test this methodology in a limited number of case studies. Our methodology combines heuristics on spreadsheet layout and formulas, with existing methods from computer science. We evaluate our constructed model with both the original developers and their peers.

1 Problem Statement

Current environmental issues, like climate change and biodiversity loss, are universal in their scale and long-term in their impact, their mechanisms are complex, and empirical data are scarce [1–3]. In addition there is an urgent need to find strategies to cope with these issues, and political pressure on the research community is high [3]. Environmental computer models are considered essential tools in supporting environmental decision making by exploring the consequences of alternative policies or management scenarios [1, 2].

Environmental computer models are mainly developed and used by domain scientists and typically implemented as spreadsheets, Fortran programs or in MatLab. These domain scientists have a knowledge level model [4] in their minds containing the important concepts in their domain, and corresponding definitions and interrelations. In the model development process (figure 1) they inevitably make choices about which entities and processes they should include to describe their study area, and how these should be translated and implemented in their computer model. In this way their knowledge model is implicitly included in the computer model, as it is reflected in, for example, the used modelling paradigm, the model structure, the chosen concepts and their interrelations, and the mathematical equations [5].

It is hardly possible to obtain the knowledge level model from the domain scientists themselves. They may give a limited textual explanation about their ideas and choices in their publications, but they rather focus on the computational side of modeling [6]. In fact, they may not even be aware of the knowledge

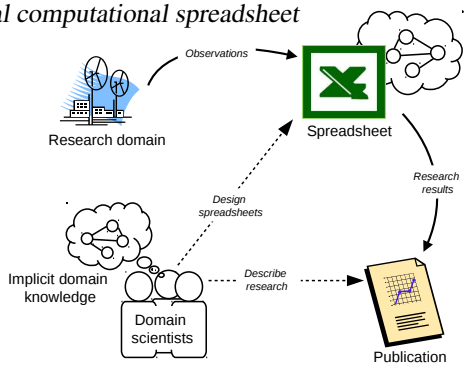


Fig. 1. Rough schematic overview of the current practice of development and use of scientific spreadsheets by domain scientists

level model in their mind [7]. The knowledge level model is, however, essential to understand the meaning and context of the results and insights generated with the computer model. As a consequence, it is hard to make efficient and effective use of environmental computer models by other people than the original developers [6].
The focus of this research is on environmental computer models that are implemented as spreadsheets, from now on called 'environmental computational spreadsheets'. Spreadsheets are widely used by domain scientists to store and manipulate quantitative data from their research projects [8, 9]. A drawback of current spreadsheets is that their free format leads to both complex layout of tables, and sloppy or limited specification of the semantics of the data and calculations [10, 11]. The goal of this project is therefore to provide a methodology for making the underlying knowledge level model of environmental computational spreadsheets explicit. Ideally the various elements in the research process, i.e. observational data, spreadsheet and publications, could be connected to each other through this explicit knowledge level model.

2 Relevancy

Results of this research could enable peers to discuss and assess the scientific quality of environmental computational spreadsheets and to reuse corresponding results and insights. This could contribute to both scientific cooperation and progress, and reliable environmental decision making.

Our research is focused on spreadsheets from the domain of environmental science. However, scientists from other domains may have a similar way of designing and using their spreadsheet models as environmental scientists. We therefore think that the methods and insights from this study might also be applied to spreadsheets from other domains, provided that these spreadsheets contain both domain knowledge and quantitative data.

3 Related Work

Many authors in the field of environmental science advocate standardization of the modelling process, summarized to as 'Good Modelling Practice', to enhance transparency of environmental computer models [12, 1, 13]. Similarly, several studies in computer science, especially in the field of software engineering, suggest how scientific software development could benefit from, for example, clear documentation, relevant training options for scientists and publication of source code [14, 15, 17]. The suggested procedures and guidelines will likely yield more reliable software. However, to guarantee more reliable science, the knowledge included in that software should also be taken into account.

In recent years significant progress has been made in the semantic annotation of scientific models, data sets, and publications. Many tools and techniques are avaliable to connect measurements and terms to the identity of observable entities they quantify [18–21]. A higher level of abstraction that is being investigated is the semantic annotation of scientific practice as a whole. The open provenance model, PROV, [1] helps scientists to document and process provenance information to ensure reproducibility of their analyses [22]. Furthermore, in several scientific disciplines workflow systems [23, 24] are used to integrate and analyse data in a correct and meaningful way.

Several tools and techniques can be used to annotate tabular data. The Data Cube vocabulary [2], for example, provides a means for publishing statistical data as linked data with associated metadata in order to support interpretation and reproducibility. Existing conversion systems like RDF123 [11] and XLWrap [25] allow mapping information from spreadsheets to RDF. And some tools, like Rightfield [8] and Anzo [3], allow the direct annotation of data inside spreadsheet tables.

4 Research Question(s)

In the above described annotation methods the spreadsheets themselves remain largely black-boxes. As a consequence, we may miss out on valuable information on the developers' understanding and interpretation of the system of interest. However, related work also shows that there are plenty solutions to the issue of representing scientific tabular data. As such these studies provide useful tools and information that can be used as a starting point for present study.
The general research question we wish to answer in our study is the following:
To what extent can the underlying knowledge level model of an environmental computational spreadsheet be made explicit?
We refine this question into two more specific subquestions.

1. *How can the underlying knowledge level model of an environmental computational spreadsheet be adequately described?*

[1] W3C Provenance Working Group, http://www.w3.org/2011/prov/
[2] Data Cube, http://www.w3.org/TR/vocab-data-cube/
[3] Anzo, http://www.cambridgesemantics.com

An adequate description of the underlying knowledge level model is defined as a description that
- agrees with the views of the original developers of the spreadsheets.
- can be understood and applied by the original developers of the spreadsheets and their peers.
- allows representation of domain concepts, their hierarchical and property relations, and the computational relations that exist between these concepts

2. *What are the requirements for a methodology for constructing the underlying knowledge level model of an environmental computational spreadsheet?*

5 Hypotheses

When we apply our methodology to an environmental computational spreadsheet, we expect that the resulting constructed knowledge level model is an adequate description of the underlying knowledge level model.

6 Preliminary results

We did two case studies on an existing environmental computer model,i.e., a spreadsheet model that enables policy analyses concerning the Dutch energy system .

In the first case study [7] we manually analyzed the design of the tables and the formulas in the spreadsheets [4]. We semantically characterized the underlying concepts and their interrelations (figure 2) and represented these as an instantiation of an existing ontology, the OM Ontology for units of Measure and related concepts [10]. The main concepts and their interrelations as we identified them in our resulting ontology did not conflict with the developer's views. However, we also discovered that the developers see their models mainly as instruments to perform simulation studies, and therefore focus on the computational aspects.

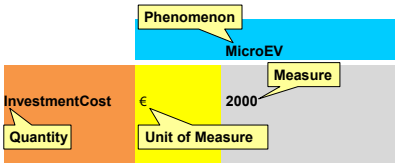


Fig. 2. Example, in outline, of the semantic characterization of terms in a spreadsheet table.

In the second case study [26] we combined automatic and manual methods to analyze the calculation procedures in the spreadsheets. This resulted in a huge

[4] Spreadsheet Examples, http://semanticweb.cs.vu.nl/edesign/

network of interconnected spreadsheet cells (figure 3). We used network analysis to determine which nodes in the graph are the most important, and manually connected these in a simplifed calculation workflow.

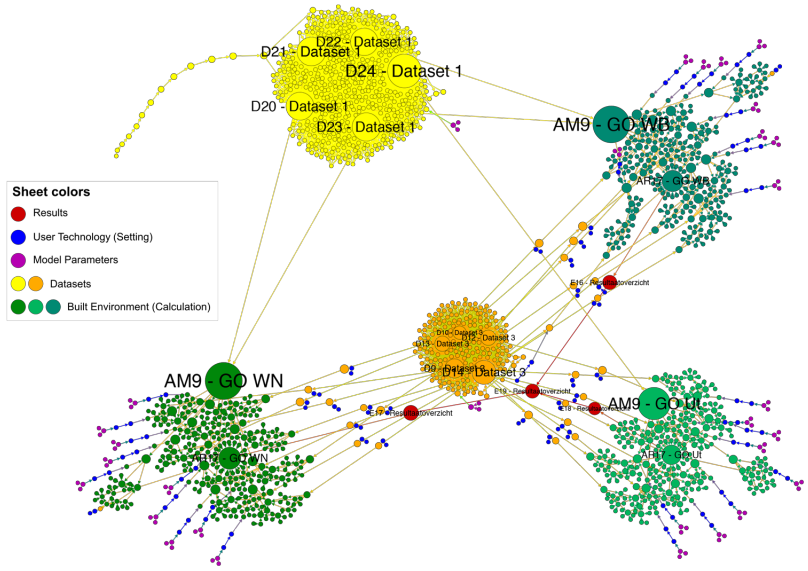


Fig. 3. Network of spreadsheet cells connected through formulas.

7 Approach

In this project we aim at developing a methodology for semi-automatic construction of the underlying knowledge model of an environmental computational spreadsheet. As described above, there are no similar studies on this topic, nor is it possible to access the knowledge level model in the minds of the original developers of environmental computational spreadsheets. We therefore consider it not feasible to set up a study based on quantitative experiments. Instead we choose an approach based on the analyses of a limited number of case studies, and as a consequence, our research has an exploratory character.

Our case studies are all scientific spreadsheet models of existing research projects from the domain of environmental science. We have access to the actual spreadsheets and corresponding datasets, as well as to the publications describing the models and analyses. Furthermore, we have personal contact with the model developers and users.

We develop our methodology based on the in-depth, qualitative analysis of one case study. We will manually analyze the layout of the spreadsheet tables, as well as the formulas connecting the spreadsheet cells. We determine to what extent the observed patterns provide insight in the semantics of the content of the tables, and record our findings in heuristics. Spreadsheet terms can be matched

automatically with concepts of external vocabularies on domain concepts, and on quantitative tabular data. We combine this matching with our layout heuristics to recognize the concepts in the spreadsheets and their interrelations. In addition, we will automatically trace the dependencies between spreadsheet cells through formulas and analyze the resulting networks using techniques for network analysis. We combine these analyses with our heuristics on formulas to construct the calculation workflow in the spreadsheets.

Research question 1 is studied by focusing on the performance of our method in each case study. The different steps in our methodology of constructing the knowledge level model are performed manually by the original developers, and their results are compared with results from our semi-automatic method. We test the applicability of the constructed model by using it to connect concepts from the spreadsheets, with concepts from corresponding publications, or data sets. In a separate user study we will test to what extent peers are able to understand and apply the constructed knowledge level model.

Research question 2 is studied by focusing on the different techniques that are used to describe the knowledge level model. The use of external vocabularies is evaluated by determining how many of the spreadsheet terms could be matched, and how relevant these matches are. We also determine which properties of these vocabularies influence this matching. The use of network analysis techniques is evaluated by determining to what extent these techniques are able to recognize the important variables, as indicated by the original developers, in the calculation workflow. We determine which properties of the spreadsheets influence the performance of our method.

8 Evaluation plan

In order to test our hypothesis we will formulate measurable definitions on what it means for original developers and peers to understand and apply the constructed knowledge level model. Possible indicators we could use are, for example,

- the number of concepts, relations and variables that occur both in the constructed model and in the manual analysis of the original developers.
- the number of connections that can be made from the spreadsheet to corresponding publications and datasets.

9 Reflections

We think our approach is likely to succeed as it is targeted at existing environmental computational spreadsheets. We expect that studying the patterns in these spreadsheets will provide us useful insights on environmental modeling. We also see several promising external developments. Firstly, there is a growing awareness of both the importance of open source code and data, and the importance of methods to provide corresponding credits to modelers and data

providers. Besides, there is an increasing availability of external domain vocabularies.
This PhD research is now at the half way stage. Current work is an extension of our first case study (section 6) and involves the development of a semi-automatic method for defining the concepts and interrelations in spreadsheets. We use the external vocabularies AGROVOC [27] and OM[10], to map and categorize the spreadsheet terms. The plan for the near future is to continue the work of our second case study by developing a semi-automatic method for the construction of the calculation workflow.

Acknowledgements

This publication was supported by the Data2Semantics project in the Dutch national program COMMIT. Guus Schreiber and Jan Top (supervisors), Paul Groth and Lora Aroyo are acknowledged for providing useful comments and suggestions.

References

1. Jakeman, a., Letcher, R., Norton, J.: Ten iterative steps in development and evaluation of environmental models. Environmental Modelling & Software **21**(5) (May 2006) 602–614
2. Schmolke, A., Thorbek, P., DeAngelis, D.L., Grimm, V.: Ecological models supporting environmental decision making: a strategy for the future. Trends in ecology & evolution **25**(8) (August 2010) 479–86
3. van der Sluijs, J.P.: A way out of the credibility crisis of models used in integrated environmental assessment. Futures **34**(2) (March 2002) 133–146
4. Newell, A.: The knowledge level. Artificial Intelligence **18**(1) (January 1982) 87–127
5. Villa, F., Athanasiadis, I., Rizzoli, A.E.: Modelling with knowledge: A review of emerging semantic approaches to environmental modelling. Environmental Modelling & Software **24**(5) (May 2009) 577–587
6. De Vos, M., Janssen, S., Van Bussel, L., Kromdijk, J., Van Vliet, J.V., Top, J.L.: Are environmental models transparent and reproducible enough ? In Wongsosaputro, J., Pauwels, L., Chan, F., eds.: Proceedings of 19th International Congress on Modelling and Simulation. (2011) 2954–2961
7. De Vos, M., Van Hage, W.R., Ros, J., Schreiber, A.: Reconstructing Semantics of Scientific Models : a Case Study. In: Proceedings of the OEDW workshop on Ontology engineering in a data driven world, EKAW 2012, Galway, Ireland (2012)
8. Wolstencroft, K., Owen, S., Horridge, M., Krebs, O., Mueller, W., Snoep, J.L., du Preez, F., Goble, C.: RightField: embedding ontology annotation in spreadsheets. Bioinformatics (Oxford, England) **27**(14) (July 2011) 2021–2
9. Rocha Bernardo, I., Mota, M.S., Santanchè, A.: Extracting and Semantically Integrating Implicit Schemas from Multiple Spreadsheets of Biology based on the Recognition of their Nature. Journal of Information and Database Management **4**(2) (2013) 104–113

10. Rijgersberg, H., Wigham, M., Top, J.L.: How semantics can improve engineering processes: A case of units of measure and quantities. Advanced Engineering Informatics **25**(2) (April 2011) 276–287
11. Han, L., Finin, T., Parr, C., Sachs, J., Joshi, A.: RDF123 : From Spreadsheets to RDF. In: The Semantic Web-ISWC 2008, Springer Berlin Heidelberg (2008) 451–466
12. Refsgaard, J.C.: Modelling guidelinesterminology and guiding principles. Advances in Water Resources **27**(1) (January 2004) 71–82
13. Rykiel, E.J.J.: Testing ecological models: the meaning of validation. Ecological Modelling **90** (1996)
14. Hannay, J.E., Macleod, C., Singer, J., Langtangen, H.P., Wilson, G.: How Do Scientists Develop and Use Scientific Software ? In: Proceedings of the 2009 ICSE workshop on Software Engineering for Computational Science and Engineering, IEEE Computer Society (2009)
15. Segal, J., Morris, C.: Developing scientific software, Part 2. IEEE software **26**(1) (2009) 79
16. Merali, Z.: Why scientific programming doesn't compute. Nature **467** (2010) 6–8
17. Bizer, C., Berlin, F.U., Seaborne, A., Labs, H.p.: D2RQ Treating Non-RDF Databases as Virtual RDF Graphs. In: Proceedings of the 3rd International Semantic Web Conference (ISWC2004). (2004)
18. Navigli, R., Velardi, P., Cucchiarelli, A., Neri, F.: Quantitative and Qualitative Evaluation of the OntoLearn Ontology Learning System. In: Proceedings of the 20th international conference on Computational Linguistics. (2004)
19. Ngomo, A.c.N., Auer, S.: LIMES - A Time-Efficient Approach for Large-Scale Link Discovery on the Web of Data. In: Proceedings of the Twenty-Second international joint conference on Artificial Intelligence, Volume Three., AAAI Press, (2011) 2312–2317
20. Volz, J., Bizer, C., Gaedke, M., Kobilarov, G.: Silk A Link Discovery Framework for the Web of Data. (2009)
21. Moreau, L., Clifford, B., Freire, J., Futrelle, J., Gil, Y., Groth, P., Kwasnikowska, N., Miles, S., Missier, P., Myers, J., Plale, B., Simmhan, Y., Stephan, E., den Bussche, J.V.: The Open Provenance Model core specification (v1.1). Future Generation Computer Systems **27**(6) (June 2011) 743–756
22. Ludäscher, B., Altintas, I., Berkley, C., Higgins, D., Jaeger, E., Jones, M.B., Lee, E.a., Tao, J., Zhao, Y.: Scientific workflow management and the Kepler system. Concurrency and Computation: Practice and Experience **18**(10) (August 2006) 1039–1065
23. Sroka, J., Hidders, J., Missier, P., Goble, C.: A formal semantics for the Taverna 2 workflow model. Journal of Computer and System Sciences **76**(6) (September 2010) 490–508
24. Langegger, A., Wolfram, W.: XLWrap Querying and Integrating Arbitrary Spreadsheets with SPARQL. (2009) 359–374
25. De Vos, M., van Hage, W.R., Wielemaker, J., Schreiber, A.: Knowledge Representation in Scientific Models and their Publications : a Case Study. In: Proceedings of K-CAP 2013 Knowledge Capture Conference, Banff, Canada (2013) 1–2
26. Soergel, D., Lauser, B., Liang, A., Fisseha, F., Keizer, J., Katz, S.: Reengineering Thesauri for New Applications : the AGROVOC Example. Journal of Digital Information **4**(4) (2004) 1–15

Optimizing SPARQL Query Processing On Dynamic and Static Data Based on Query Response Requirements Using Materialization

Soheila Dehghanzadeh

Insight Center for Data Analytics, Ireland, Galway
soheila.dehghanzadeh@insight-centre.org

Abstract. To integrate various Linked Datasets, the data warehousing and the live query processing approaches provide two extremes for the optimized response time and quality respectively. The first approach provides very fast responses but suffers from providing low-quality responses because changes of original data are not immediately reflected on materialized data. The second approach provides accurate responses but it is notorious for long response times. A hybrid SPARQL query processor provides a middle ground between two specified extremes by splitting triple patterns of the SPARQL query between live and local processors based on a predetermined coherence threshold specified by the administrator. However, considering quality requirements while splitting the SPARQL query, enables the processor to eliminate the unnecessary live execution and releases resources for other queries and is the main focus of my work. This requires estimating quality of the response provided with the current materialized data, compare it with user requirements and determine the most selective sub-queries which can boost the response quality up to the specified level with least computational complexity. In this work, we discuss the preliminary result for estimating the freshness of materialized data, as one dimension of the quality, by extending cardinality estimation techniques and explain the future plan.
Keywords. RDF Data Warehouse, View Materialization, SPARQL live querying, quality estimation.

1 Problem Description

Content of the Linked Data is constantly growing and distributed among heterogeneous sources that change their data with various update rates. To process queries over the Linked Data, a data warehousing approach [13] creates a central repository of all triples that is collected by crawlers. However, crawling, storing and maintaining this huge amounts of data is a challenging task due to data volume, velocity and variety. The incremental view maintenance or more recently the higher order incremental view maintenance [10] are designed to efficiently maintain views based on an *update stream* in a relational database environment. But in a Linked Data environment individual RDF datasets or SPARQL endpoints are not designed to report every single update. Thus the data warehouse

has to extract updates by querying original sources. Hence, the view maintenance translates to the live query execution which is a very time consuming job and it is more efficient to defer it as far as current materialized data could fulfil response quality requirements.

Another approach to manage the Linked Data is live querying [6] which processes queries by dereferencing URIs and following relevant links on-demand and naturally incurs very slow response times but with high quality (fresh and complete). Thus, the more time spent on fetching data from source the higher the response quality and vice versa. This represents an inherent trade-off between the time spent on processing the query and the quality of the response which can be considered as a spectrum from a response with high quality and long retrieval time to a response with low quality but short retrieval time.

To mitigate time consuming live querying, the recently proposed hybrid query processing technique [15] suggested to combine the data warehousing with the live query processing techniques. [15] uses coherence values to split triple patterns of a query to a dynamic predicate set for live execution, and a static predicate set for local processing. The coherence of each predicate is defined as the ratio between the cardinality of live results that exist in the materialized data and the total cardinality of live results. [15] achieved individual points in the response time/quality trade-off spectrum by splitting predicates using different coherence thresholds. The coherence threshold of the hybrid approach is strictly defined by the system administrator and has no flexibility depending on response quality requirements of individual queries. However, some query requirements can be fulfilled using the existing local store with no or less live execution. Thus we hypothesize that query response requirements can be exploited to adaptively optimize the splitting process. This releases computational resources for other queries and leads to better scalability and efficient load balancing.

Motivation To motivate the described problem, consider a user who is willing to broadcast a commercial advertisement to specific emails and is satisfied with say 80% freshness in email addresses provided by the response. The incentive of being satisfied with less freshness or quality is to get faster response and consume less computational resources and pay less accordingly. Response requirements are expressed based on the response time and quality by the user or service issuing the query.

Research Question The fundamental research question is how to optimally split an SPARQL query among live and local query processors based on response time and quality requirements? Currently, there exists no automatic way to guide the splitting decision and to adaptively refine it. The fastest response is achieved by fully executing the query on materialized data. In order to compare the quality of response provided with materialized data with required quality, we need to estimate it. Thus, estimating the quality of response provided with materialized data is our first sub-problem. If materialized data couldn't fulfil user quality requirements, parts of the query must be executed lively to boost the quality of the response. Various sub-queries can be redirected to the live engine which leads to various *splitting strategies*. The second sub-problem is to estimate the

quality of the response achieved with an splitting strategy. The optimization is to choose the least costly splitting strategy which is estimated to fulfil required quality.

Contribution Of The Thesis The main contribution in this thesis is to optimize query splitting according to specific requirements of each query. Choosing the splitting strategy that estimated to fulfil required quality with lowest execution time is the ultimate goal of our query processor. To do so, we break down the problem into two sub-problems mentioned in the research question. Here we propose solutions for the first sub-problem. Also quality metrics should be explicitly defined in a Linked Data query processing environment.

In Fig 1 suppose the shaded circle represents the result of executing query on the materialized data and the transparent circle represents the query results of the live engine. "A", which represents query results in the materialized data that doesn't exist in the result provided with the live engine, could contain inferred results from materialized data, data from not available sources and results which has been removed from original sources. However we relax the problem by only considering the latter reason which means we assumed all sources to be available and no inferred data is added to the materialized data. "B" represent the result set which exists both in live and local store. "C" represents the newly added query responses to the live data which still have not been reflected in the materialized data. We simplify the problem by assuming that live engine is able to cover all potential responses. With the above assumptions, *Freshness* quantifies the effect of the deletion on response quality and is defined as $B/(A+B)$. *Completeness* quantifies the effect of addition on response quality and is defined as $B/(B+C)$. In our preliminary experiment we only consider the freshness as the quality metric of the response because in our synthetic data set we are assuming that data can only be removed from real world after materialization which makes triples to become stale. However, in this thesis, we are aiming to consider real world snapshots with both addition and deletion. Thus, both completeness and freshness need to be considered and estimated as quality metrics of various splitting strategies.

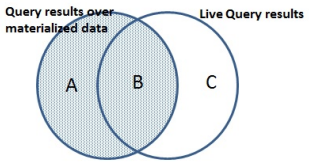


Fig. 1. freshness and completeness

Problem Relevancy By adaptively splitting the query between local and live processors, we prevent unnecessary live execution and reduce network traffic and response time.

2 Related Work

The problem of efficiently processing queries by exploiting the materialized data requires a comprehensive *view management* procedure. This includes the view selection, the view maintenance, the view exploitation and the cost modelling. Interested readers are referred to [5] for detailed explanation on each phase.

Various strategies for view selection(i.e. full or partial materialization) and view maintenance(i.e. immediate or deferred maintenance) can adjust the response time/quality trade-off. DBToaster [10] fully materialize data and immediately apply updates. Thus, there is no option to adjust the time/quality trade-off. It minimizes the cost of processing updates by converting the maintenance task to an efficient code for execution in a relational data model.

[4] chooses the best query set which fully covers a fixed query set and they didn't discuss the effect of postponing maintenance on time/quality trade-off.

The hybrid approach introduced by [15], considers the existing store of a data warehouse as a predefined set of materialized data to be exploited for responding queries. Thus, the hybrid approach actually relaxes the view selection. They use a coherence value to split the query for local or live execution(i.e, maintenance). As alluded before, to maintain the views according to response requirements, we need to adaptively refine the coherance threshold which is not addressed by [15]. On the other hand, [2] recommended RDF indices to materialize based on a given workload aiming to improve performance of the query evaluation. However, in contrast to [4], queries still need to access the original data set because indices are partially covering queries. This approach assumes the original data are not changing and materialized data never gets out-of-date.

[9] is using response requirements to defer unnecessary maintenance tasks based on user preferences when an update stream exists at the data warehouse. We are aiming to target a similar problem but in a Linked Data querying environment where an update stream doesn't exist.

There has been research to estimate the quality of query response provided by materialized data in relational database which requires accurate cardinality estimation and accuracy of involved attributes [3]. The estimation of quality metrics are based on the identity attribute and is achieved by tracking the category change of each type of tuple during each operation. However, in an RDF setting there is no notion of id for tuples. Thus applying that approach for the Linked Data is not directly possible. We hypothesize that statistics of cardinality estimation techniques can be extended to estimate the quality of a query response.

3 Approach

Following our hypothesis, we extend indexing and multi-dimensional histograms for estimating the freshness of a join.

Indexing Based Approaches We designed two indexing structures to estimate join freshness:

- **Simple** Estimate join freshness by simply multiplying freshness of join's predicates. It requires indexing predicates along with their observed freshness. This approach works very well when join result is the Cartesian product of each predicate's result set.
- **CS** Estimate join freshness by using the characteristic set [11] technique. It groups subjects with the same set of predicates together and index it as a

"subject group". The whole dataset can be summarized into a set of "subject group"s with their associated predicates and freshness for each predicate. Analogously, the join's characteristic set(s) consist of individual subject(s) with their requested property(es). To estimate the freshness of a join, we simply sum up the fresh cardinality and the total cardinality of characteristic sets that are super set of the join's characteristic set and divide the fresh cardinality by the total cardinality.

Histogram Based Approaches Histograms and Qtrees are among successful approaches for the data summarization and the join cardinality estimation. Summerization in histogram-based approaches (histogram and Qtree) is achieved by grouping attribute values into buckets and estimating all bucket entries with one summarized value. Join between triple patterns translates to intersection of buckets, assuming that entries are uniformly distributed all over each bucket. Interested readers are referred to [14] for more explanation. To adapt the histogram-based approaches for the freshness estimation problem, we proposed keeping two entries per bucket; the number of fresh and stale entries.

Histogram based approaches require a hashing function to transfer the string representation to numeric representation for processing data. It will determine the uniformity of data distribution which is the main trick leveraged by the histogram to summarize data. Histogram bucket boundaries for each dimension are determined based on a *partitioning rule* which requires a sort and source parameter to specify buckets [12]. We investigated different sort and sources and results are presented in Section 4.

Qtree is an optimized histogram and its buckets are determined by identifying populated areas in the multi dimensional cube using a distance metric. Interested readers are referred to [14] for more detailed explanation.

Evaluation Plan In our preliminary work, we assumed that data can only be deleted from the original dataset after materialization. Thus we could only measure freshness metric. We summarized a synthetically labeled dataset to estimate the freshness of queries without executing query on the original labeled dataset. However, in fact both addition and deletion occur in original data after materialization and therefore we need to be able to estimate both freshness and completeness of the response. For that we need to either extend the cardinality estimation with statistics of both addition and deletion or create individual indexes for estimation of each quality metric. A real example of addition and deletion occurring in materialized data can be observed within consecutive snapshots of the Linked Data Observatory [8]. To solve the first sub-problem in a realistic scenario, i.e.,estimating the quality of response provided with materialized data, we summarize these snapshots with extension of above approaches, compare their estimation performance for individual quality metrics and choose the one with lowest estimation error. We use the same summarization technique for estimating the quality of response provided with an splitting strategy considering that the quality of the live sub-query has increased to 100%. Having the quality estimations of each splitting strategy, we choose the splitting strategy that is estimated to fulfil response requirements and has the lowest

execution time. To evaluate the effectiveness of considering user requirements for optimizing query processing, we will compute the difference among required quality and achieved quality of the response provided with various query processing approaches(i.e., materialized, live, hybrid, adaptive-hybrid) considering their execution time. Adaptive-hybrid approach is aiming to achieve lowest execution time and lowest quality difference.

4 Preliminary Results

Experiment set-up In this experiment, we are tackling the join freshness estimation problem. We used the BSBM benchmark [1] to generate a dataset (374,920 triples with 40 distinct predicates) and a query set. Each triple is either fresh i.e, triple exists after maintenance or stale i.e, triple doesn't exist after maintenance. To split triples between fresh and stale category, we divide predicates among 10 levels of freshness(0-10%, 10-20%, ..., 90-100%) according to r-beta distribution of predicate freshness observed in [15] and assign true or false to triples in dataset based on the freshness value of their predicate. We used the BSBM query templates to generate queries and extract individual joins out of them. In this paper, due to space limitation, we only present actual and estimated freshness for 557 subject-subject joins. To estimate freshness of joins, an index(histogram) is built ,by inserting all individual triples with their associated label to their corresponding index entry (histogram bucket), and used for join processing as explained below:

- **Indexing** We estimated the freshness of subject-subject joins using two indexing approaches: simple freshness multiplication and characteristic set [11]. Figure 2 shows actual and predicted freshness started to disagree after query 285 which is due to existence of bounded triple patterns in joins. Thus index-based approaches lack on joins having a bounded triple pattern.

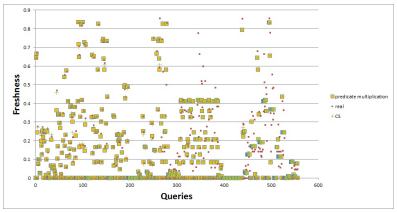
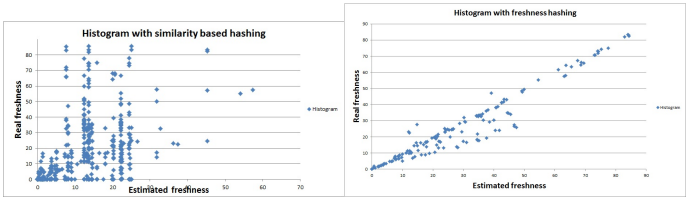


Fig. 2. freshness estimation in subject-subject joins using indexing approaches

- **Histogram** Histogram requires a proper hashing technique to transfer data from the string representation to the numerical representation.
  Choose a Proper Hashing Figure 3(a) shows actual versus predicated freshness using the histogram with similarity-based hashing (mixed hashing

proposed in [14]). In figure 3(a), joins with different actual freshness have



(a) (b)

Fig. 3. predicted vs real freshness in histogram using (a)similarity based hashing (b) freshness hashing for S-S joins

been predicted with similar values due to similarity among bounded objects. This observation along with the fact that histogram estimates neighbouring entries with the same value, strikes the idea of keeping entries with similar freshness close together. Hence, we proposed to sort dimension entries based on their freshness values. Figure 3(b) depicts that sorting dimension entries based on their observed freshness (freshness hashing) leads to better freshness estimates for joins.

Estimation Error We quantified the estimation error of proposed techniques using RMSD normalized error [7] to compare the estimation error over the course of storage space. The normalized RSMD of histograms using the freshness hashing is plotted in Figure 4 and it shows the estimation error of the histogram and the Qtree converged to 0.07 in summary size of 3000 buckets while the simple predicate multiplication (an indexing based approach) consumes less space with a lower estimation error. The error of histogram based approaches can be further decreased by increasing the summary size or implementing more advanced type of histograms.

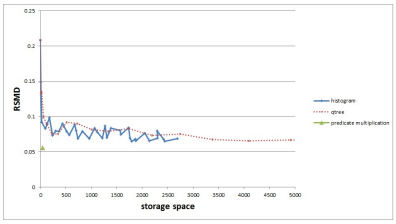


Fig. 4. Freshness estimation error in s-s join in Qtree and histogram using sort hashing

Reflections Traditional approaches estimate join freshness by multiplying freshness of join counterparts. We showed that, this approach mainly lacks on joins with bounded triple patterns(Figure 2). We compared its estimation performance with adapted histograms which leads to less estimation error only by increasing the histogram's summary size. We are planning to reduce the estimation error by using histograms with advanced hashing and adapting other cardinality estimation techniques such as sampling and wavelet.

Acknowledgements I would like to thank Manfred Hauswirth, Josiane Xavier Parreira and Marcel Karnstedt for their valuable comments on the paper.

References

1. Christian Bizer and Andreas Schultz. The berlin sparql benchmark. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 5:1–24, 2009.
2. Roger Castillo, Christian Rothe, and Ulf Leser. *RDFMatView: Idexing RDF Data for SPARQL Queries*. Professoren des Inst. für Informatik, 2010.
3. Debabrata Dey and Subodha Kumar. Data quality of query results with generalized selection conditions. *Operations Research*, 61(1):17–31, 2013.
4. François Goasdoué, Konstantinos Karanasos, Julien Leblay, and Ioana Manolescu. View selection in semantic web databases. *Proceedings of the VLDB Endowment*, 5(2):97–108, 2011.
5. Jonathan Goldstein and Per-Åke Larson. Optimizing queries using materialized views: a practical, scalable solution. In *ACM SIGMOD Record*, volume 30, pages 331–342. ACM, 2001.
6. Olaf Hartig, Christian Bizer, and Johann-Christoph Freytag. Executing sparql queries over the web of linked data. In *The Semantic Web-ISWC 2009*, pages 293–309. Springer, 2009.
7. Rob J Hyndman and Anne B Koehler. Another look at measures of forecast accuracy. *International journal of forecasting*, 22(4):679–688, 2006.
8. Tobias Käfer, Jürgen Umbrich, Aidan Hogan, and Axel Polleres. Towards a dynamic linked data observatory. *LDOW at WWW*, 2012.
9. Alexandros Labrinidis and Nick Roussopoulos. Exploring the tradeoff between performance and data freshness in database-driven web servers. *The VLDB Journal*, 13(3):240–255, 2004.
10. Daniel Lupei, Amir Shaikhha, Christoph Koch, Andres Nötzli, Oliver Andrzej Kennedy, Milos Nikolic, and Yanif Ahmad. Dbtoaster: Higher-order delta processing for dynamic, frequently fresh views. Technical report, 2013.
11. Thomas Neumann and Guido Moerkotte. Characteristic sets: Accurate cardinality estimation for rdf queries with multiple joins. In *Data Engineering (ICDE), 2011 IEEE 27th International Conference on*, pages 984–994. IEEE, 2011.
12. Viswanath Poosala, Peter J Haas, Yannis E Ioannidis, and Eugene J Shekita. Improved histograms for selectivity estimation of range predicates. *ACM SIGMOD Record*, 25(2):294–305, 1996.
13. Giovanni Tummarello, Renaud Delbru, and Eyal Oren. Sindice. com: Weaving the open linked data. In *The Semantic Web*, pages 552–565. Springer, 2007.
14. Jürgen Umbrich, Katja Hose, Marcel Karnstedt, Andreas Harth, and Axel Polleres. Comparing data summaries for processing live queries over linked data. *World Wide Web*, 14(5-6):495–544, 2011.

15. Jürgen Umbrich, Marcel Karnstedt, Aidan Hogan, and Josiane Xavier Parreira. Hybrid sparql queries: fresh vs. fast results. In *The Semantic Web–ISWC 2012*, pages 608–624. Springer, 2012.

A knowledge-based model for instructional design

Frosina Koceva

Università degli Studi di Genova, Italy
frosina.koceva@edu.unige.it

Abstract. This thesis will discuss a knowledge-based model for the design and development of units of learning and teaching aids. The idea behind this work originates from previous theoretical work on ECM - Educational Concept Map (a logical and abstract annotation system, derived from the theories of instructional design), from the open issues in designing instructional authoring system, and from the lack of a well-defined process able to merge pedagogical strategies with systems for the knowledge organization of the domain.

Keywords: Knowledge Management, Topic Maps, Instructional Design, Semantic technologies

1 Problem Statement

Teaching and learning have undergone profound changes in recent years, partly a consequence of the evolution of learning theories, in part dependent on the development and evolution of network technologies. The emergence of constructivist theories of learning models [1] was accompanied by the evolution of the management of learning processes that have facilitated the dynamics of sharing and co-construction of knowledge. The evolution of this scenario prepared the ground to new challenges to research on issues such as interoperability and reusability of learning materials, accessibility, personalization, the definition of standards, quality, etc.

The basic idea that drove this PhD thesis starts from this awareness. The final goal is the definition and development of a knowledge-based model for instructional design with specific focus on educational content designed, to be used in e-learning environments, taking into account the perspectives of development that appears to promise the web today, grounded also on a pedagogical reflection and scientific knowledge we have today.

The approach proposed in this thesis finds its foundation in the work of those who in recent decades have addressed the problems underlying the processes of learning on the one hand and the other knowledge representation, with particular attention to the area of research that goes under the name of the semantic web.

The specific problem I address is a knowledge-based model for the design and development of units of learning and teaching aids. The idea originates from the analysis of the open issues in instructional authoring system, and from the lack of a well-defined process able to merge pedagogical strategies with systems for the knowledge

organization of the domain. In particular, the plan is to ground the work on the ECM - Educational Concept Map - model: a logical and abstract annotation system, derived from the theories of instructional design, developed with the aim of guaranteeing the reusability of both teaching materials and knowledge structures [2]. By means of ECMs, will be possible to design lessons and/or learning paths from an ontological structure characterized by the integration of hierarchical and associative relationships among the educational objectives. Within this context, I will address also the problem to find a "suitable" teaching and learning path through an ECM, i.e., a sequence of concept characterizing the subject matter under definition (a lesson or an entire course), and how these maps can be implemented by means of semantic web standards and technologies [3, 4, 5, 6].

An ECM has a two level structure: the level of concept, i.e. a model of representation of the subject matter where each topic can be associated (level of resources) with one or more resources describing the topic itself (documents, pictures, movies, ...). The plan is therefore to use the level of concept to automatically search relevant resources that are, in turn, associated to each topic of the level of concept in semi automatic way (by approval of the teacher). That is planned to do translating the ECM structure in RDF triples and activating web search extracting data from educational datasets by means of a combination of triples (see research questions).

2 Relevancy

The problems I address in this thesis are still open issues in instructional authoring system, and there is still a lack of a well-defined process able to merge pedagogical strategies with systems for the knowledge organization of the domain. By means of the logical and abstract annotation model of ECMs, it will be possible to design lessons and/or learning paths (see previous section). Once an ECM for a subject matter is defined by a teacher, the design of a lesson (for the teacher) and the surfing through a learning path for a student become a problem of topological sorting (on a graph) [7]. The possibility to make adaptive topological sort on an ECM become a powerful tool both for teachers, during the instructional design phase, and for students, during the learning phase.

Indeed, once an ECM is defined, the teacher can design a lesson adapting it on the previous background of its class, and a student can personalize the learning path depending on its specific knowledge and skills.

3 Related Work

This thesis addresses the problem of instructional authoring system from different point of view trying to integrate into a same model distinct aspects. From the pedagogical point of view, the framework of reference is that depicted by Stelzer and Kingsley in [8] and later revised in [9]; from the point of view of the representation of the subject matter the reference model is that of subject centric networks with specific focus on the Topic Maps model [10]; from the point of view of technology related

works are that carried out by projects and research consortia working on Topic Maps [3, 4, 5, 6]. The real difficulty is the integration between pedagogical and technological aspects in a common tool easy to be used by teachers and students.

4 Research Question(s)

The goal of this thesis is to development a system that assist the teacher for the design of a course by proposing a pliable model of domain knowledge on the base of a course (see relevancy) with the aim of guarantee the reusability of both the teaching aids and knowledge structure of a single disciplines. As to reusability, the ECM are designed to maintain the concept layer separate from the resources, making it possible to provide courses with the same CCM from the ECM but with different resources. Furthermore. for the implementation of efficient information search, metadata will be a central component and an pedagogical ontology describing the characteristics of the didactic resource will be defined. In TM metadata can be isolated and stored separately from the object, but still closely connected to the object. Since we need a representation of domain that can be seen from different points of view, each view showing a different structure, different set of parts, differently related [Prietula and Marchak, 1985] it seemed to us that TM are an appropriate abstraction for designing units of learning. Once an educational objective is define the system will assist the design of the course by automatically identifying the "prerequisites", in other words the concept that a student must know before attending a given unit of learning and the learning outcomes, on base of the relations (see approach). Still in assisting the teacher it remains an open problem how to propose and identify automatically resources to him/her. Accessibility, readability and searchability of web information are crucial for the semi-automatic extension of the knowledge base of our ECM. Integrating information from the two coexisting semantic web exchanging formats (RDF and TM) it's not a straightforward process, but our idea of web information retrieval is based on simplificate mapping of topics to RDF triples for RDF extraction of data from educational datasets. In order to propose to the teacher a possible sequence of topics where each topic can appear only once and cannot be preceded by any of his successors the systems implements a topological order modified algorithm that provides all the possible sequence of topological sorting (see approach). This is possible since between the units of learning and between the topics there could be a propedeutic relations (is-requirement-of) which is unidirectional relation that impose a precedence relationship that makes the unit of learning an acyclic graph.

5 Hypotheses

The availability of "sound" knowledge-based tools increases the productivity of teachers (time and quality) in the daily process of instructional design.

6 Preliminary results

The system is in the initial stage of implementation. The decision on the implementation framework to use for the development was conditioned on the usage of an open source framework that implements the TM standard possibly with active community. One further requirements was the mapping functionality of TM to RDF and vice versa. At first we focused on testing two open-source tools, Ontopia [5] and Wandora [6]; then we opt for building the system on top of Ontopia as being a well-established topic maps creation tool, with good reputation and with a powerful and flexible graphical presentation tool. The system besides the Ontopia engine for the creation of the TM and TM repository has a Resource Engine that handles the versioning and the metadata of the resources. Also further functionalities should be implemented for the topological ordering (see next section) and the assistance in the building of the CCM.

7 Approach

Educational Concept Maps (ECMs) are a formal representation of the subject matter structure in the context of learning environments, and a formal definition of the model is available in [2]. To understand the work of this thesis it is necessary, however, report here some concepts. An ECM is a logical and abstract annotation model created with the aim of guaranteeing the reusability of teaching materials, as well as of knowledge structures, and designed taking into account the pedagogical requirements defined by Educational Modeling Language research group [14]. It has been developed by means of an ontological structure characterized by the integration of hierarchical and associative relationships. Firstly, it asks teachers and instructional designers to focus their attention on learners' profile (in particular educational background, learning and cognitive styles) and objectives. Taking into account these elements, the model suggests how to identify, within the discipline's subject matter, the key concepts and their relationships so as to identify effective strategies of contents presentation and to support the activation of meaningful learning processes.

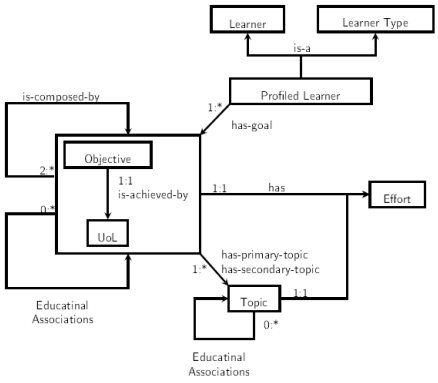


Fig. 1. CADDIE model

According to that model, a profiled learner has a goal identified by an objective (or a composition of objectives) that is achieved by a Unit of Learning (UoL), or by a composition of UoLs. The Course Unit (CU) is the indivisible union of an objective with its unit of learning and can be composed by creating the tree structure of the course (learning units, sub-learning units, etc.). The course units may be connected each other by means of the Educational Associations (EA) that may represent a link or a propaedeutic relationship the units have (see Fig. 1.). In particular, four types of EA have been identified:

· *is-requirement-of*: identifying a transitive and propaedeutic association between two or more topics (e.g., it may be used with the aim of specifying the logical order of contents);

· *is-related-to*: identifying a symmetric association among closely related topics (e.g., it may be used with the aim of creating learning paths without precedence constraints);

· *is-not-related-to*: identifying a symmetric relation of indifference between two or more topics (e.g., it may be used with the aim of making explicit the absence of association among topics);

· *is-suggested-link-of*: identifying not-closely related concepts (e.g., this relationship type may be used in order to suggest in-depth resources, internal or external to the contents repository).

These relation types have been defined with the aim of allowing teachers to create different learning paths (with or without precedence constraints among topics).

The same types of relationship can be found between topics. The latter are the smaller granularity of the ECM model. They represent the concepts of the domain: any subjects a teacher may want to talk about. Moreover, the units of learning are connected to the topics through two relationships:

o *has-primary-topic*: where a primary topic identifies the "prerequisites", in other words the concept that a student must know before attending a given unit of learning;

o *has-secondary-topic*: where secondary topic identifies the concepts that will be explained in the present unit of learning (this kind of topics will have specific learning materials associated).

In the ECM model, a course unit contains an educational objective and a unit of learning. Connected to the UoL there are the topics of the conceptual map describing the domain of the course itself. These topics can be both primary or secondary, depending on the context they are included in, within the unit of learning. Finally the secondary topics contain the material aid. Such resources, grouped in a unit of learning, enable to reach the objective connected to the UoL itself. The CUs allow the teachers to create complex nested structures using the EA.

The ECM model is the theoretical framework for the design of a system, currently in the implementation phase, with some innovative features described in the following:

1. The possibility to publish an Educational Concept Map on the Web and the relationships suggest the different navigation strategies of the underlying subject matter. The possibility to generate a linearized path, useful, for ex-

ample, for a teacher to produce a lesson or a document about a given subject matter. In this latter case, a *Suggested Paths Strategy* is necessary, to be expressed by means of is-requirement-of relationships.

To explain the strategy behind the Suggested Paths Strategy, let us also consider the idea of preparing a lesson on a given argument, using the previous ECM model.

The R_{req} (is-requirement-of) relationships order the topics *T* of the lesson according to the propaedeutics rules, therefore in the graph *G*=(*T*, *E*) there cannot be loops, thus obtaining a Direct Acyclic Graph (DAG), where *T* are nodes and *E arcs, with:* $(t_i,t_j) \in E \leftrightarrow R_{req}(t_i, t_j)$.

In this context, a *Topological Order* is a sequence $S = \{s_1, s_2, \ldots s_{|T|}\}$ where each element *T* appears only once and cannot be preceded by any of his successors; given pair of nodes (t_i,t_j) in S if there exists an arc from t_i to t_j, it follows that the node t_i is before the node t_j in the list: $\forall(t_i,t_j) \in S: (t_i,t_j) \in E \rightarrow i < j$.

The algorithm implementing the Topological Order is derived by Topological sorting algorithm [7] with a main modification in order to get all the possible sequences of topological sorting. Therefore we let the teacher to chose which of this sequences better answers the accomplishment of the didactic objectives. For as much as the topics are topologically ordered this doesn't take into account the distance factor in between the topics, thus a new element (Topic Aider - TA) is introduced in the sequence S before the distant topic to recall the subject. The TA could be an exercise, an example, a text or a valuation test. This recall is also reported in the final sequence in order to highlight not only to the teacher, but also to the student the place where s/he should evoke an determinate argument. The choice to have not a single path but a list of paths to suggest to the author leaving the final choice to the author him/herself, is also to answer to the non-equifinality problem posed in [15]. The "suggested" order lists is on the basis of the principle of reducing as much as possible the distance between two topics of the list that are contiguous on the graph.

In order to implement such a model, Topic Maps (TM) has been chosen. TM is an ISO multi-part standard [3] designed for encoding knowledge and connecting this encoded knowledge to relevant information resources. The standard defines a data model for representing knowledge structures and a specific XML-based interchange syntax, called XML Topic Maps (XTM) [4]. The main elements in the TM paradigm are: *topic* (a symbol used to represent one, and only one, subject), *association* (a relationship between two or more topics) and *occurrence* (a relationship between a subject and an information resource).

Therefore, two layers can be identified into the TMs paradigm:

- the *knowledge layer* representing topics and their relationships, allowing to construct the ECM model;
- the *information layer* describing information resources, to be attached to the ECM topics.

Each topic can be featured by any number of *names* (and *variants* for each name); by any number of *occurrences*, and by its *association role*, that is a representation of the involvement of a subject in a relationship represented by an association. All these features are statements and they have a *scope* representing the context a statement is valid in. Using scopes it is possible to avoid ambiguity about topics; to provide differ-

ent points of view on the same topic (for example, based on users' profile) and/or to modify each statement depending on users' language, etc. Therefore, to solve ambiguity issues, each subject, represented by a topic, is identified by a *subject identifier*. This unambiguous identification of subjects is also used in TMs to merge topics that, through these identifiers, are known to have the same subject (two topics with the same subject are replaced by a new topic that has the union of the characteristics of the two originals).
The knowledge layer can also be used, as introduced in the Problem Statement section, to automatically search relevant resources that are, in turn, associated to each topic of the information layer in semi automatic way (by approval of the teacher).

8 Evaluation plan

The system is presently in a first stage of implementation. Particular attention will be paid to the design and implementation of the user interface. There is a plan during the second half of the next year of my PhD course to experiment the prototype of the system within a selected teachers of EPICT community (www.epict.it), a large community of teachers of the Italian secondary schools. The plan is to measure both the usability of the user interface and the instrument's effectiveness in terms of improving the work of the teacher. In particular, it will seek to evaluate the improvement of daily activities of instructional design carried out by the teacher in terms of both the reduction of design time, and of increased efficacy of the process of instructional design.

I will prepare questionnaire to collect quantitative data, deepen then the results with focus groups.

The experience of teachers with this system will be compared with the previous experience of the same teachers

9 Reflections

The idea behind this thesis has been stimulated by the real "needs" of a community of teachers to have model and tools that facilitates some phases of instructional design. Since the concept representation is independent of its implementation, ECM lends itself for reusability of both teaching materials and knowledge structure. Thus the knowledge structure could be reused for the design of a different course according to the learner target, and new resources could be automatically proposed for the information layer, hence semi-automatically populating (by approval of the teacher) the course map. From student point of view, the subject-centric nature of the TM help learners to identify core concepts, while the extended TM with the learning path assists the student for proper order sequence of studying. Moreover, the underlying model, ECM, is grounded on pedagogical reflections. For these reasons we believe that this model will have a good acceptance by the community of teachers we plan to select for the testing phase.

Acknowledgments.

I would like to express my very great appreciation to my PhD Advisor Prof. Giovanni Adorni, acknowledging his valuable ideas, constructive suggestions and support during the planning and development of the ECM model and this research work.

References

1. Bodner G. M.: Constructivism: A theory of knowledge , J. of Chem. Education, 1986, 63: 873-878.
2. Adorni G., Brondo D., Vivanet G.: A formal instructional model based on Concept Maps, J. of E-Learning and Knowledge Society, 2009, 5(3): 33-43.
3. ISO/IEC 13250-2:2006 Information Technology -- Topic Maps -- Part 2: Data Model. Available at: http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=40017
4. Garshol, L.M., Graham M.: Topic Maps - XML Syntax. Final Draft International Standard, 2006. Available at: http://www.isotopicmaps.org/sam/sam-xtm/
5. Ontopia Project, Available at: http://www.ontopia.net/
6. Wandora Project, Available at: http://wandora.org/
7. Kahn A.B.: Topological sorting of large networks. Communications of the ACM, 1962, 5(11): 558-562.
8. Stelzer J., Kingsley E.H.: An Axiomatic Theory of Subject Matter Structure. Human Resources Research Organization, Alexandria, Virginia (1974).
9. Adorni G., Di Manzo M., Frisiani A.: Evaluation of a formal approach to the structuring of subject matter. J. of Computer Based Instruction, 1981, 2: 35-42.
10. Weber G.E., Eilbracht R., Kesberg S.: Topic Maps as application data model for Subject-centric applications. In: Maicher L, Garshol L.M. (eds.), Procs. 4th. Int. Conf. on Topic Maps Research and Applications, Leipzig, Germany, 15–17 October 2008.
11. Pepper S., Vitali F., Garshol L. M., Gessa N., Presutti V.: A Survey of RDF/Topic Maps Interoperability Proposals. W3C Consortium Working Draft. Available at: http://www.w3.org/TR/rdftm-survey/
12. Garshol, L.M.: The RTM RDF to topic maps mapping: Definition and Introduction, 2003.
13. Shiladitya Munshi, Ayan Chakraborty, Debajyoti Mukhopadhyay: A Hybrid Graph based Framework for Integrating Information from RDF and Topic Map: A Proposal, 2012.
14. Koper R.: Modelling Units of Study from a Pedagogical Perspective: the pedagogical metamodel behind EML. Heerlen: Open Universiteit Nederland, 2001 (http://dspace.learningnetworks.org/handle/1820/36?mode=simple).
15. Ohlsson S.: Some principles of intelligent tutoring. In Lawler, R.W., and Masoud Yazdani, M. (eds.), Artificial Intelligence and Education: Learning Environments and Tutorial Systems v. 1, Intellect Books, 1987.

Profiling the Web of Data

Anja Jentzsch
supervised by Prof. Dr. Felix Naumann
anja.jentzsch@hpi.uni-potsdam.de

Hasso-Plattner-Institute, Potsdam, Germany

Abstract. The Web of Data contains a large number of openly-available datasets covering a wide variety of topics. In order to benefit from this massive amount of open data such external datasets must be analyzed and understood already at the basic level of data types, constraints, value patterns, etc.
For Linked Datasets such meta information is currently very limited or not available at all. Data profiling techniques are needed to compute respective statistics and meta information. However, current state of the art approaches can either not be applied to Linked Data, or exhibit considerable performance problems. This paper presents my doctoral research which tackles these problems.

1 Problem Statement

Over the past years, an increasingly large number of data sources has been published as part of the Web of Data[1]. At the time of writing the Web of Data comprised already roughly 1,000 datasets totaling more than 82 billion triples[2], including prominent examples, such as DBpedia, YAGO, and DBLP. Furthermore, more than 17 billion triples are available as RDFa, Microdata and Microformats in HTML pages[3]. This trend, together with the inherent heterogeneity of Linked Datasets and their schemata, makes it increasingly time-consuming to find and understand datasets that are relevant for integration. Metadata gives consumers of the data clarity about the content and variety of a dataset and the terms under which it can be reused, thus encouraging its reuse.

A Linked Dataset is represented in the Resource Description Framework (RDF). In comparison to other data models, e.g., the relational model, RDF lacks explicit schema information that precisely defines the types of entities and their attributes. Therefore, many datasets provide ontologies that categorize entities and define data types and semantics of properties. However, ontology information is not always available or may be incomplete. Furthermore, Linked Datasets are often inconsistent and lack even basic metadata. Algorithms and tools are needed that profile the dataset to retrieve relevant and interesting metadata analyzing the entire dataset.

[1] The Linked Open Data Cloud nicely visualizes this trend: http://lod-cloud.net
[2] http://datahub.io/dataset?tags=lod
[3] http://webdatacommons.org

Data profiling is an umbrella term for methods that compute metadata for describing datasets. Traditional data profiling tools for relational databases have a wide range of features ranging from the computation of cardinalities, such as the number of values in a column, to the calculation of inclusion dependencies; they determine value patterns, gather information on used data types, determine unique column combinations, and find keys.

Use cases for data profiling can be found in various areas concerned with data processing and data management [12]:

Query optimization is concerned with finding optimal execution plans for database queries. Cardinalities and value histograms can help to estimate the costs of such execution plans. Such metadata can also be used in the area of Linked Data, e.g., for optimizing SPARQL queries.

Data cleansing can benefit from discovered value patterns. Violations of detected patterns can reveal data errors, and respective statistics help measure and monitor the quality of a dataset. For Linked Data, data profiling techniques help validate datasets against vocabularies and schema properties.

Data integration is often hindered by the lack of information on new datasets. Data profiling metrics reveal information on, e.g., size, schema, semantics, and dependencies of unknown datasets. This is a highly relevant use case for Linked Data, because for many openly available datasets only little information is available.

Schema induction: Raw data, e.g., data gathered during scientific experiments, often does not have a known schema at first; data profiling techniques need to determine adequate schemata, which are required before data can be inserted into a traditional DBMS. For the field of Linked Data, this applies when working with datasets that have no dereferencable vocabulary. Data profiling can help induce a schema from the data, which then can be used to find a matching vocabulary or create a new one.

Data Mining: Finally, data profiling is an essential preprocessing step to almost any statistical analysis or data mining task. While data profiling focuses on gathering structural metadata about a dataset, data mining is usually more concerned with gaining new insights about data.

2 Relevancy

There are many commercial tools, such as IBM's Information Analyzer, Microsoft's SQL Server Integration Services (SSIS), or others for profiling relational datasets. However these tool were designed to profile relational data. Linked Data has a very different nature and calls for specific profiling and mining techniques.

Finding information about Linked Datasets is an open issue on the constantly growing Web of Data due to the use cases mentioned above. While most of the Linked Datasets are listed in registries as for instance at the Data Hub (`datahub.io`), these registries usually are manually curated, and thus incomplete or outdated. Furthermore, existing means and standards for describing datasets

are often limited in their depth of information. VoiDand Semantic Sitemapscover basic details of a dataset, but do not cover detailed information on the dataset's content, such as their main classes or number of entities. More detailed descriptions, e.g., information on a dataset's RDF graph structure, topics etc., is usually not available. Data profiling techniques can help to fulfil the need for information about, e.g., classes and property types, value distributions, or entity interlinking.

3 Related Work

While many general tools and algorithms already exist for data profiling, most of them cannot be used for graph datasets, because they assume a relational data structure, a well-defined schema, or simply cannot deal with very large datasets. Nonetheless, some Linked Data profiling tools already exist. Most of them focus on solving specific use cases instead of data profiling in general.

One relevant use case is schema induction, because the lack of a fixed and well-defined schema is a common problem with Linked Datasets. One example for this field of research is the ExpLOD tool [9]. ExpLOD creates summaries for RDF graphs based on class and property usage as well as statistics on the interlinking between datasets based on `owl:sameAs` links.

Li describes a tool that can induce the actual schema of an RDF dataset [11]. It gathers schema-relevant statistics like cardinalities for class and property usage, and presents the induced schema in a UML-based visualization. Its implementation is based on the execution of SPARQL queries against a local database. Like ExpLOD, the approach is not parallelized. Both solutions still take approximately 10h to process a 10 million triples dataset with 13 classes and 90 properties. These results illustrate that performance is a common problem with large Linked Datasets.

An example for the query optimization use-case is presented in [10]. The authors present RDFStats, which uses Jena's SPARQL processor to collect statistics on Linked Datasets. These statistics include histograms for subjects (URIs, blank nodes) and histograms for properties and associated ranges.

Others have worked more generally on generating statistics that describe datasets on the Web of Data and thereby help understanding them. LODStats computes statistical information for datasets from the Data Hub [2]. It calculates 32 simple statistical criteria, e.g., cardinalities for different schema elements and types of literal values (e.g., languages, value data types).

In [4] the authors automatically create VoID descriptions for large datasets using MapReduce. They manage to profile the BTC2010 dataset in about an hour on Amazon's EC2 cloud, showing that parallelization can be an effective approach to improve runtime when profiling large amounts of data.

Finally, the ProLOD++ tool allows to navigate an RDF dataset via an automatically computed hierarchical clustering [5] and along its ontology class tree [1]. Data profiling tasks are performed on each cluster or class dynamically and independently to improve efficiency.

4 Challenges

This section describes selected challenges that I identified as specific to profiling Linked Data and web data, as opposed to profiling relational tables.

Profiling along hierarchies

Vocabularies define classes and their relationships. Ontology classes usually are arranged in a taxonomic (subclass–superclass) hierarchy. While the Web of Data spans a global distributed data graph, its ontology classes build a tree with `owl:Thing` as its root. Analyzing datasets along the vocabulary-defined taxonomic hierarchies yield further insights, such as the data distribution at different hierarchy levels, or possible mappings betweens vocabularies or datasets.

Keys are clearly of vital importance to many applications in order to uniquely identify individuals of a given class by values of (a set of) key properties. In OWL 2 a collection of properties can be assigned as a key to a class using the `owl:hasKey` statement [8].

Nevertheless it has not yet fully arrived on the Web of Data: only one Linked Dataset uses `owl:hasKey` [7]. Thus, actually analyzing and profiling Linked Datasets requires manual, time consuming inspection or the help of tools.

Many languages have a so-called "unique names" assumption. On the web, such an assumption is not possible as real-world entities can be referred to with different URI references.

Heterogeneity

A common practice in the Linked Data community is to reuse terms from widely deployed vocabularies whenever possible, in order to increase homogeneity of descriptions and, consequently, easing the understanding of these descriptions. There are at least 416 different vocabularies to be found on the Web of Data[4]. Some datasets, however, also exist without any defined or dereferencable vocabularies. And even if common vocabularies are used, there is no guarantee that the specifications and constraints are followed correctly.

Nearly all datasets on the Web of Data use terms from the W3C base vocabularies RDF, RDF Schema, and OWL. In addition, 191 (64.75 %) of the 295 datasets in the Linked Open Data Cloud Catalogue use terms from other widely deployed vocabularies [3].

As Linked Datasets cover a wide variety of topics, widely deployed vocabularies that cover all aspects of these topics may not exist yet. Thus, data providers often define proprietary terms that are used in addition to terms from widely deployed vocabularies in order to cover the more specific aspects and to publish the complete content of a dataset on the Web. Currently 190 (64.41 %) out of the 295 datasets use proprietary vocabulary terms with 83.68 % making the term URIs dereferenceable.

Topical profiling

The Web of Data covers not only a wide range of topics, it also contains a number of topically overlapping data sources. Since it provides for data-

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

coexistence, everyone can publish data to it, express their view on things, and use the vocabularies of their choice. Integrating topically relevant datasets requires knowledge on the datasets' content and structure.

The State of the LOD Cloud document **??** gives an overview of the Linked Datasets for each topical domain but there is no fine-grained topical clustering for Linked Datasets. With 504 million inter-dataset links the Web of Data is highly interlinked; 1.6% of all triples are links stating the relationship between the real-world entities in different datasets. Thus a huge topical overlap amongst the datasets is given.

Large scale profiling

With more than 82 billion triples distributed among roughly 1,000 Linked Datasets and more than 17 billion triples available as RDFa, Microdata and Microformats, the need for efficient profiling methods and tools is apparent.

The runtime of profiling tasks as presented in Sec. 7 takes up to hours, e.g., for determining property co-occurrences [6]. Profiling tasks often have the same preprocessing steps, e.g., filtering or grouping the dataset. Thus there is a large incentive and potential to optimize the execution of multiple scripts.

5 Research Questions

The main question in my doctoral research is:

What are the challenges that are specific to profiling Linked Data and web data, as opposed to profiling relational tables?

After identifying four selected challenges, the following questions arise:

Profiling along hierarchies *Does analyzing Linked Datasets along the vocabulary-defined taxonomic hierarchies, such as the data distribution at different hierarchy levels, yield further insights?*

Heterogeneity *How does profiling help analyzing the heterogeneity on the Web of Data?*

Topical profiling *How can topical clusterings for unknown datasets on the constantly growing Web of Data be derived efficently?*

Large scale profiling *How can these huge amounts of Linked Data be profiled efficiently?*

6 Approach

My approach to address the research questions is to tackle each of the identified challenges. The main goal is to reuse existing profiling techniques and adapt them to the Linked Data world.

This section presents possible and if available developed solutions by me to the presented challenges.

Profiling along hierarchies

One example of profiling tasks along the class hierarchy is determining the *uniqueness* of properties as well as the unique property combinations, which can bring insights into the property distribution inside the dataset. It allows for finding relevant (key-candidate) properties for each level in the class hierarchy and see if the relevance is increasing or decreasing along hierarchy.

As I have found, due to the sparsity on the Web of Data, usually neither full key-candidates of properties nor unique property combinations can be retrieved using traditional techniques. Thus I defined the concept of *keyness* as the Harmonic Mean of uniqueness and density of a property[5], allowing to find potential key candidates.

Heterogeneity

Data profiling can be used to provide metadata describing the characteristics of a dataset, for instance its topic and more detailed statistics, like the main classes and properties. Furthermore, data profiling can not only determine the usage of vocabularies but also the help understanding and reusing existing vocabularies. Additionally, it can assist when mapping vocabulary terms.

Topical profiling

The first profiling task is, of course, to discover (and possibly label) these topical clusters. The discovery of which topics an unknown dataset is even about, is already a very helpful insight. Next, any profiling task can be executed on data of a particular topic and compared against the metadata of other topics.

Large scale profiling

The runtime of the profiling tasks takes up to hours already on 1 million triples, e.g., for determining property co-occurrences [6]. A number of different approaches can be chosen when trying to optimize the execution time of algorithms dealing with RDF data in general and data profiling tasks in particular.
Algorithmic optimization: Profiling tasks that have high computational complexity cannot be computed naïvely, e.g., it is infeasible to detect property co-occurrence by considering all possible property combinations. Such metrics require innovative algorithms for efficiently computing the targeted result. If such an algorithm can not be found, approximation techniques (e.g., sampling) may be required. Because these algorithms are often highly specialized for a specific profiling task, they usually do not benefit other tasks.
Parallelization: When dealing with large datasets, a good approach for improving performance is to perform calculations in parallel when possible [12]. This can be done on different levels: dataset, profiling run, profiling task and triples. Cluster-based parallelization based on MapReduce is a reasonable choice when working with Linked Data.
Multi-Query Optimization: A data profiling run usually consists of a number of different tasks, which all have to be computed on the same dataset. Depending on the set of data profiling tasks, different tasks may require the same prepro-

[5] We define the *uniqueness* of a property as the number of unique values per number of total values for a given property; and the *density* of a property as the ratio of non-NULL values to the number of entities.

cessing steps, or perform similar computation steps. Overall execution time can be reduced by avoiding duplicate computations. Similar computation steps may be interweaved to reduce runtime and I/O costs. If different tasks require similar intermediate results, these can be stored in materialized views.

7 Preliminary Results

Initially, I have defined a set of 56 useful data profiling tasks along various groupings to profile Linked Datasets. The have been implemented as Apache Pig scripts and are available online[6].
Furthermore, I illustrated the Web of Data's diversity with the results for four different Linked Datasets [6].

Profiling along hierarchies

When analyzing the uniqueness in the class hierarchy for DBpedia, I found that there are properties that become more specific by class level, thus their uniqueness gets higher for subclasses. For instance, `dbpedia:team` becomes more unique for athletes than it is for all persons. I also found properties that are generic, their uniqueness stays constant throughout the class hierarchy. For instance, `dbpedia:birthDate` is not specific to persons or their subclasses.

Furthermore, I have defined the concept of *keyness* of the property to gap the sparsity on the Web of Data and thus the possibility to find potential key candidates where traditional approaches fail.

Large scale profiling

We have addressed the different approaches to improve Linked Data profiling performance and not only developed LODOP, a system for executing, benchmarking and optimizing Linked Data profiling scripts on Hadoop but also developed and evaluated 3 multi-query optimization rules [6]. We experimentally demonstrated that they achieve their respective goals of optimizing the amount of MapReduce jobs or the amount of data materialized between jobs, thus reducing the profiling tasks runtimes by 70%.

8 Evaluation Plan

For the evaluation, there are three main lines of interest.

Metadata The main goal is to provide comprehensive dataset metadata that helps analyzing the datasets. The metadata can be evaluated on quantity and quality wrt existing metadata on the Data Hub, VoiD and Semantic Sitemaps.

Usability Tools and techniques should have a high usability in terms of results being presented in both human and machine readable ways to achieve better decision making when working with datasets.

Performance evaluation Various aspects of the developed tools should be tested for performance, especially the for huge amounts of data as it is present on the Web of Data.

[6] `http://github.com/bforchhammer/lodop/`

9 Reflections and Conclusion

The main difference in my approach with existing work on Linked Data profiling is to address the shortcomings mentioned in Sec. 3, in particular gathering comprehensive metadata in an efficient way. Within my research I am building on existing profiling techniques for relational data and adapting them according to the different nature of Linked Datasets.

This paper has presented the outline and preliminary results of my doctoral research, in which I am focussing on profiling the Web of Data.

So far I have specified and implemented a comprehensive set of Linked Data profiling tasks and illustrated the Web of Data's diversity with the results for four different Linked Datasets. Furthermore I introduced three common techniques for improving performance of Linked Data profiling and implemented three multi-query optimization rules, reducing profiling taskruntimes by 70%.

References

1. Z. Abedjan, T. Grütze, A. Jentzsch, and F. Naumann. Mining and profiling RDF data with ProLOD++. In *Proceedings of the International Conference on Data Engineering (ICDE)*, 2014. Demo.
2. S. Auer, J. Demter, M. Martin, and J. Lehmann. LODStats – an extensible framework for high-performance dataset analytics. In *Proceedings of the Int. Conf. on Knowledge Engineering and Knowledge Management (EKAW)*, 2012.
3. C. Bizer, A. Jentzsch, and R. Cyganiak. State of the LOD Cloud, 2011.
4. C. Böhm, J. Lorey, and F. Naumann. Creating VoiD descriptions for web-scale data. *Journal of Web Semantics*, 9(3):339–345, 2011.
5. C. Böhm, F. Naumann, Z. Abedjan, D. Fenz, T. Grütze, D. Hefenbrock, M. Pohl, and D. Sonnabend. Profiling Linked Open Data with ProLOD. In *Proceedings of the International Workshop on New Trends in Information Integration (NTII)*, 2010.
6. B. Forchhammer, A. Jentzsch, and F. Naumann. LODOP - Multi-Query Optimization for Linked Data Profiling Queries. In *ESWC Workshop on Profiling & Federated Search for Linked Data (PROFILES)*, 2014.
7. B. Glimm, A. Hogan, M. Krötzsch, and A. Polleres. OWL: Yet to arrive on the Web of Data? In *WWW Workshop on Linked Data on the Web (LDOW)*, 2012.
8. P. Hitzler, M. Krötzsch, B. Parsia, P. F. Patel-Schneider, and S. Rudolph, editors. *OWL 2 Web Ontology Language: Primer*. W3C Recommendation, 2009.
9. S. Khatchadourian and M. P. Consens. ExpLOD: Summary-based exploration of interlinking and RDF usage in the linked open data cloud. In *Proceedings of the Extended Semantic Web Conference (ESWC)*, Heraklion, Greece, 2010.
10. A. Langegger and W. Wöß. RDFStats – an extensible RDF statistics generator and library. In *Proceedings of the International Workshop on Database and Expert Systems Applications (DEXA)*, pages 79–83, Los Alamitos, CA, USA, 2009.
11. H. Li. Data Profiling for Semantic Web Data. In *Proceedings of the International Conference on Web Information Systems and Mining (WISM)*, 2012.
12. F. Naumann. Data profiling revisited. *SIGMOD Record*, 42(4), 2013.

Consistency criteria for a Read/Write Web of Linked Data

Luis-Daniel Ibáñez

LINA, University of Nantes
luis.ibanez@univ-nantes.fr

Abstract. The Linked Data initiative has made possible the publishing and interlinking of a tremendous amount of data, this links can be followed allowing the gathering of related concepts stored in different servers, enabling the answering of powerful queries and richer end-user applications. Current Linked Data is Read-only and many researchers, including its creator Tim Berners-Lee, stand for its evolution to Read-/Write to enable man-machine and machine-machine collaboration in the process of interlinking, cleaning and querying the Linked Data. However, when multiple agents can read/write on the same distributed data, the issue of *consistency* arise. The subject of this thesis is to determine which consistency criterion is the more adequate for the Web of Linked Data and propose tractable algorithms to maintain it.

1 Problem Statement

The quest of this thesis is stated as follows: To determine which consistency criterion is the most adequate for a Read/Write Web of Linked Data and to propose algorithms to maintain it. Such criterion must be strong enough to give guarantees to the consumers, programmers and users when updating or querying Linked Data, and tractable enough to be maintainable under the Web of Linked Data conditions: A steadily growing and projected very high number of *autonomous* participants with a wide-range of dynamics, that together hold a very large volume of data.

2 Relevancy

The Linked Data initiative [4,14] has led to the publication and interlinking of billions of pieces of data in the Resource Description Format, transforming the traditional *Web of Documents* into the *Web of Linked Data*. In the Linked Data ideal, data consumers, *i.e.*, developers and their applications make use of the links between pieces of data to discover and query on related data stored in remote servers, augmenting their added-value and enriching user experience.

However, the current Web of Linked Data is Read-Only, limiting its potential. The evolution to a *Read/Write* Web of Linked Data will have the following benefits:

- Enable truly REST-ful resource oriented Web-APIs [8].
- Break the data silos and empower the users to regain control of their data. Applications can access and modify the data they have permission to. Combined with the Linked Data principles, this also allows networking across platforms and the use of the Web as infrastructure for applications [3].
- Data could be cleaned and evolve with the collaborative intervention of human and machine agents. The knowledge stored in different communities or even by different individuals or applications could *co-evolve* [11].

The Read/Write Web of Linked Data can be seen as network of participants that can update each other's data or copy, modify and exchange updates autonomously. If this exchange is not properly managed, data may diverge uncontrollably, or in a non-deterministic way. This makes impossible the assertion of guarantees on the results of the queries and updates, severely undermining the interaction between the participants [32]. Therefore, to realize the vision of a Read/Write Web of Linked Data, a suitable *consistency* criterion is needed.

3 Related Work

Consistency is a problem transversal to several research communities. In Computer-Supported Cooperative Work is studied in the context of cooperative edition, while in Distributed Systems, Databases, and Semantic Web, often appears when studying the general *replication* problem: A set of *replicas* on which operations are issued by clients (human or agents) and communicate by exchanging messages [33]. When all messages are exchanged, the system must comply with a set of assertions: the consistency criterion.

3.1 Consistency in Computer-Supported Cooperative Work

The main study about consistency in this community was made in the context of Real-Time Editing Systems [28], where the Causality-Convergence-Intention (CCI) model was proposed. The formalisation of all three is studied in Distributed Systems.

Another very important model developed in this community is the *copy-modify-merge* paradigm [10], well known to developers thanks to its implementation in *Version Control Systems*. For the Web of Linked Data characteristics, the most related are *Distributed* Version Control Systems (DVCS), whose prime example is Git[1].

Both models are oriented to text and documents rather than to data.

3.2 Replication and Consistency in Distributed Systems

Replication algorithms and consistency criteria in Distributed Systems can be divided in two main families [25,22]. The first one is the *Pessimistic* category,

[1] `http://git-scm.org`

the goal is to attain a *Strong* consistency, *i.e.*, clients will have the illusion that there is only one replica, fully protected from the effects of failures and concurrent updates. The main consistency criteria is *linearisability* [15].

However, the fundamental CAP Theorem [5] states than in the presence of partitions, whether it be by communication disconnection or by *off-line operations*, is it not possible to have strong consistency without sacrificing high availability. Indeed, the protocols to guarantee strong consistency need to block the system, therefore, the family of *Optimistic* replication algorithms [25] was developed. Optimistic protocols focus on *weak consistency* criteria, where replicas are allowed to diverge during some time (the time of the partition) but remain available, causing the output of some reads (queries) to be different at different replicas during this time window. The main criterion to maintain in this family is *Eventual Consistency* [25]

3.3 Replication and Consistency in Databases

In Distributed Databases, the classification of strategies is done with respect to which replicas can execute updates (master-slaves vs multi-master scheme) and how updates are propagated (eagerly or lazily) [33,24]. Nevertheless, this induces two types of consistency [24]: *Transactional* consistency, which refers to the maintenance of global integrity constraints when concurrent updates occur, and *Mutual* consistency, which refers to data items having identical values at all replicas.

Mutual consistency can be weak or strong, and is equivalent to the weak and strong consistency in Distributed Systems. Transactional consistency can be seen as strong consistency with the extra constraint of integrity (therefore, harder to maintain).

Another related database subject related to our work is *Materialized View Maintenance* [7]. A *target* database stores a snapshot of the result of a query (the view) on one or more *source* databases, when updates happen at the sources the snapshot stored at the target may become inconsistent with the new data at the sources, *i.e.*, the evaluation of the view at the sources is not equal to what is stored. To regain consistency, the target needs to choose, based on a cost model, if it re-evaluates the view, or if integrates the incoming updates (that are assumed to be available somewhere, or arriving through a stream). The design of such integration or *maintenance* algorithms under various constraints is the main issue of this research area.

3.4 Replication in Semantic Web

The first work on update exchange for Semantic Stores appeared in [2]. An ontology was proposed to add semantics to the exchange of diff files. Replication for Semantic Stores has been treated in RDFGrowth [31] where an update exchange algorithm is proposed assuming *monotonic* updates; and in [30], that discusses an adaptation of the popular rsync algorithm to RDF. A full collaborative semantic platform based on these ideas is presented in [29].

However, neither of these works presents any consistency criterion. In fact, there is a gap in the Linked Data and Semantic Web literature concerning the study of criteria specific to them. For example, [14] and [13] do not treat the issue, while [1] refers to the distributed systems and databases criteria discussed in sections 3.2 and 3.3.

There also exist adaptations of DVCS to data encoded in RDF [6,26]. However, their main focus is to bring to the Web their versioning capabilities; there is no study of a general criterion for the Web of Linked Data. Nevertheless, DVCS comply with the postulates of Eventual Consistency, but adding cooperative editing functionalities.

4 Research Questions

1. Which consistency criterion is the most appropriate for a Read/Write Web of Linked Data?
2. Is there a scalable algorithm to maintain such criterion while respecting the autonomy of the participants and without compromising their availability?
3. How to handle conflicting updates with respect to ontology constraints, *i.e.*, when two members disagree semantically?

5 Hypotheses

1. The appropriate consistency criterion lies on the weak category.
2. An optimistic scalable algorithm to maintain such criteria while respecting the autonomy and without compromising the availability exists. Its complexity in time and space is at most polynomial in each of the following parameters: Number of sites, number of updates, size of the datasets, and at most linear in the number of messages exchanged (communication complexity)
3. Semantic agreement at the whole Web of Linked Data relates to transactional consistency and is not attainable in a scalable way.
4. The use of provenance, together with a consistency criterion, help users to solve semantic conflicts locally while still having weaker consistency guarantees at the global level.

6 Approach

From our study of the state of the art we can draw the conclusion that all research communities agree on the CAP theorem result - Strong consistency is not attainable in a scalable and available way - and that the criterion of choice for applications where scalability and availability are a must is *Eventual Consistency*. Therefore, our first approach is to test *Eventual Consistency* and design an algorithm to maintain it on the Web of Linked Data.

We use the recently developed formalism of Conflict-Free Replicated Data Types (CRDTs) [27] for its simple, yet powerful, theoretical foundation and multiple cases of success in industrial and collaborative scenarios [9,23,**?**]. CRDTs also have low footprint compared to DVCS. We claim that participants interested in versioning may put them on top of their stack, while others without the resources or the interest should have the choice of not doing so.

We design a CRDT for the RDF GraphStore type with the SPARQL 1.1 Update Operations, thus, following the W3C recommendations. It will be the first time that CRDTs will be applied to the Web and Linked Data world.

However, Eventual consistency requires two strong assumptions about the network: (i) all updates eventually reach all participants, which implies (ii) the network is connected.

Therefore, as a second approach, we develop a criterion strictly stronger than Eventual consistency based on the View Maintenance notion, and an algorithm to maintain it based on annotated data [20,12]. These tools were developed in the context of Collaborative Data Sharing Systems (CDSS) [21], however, CDSS do not scale beyond few hundreds of participants as their consistency criterion is closer to transactional consistency on heterogeneous databases. Our innovation here is the realization that for the Web of Linked Data we look for the reverse goal, weaker consistency to boost scalability, but we still can adapt the same tools.

Both strategies adapt well to the inclusion of provenance expressed as semirings [20], allowing us to test our fourth hypotheses.

7 Evaluation Plan

Both approaches will undergo a thorough complexity analysis to identify their worst case complexities and test of our hypotheses that the algorithm to maintain consistency is polynomial in the key parameters: number of participants, number of updates and size of the datasets.

With the worst cases established, we plan to analyze the average cases by implementing them and testing them with a high number of participants and current Linked Data dynamics and size [19].

Comparisons will be done with respect to doing nothing and between the two approaches. The questions to answer are: how much is the overhead we need to pay for having each of these levels of consistency? How much more expensive is our view-maintenance based criterion with respect to the eventual consistency provided by our CRDT? Is it affordable at all cases? If not, in which ones it is?

The planned experimental measures are:

- Disk usage (Maximum and average).
- Execution Time of the update exchange protocol at a given participant.
- Time of convergence to the criteria of all the network (still open if its better to measure this in number of times that the algorithm was executed at each participant instead of wall time)
- Number of messages exchanged before convergence.

8 Preliminary Results

SU-Set, a CRDT for RDF-Graph and SPARQL 1.1 Update operations was first sketched in [16] and subsequently specified and analyzed in [17]. The worst case complexities were showed to be indeed polynomial (the communication complexity being constant), confirming our hypotheses.

The view-maintenance based criterion was presented in [18]. The algorithm to maintain it also showed to be polynomial in the key factors except for space in terms of number of updates when the network has a high density: an integer coefficient required by the algorithm to be stored as part of the data annotation may attain values in the order of factorial of the number of participants if the network forms a complete graph.

The experimentation to estimate if this is a concern in the average case, and to estimate the performance in all average cases, is ongoing work. We also work on the full characterization of the parameters that affect our solutions (*e.g.* we already found that network topology is one of them).

9 Conclusion and Reflections

Shifting the Web of Linked Data paradigm from Read-Only to Read/Write will benefit its data quality and general usability. However, when humans and machines have permission to write and exchange updates, consistency criteria are needed to state some guarantees on the evolution of the knowledge it stores and on the queries posed on it.

The main quest of this thesis is to find the most appropriate consistency criteria for the Linked Data Web and to design and analyze the algorithms to maintain them. Such criteria must be strong enough to be useful, but tractable enough to be maintained considering the characteristics of the projected Web of Linked Data: Very high number of autonomous participants that together represent a very high volume of data, a potentially high number of clients (human and machine) performing queries on it, and highly dynamic in nature.

We test the Eventual Consistency criteria from Distributed Systems by designing a Conflict-Free Replicated Data Type for the RDF-Graph type with SPARQL Update operations and found it complied with the imposed requirement of polynomial complexity. We also proposed a second criterion, stronger than Eventual Consistency, based on the notion of View Maintenance and an algorithm to maintain it founded on the concept of semi-ring annotated data. This second criterion complies with the polynomial complexity requirements except for when the network is dense. Experimentation to estimate the performance on average cases is ongoing work.

We believe that in the end our second criteria will be proven affordable for two reasons: first, the Linked Data Web will tend to be a socially-organized network more than random or self-organized and such networks are far from being complete; second, if the value of some of these coefficients is high, implementations may switch to BigInt arithmetics, meaning that the effective cost in storage is still affordable except in the most extreme cases.

Acknowledgements: This thesis is principally supervised by Prof. Pascal Molli (University of Nantes), and co-supervised by Associate Prof. Hala Skaf-Molli (University of Nantes) and Olivier Corby (Researcher at INRIA Sophia-Antipolis Mediterranée). Funding is provided by the French National Research Agency (ANR) through the KolFlow project (code: ANR-10-CONTINT-025).

References

1. Abiteboul, S., Manolescu, I., Rigaux, P., Rousset, M.C., Senellart, P.: Web Data Management. Cambridge University Press, 1st edn. (February 2012)
2. Berners-Lee, T., Connolly, D.: Delta: an ontology for the distribution of differences between rdf graphs. `http://www.w3.org/DesignIssues/Diff` (2004)
3. Berners-Lee, T., O'Hara, K.: The read-write linked data web. Philosophical Transactions of the Royal Society (A 371) (February 2013)
4. Bizer, C., Heath, T., Berners-Lee, T.: Linked data - the story so far. International Journal of Semantic Web Information Systems 5(3), 1–22 (2009)
5. Brewer, E.: Cap twelve years later: How the "rules" have changed. IEEE Computer 45(2) (2012)
6. Cassidy, S., Ballantine, J.: Version control for rdf triple stores. In: Proceedings of the Second International Conference on Software and Data Technologies (ICSOFT) (2007)
7. Chirkova, R., Yang, J.: Materialized views. Foundations and Trends in Databases 4(4) (2011)
8. Coppens, S., Verborgh, R., Sande, M.V., Deursen, D.V., Mannens, E., de Walle, R.V.: A truly read-write web for machines as the next generation web? In: Proceedings of the SW2012 workshop: What will the Semantic Web look like 10 years from now? (2012)
9. Deftu, A., Griebsch, J.: A scalable conflict-free replicated set data type. In: IEEE 33rd International Conference on Distributed Computing Systems (ICDCS) (2013)
10. Dourish, P.: The parting of the ways: Divergence, data management and collaborative work. In: Proceedings of the fourth European Conference on Computer-Supported Cooperative Work (ECSCW) (1995)
11. Engelbart, D., Lehtman, H.: Working together. Byte 13(13) (1988)
12. Green, T.J., Ives, Z.G., Tannen, V.: Reconcilable differences. Theory of Computer Systems 49(2) (2011)
13. Groppe, S.: Data Management and Query Processing in Semantic Web Databases. Springer (2011)
14. Heath, T., Bizer, C.: Linked Data: Evolving the Web into a Global Data Space. Morgan and Claypool (2011)
15. Herlihy, M., Wing, J.: Linearizability: A correctness condition for concurrent objects. ACM Transactions on Programming Languages and Systems (TOPLAS) 12(3) (1990)
16. Ibáñez, L.D., Skaf-Molli, H., Molli, P., Corby, O.: Synchronizing semantic stores with commutative replicated data types. In: Proceedings of the first Semantic Web Collaborative Spaces Workshop (SWCS@WWW'12) (2012)
17. Ibáñez, L.D., Skaf-Molli, H., Molli, P., Corby, O.: Live linked data: Synchronizing semantic stores with commutative replicated data types. International Journal of Metadata, Semantics and Ontologies 8(2) (2013)

18. Ibáñez, L.D., Skaf-Molli, H., Molli, P., Corby, O.: Making linked data writable with provenance semi-rings. Tech. rep., Université de Nantes (2014)
19. Käfer, T., Abdelrahman, A., Umbrich, J., O'Byrne, P., Hogan, A.: Observing linked data dynamics. In: The Semantic Web: Semantics and Big Data, 10th International Conference (ESWC) (2013)
20. Karvounarakis, G., Green, T.J.: Semiring-annotated data: Queries and provenance. SIGMOD Record 41(3) (2012)
21. Karvounarakis, G., Green, T.J., Ives, Z.G., Tannen, V.: Collaborative data sharing via update exchange and provenance. ACM Transactions on Database Systems (TODS) 38(3) (August 2013)
22. Kemme, B., Ramalingam, G., Schiper, A., Shapiro, M., Vaswani, K.: Consistency in distributed systems. Dagstuhl Reports 3(2), 92–126 (2013)
23. Nédelec, B., Molli, P., Mostéfaoui, A., Desmontils, E.: Lseq: an adaptive structure for sequences in distributed collaborative editing. In: ACM Symposium on Document Engineering (DocEng) (2013)
24. Özsu, M.T., Valduriez, P.: Principles of Distributed Database Systems. Springer (2011)
25. Saito, Y., Shapiro, M.: Optimistic replication. ACM Comput. Survey 37(1), 42–81 (2005)
26. Sande, M.V., Colpaert, P., Verborgh, R., Coppens, S., Mannens, E., de Walle, R.V.: R&wbase:git for triples. In: Proceedings of the WWW2013 Workshop on Linked Data on the Web (LDOW) (2013)
27. Shapiro, M., Preguiça, N., Baquero, C., Zawirski, M.: Conflict-free replicated data types. In: International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS). pp. 386–400 (2011)
28. Sun, C., Jia, X., Zhang, Y., Yang, Y., Chen, D.: Achieving convergence, causality preservation, and intention preservation in real-time cooperative editing systems. ACM Transactions on Computer-Human Interaction 5(1), 63–108 (1998)
29. Tummarello, G., Morbidoni, C.: The dbin platform: A complete environment for semantic web communities. Journal of Web Semantics 6(4) (2008)
30. Tummarello, G., Morbidoni, C., Bachmann-Gmür, R., Erling, O.: Rdfsync: Efficient remote synchronization of rdf models. In: 6th International and 2nd Asian Semantic Web Conference (ISWC + ASWC). pp. 537–551 (2007)
31. Tummarello, G., Morbidoni, C., Petersson, J., Puliti, P., Piazza, F.: Rdfgrowth, a p2p annotation exchange algorithm for scalable semantic web applications. In: Proceedings of the MobiQuitous'04 Workshop on Peer-to-Peer Knowledge Management (P2PKM 2004) (2004)
32. Umbrich, J., Karnstedt, M., Parreira, J.X., Polleres, A., Hauswirth, M.: Linked data and live querying for enabling support platforms for web dataspaces. In: Third International Workshop on Data Engineering Meets the Semantic Web (DESWEB) (2012)
33. Wiesmann, M., Pedone, F., Schiper, A., Kemme, B., Alonso, G.: Understanding replication in databases and distributed systems. In: Proceedings of the 20th International Conference on Distributed Computing Systems (ICDCS) (2000)

Retrieval of the most relevant combinations of data published in heterogeneous distributed datasets on the Web*

Shima Zahmatkesh

DEIB - Politecnico di Milano, Milan, Italy
shima.zahmatkesh@polimi.it

Abstract. Finding the most relevant data items among heterogeneous data published on the Web is getting a growing attention in recent years. Retrieving the most relevant data items from a collection of data is a challenge addressed by top-k databases. Accessing heterogeneous and distributed data sources is a challenge addressed by the Semantic Web. How to combine methods and techniques from those two fields is still an open research issue. This doctoral thesis will investigate how the presence of an ontology describing an integrated conceptual model of the data sources and the possibility to encode the users' information needs in top-k queries can make the query answering process faster, more efficient, and able to get more relevant results.
Keywords: Top-k query, Federated databases, heterogeneous data, OBDA, SPARQL, Query Optimization.

1 Relevancy

While a massive amount of data is getting published on the web, searching for data is also attracting a growing attention. Notably, most of the time, users try to satisfy their information needs integrating the results of multiple (vertical) search engines. Those users expect relevant answers to appear in the few first pages of the results, are sensible to correctness, but are rarely interested in completeness. As an example, imagine a student who may want to find the best university for studying; he would take in to account various criteria. He is certainly interested in finding information about the university such as its ranking and the quality of education program. However, he is also concerned with: the quality of life of the city where the university is located in, the public transportation in that city, and the possibility to find cheep accommodation. What he would be satisfied to collect is a set of resources that, once integrated, answer his information need.

2 Problem Statement

The problem that I want to address in this work is how to quickly find on the Web the most relevant answers to queries that span multiple domains and that include user preferences.

*This research is developed under the supervision of Professor Emanuele Della Valle.

Retrieval of the most relevant combinations of data published in heterogeneous distributed datasets on the Web

The nature of the Web implies that the answers have to be found in a multitude of structured and unstructured information, stored in heterogeneous formats across multiple, distributed and possibly overlapping data sources.

Most important for my research work is the way that data is accessed. At a first glance it seems that it is easy to access data in the Web, as all the resources in the Web have a URL. However, in most of the cases, the URLs of the desired resources are unknown. So, usually, users employ search engines or search services to find those "unknown" URLs. It is worth to note that search engines provide only *sorted access* to result items that are return as a ranked list where more relevant results appear first. Note that *random access* to results in those ranked lists is not possible. For example, let assume to search the same term in two search engines (A and B), a user cannot know which is the position of a specific result A in the results of B. To cross check the results of two search engines one has to sequentially read the results of the two search engines.

Moreover, request for (*random*) access to a data resource over the web is more expensive than on a hard disk due to delays introduced by network transmission of the data and the overhead introduced by the usage of the HTTP protocol. Even the request for large amount of data could be expensive because of long transmission times and of protracted processing of the service.

Last but not least, accessing to data resources can be challenging and complex in the case that data is distributed over heterogeneous sources. Structured data can be in relational, XML or RDF formats that can be accessed using SQL, xQuery, SPARQL and, more and more often, Web APIs. Unstructured data like text and multimedia content are even more challenging due to lack of common standards for accessing search services.

The problem I intend to address is how to improve the retrieval of the most relevant combinations of data from a variety of distributed data sources published on the Web caring about the query latency (of the first results), which must be under-second, and relevancy of the first results, which really matters for the users, without posing too much emphasis on completeness, which has little importance in the considered application case. Resource consumption is another important metric in the problem space, because the solution has to scale to thousands of concurrent users as current search engines do.

3 State-of-the-Art[1]

Current search engines do not address this problem; they have just started to offer structured query answering (e.g., Google Knowledge Graph or Wolfram Alpha). Methods for top-k query answering in databases can quickly answer queries, which requires relevant answer first, but they do not scale to the amount of resources published on the Web and cannot deal with data heterogeneity. On the contrary, semantic technologies are able to deal with data heterogeneity. In particular, OBDA uses ontology as a conceptual integrated model for representing the schema of multiple databases and allow issuing federated queries against

[1]More detailed analysis of related work follows in Section 5

a set of heterogeneous data sources. But, semantic technologies are still not optimized to find the most relevant answer first. In the Semantic Web community, approaches to retrieve most relevant data resources are still using the naïve *materialize then sort* query execution schema. The works in top-k query answering using SPARQL are in their initial stage and no work has been done so far on top-k and federated SPARQL.

4 Research Question

Given a user-information need formulated as a top-k query over conceptual integrated model (OBDA), which describes multiple heterogeneous distributed, structured and unstructured data sources published on Web, is it possible to return the top-k best combinations of resources, which answer the information need, in less than a second and to incrementally obtain more results ordered by decreasing relevance in hundreds of milliseconds?

5 Related Works

In this section, I extend the short review of the state-of-the-art presented in Section 3. I start from two important step-stones for my work (Ontology Based Data Access and federated databases) and then I cover top-k query answering in Databases and Semantic Web.

Ontology Based Data Access (OBDA) is a method that I aim to use to address the heterogeneity problem in my research. I chose OBDA because it appears to be a mature approach. Its foundational theory was set in the beginning of 2000s [1] and focused on the DL-Lite family [2] of ontological languages. In 2012, W3C published a recommendation for an ontological language (OWL-2QL) suitable for OBDA using results of those studies, and Gartner foresee its industrial uptake in the next 2-3 years [3].

Federated database is a collection of multiple distributed, autonomous, potentially heterogeneous databases. Federated database systems provides a uniform user interface, enabling users and clients to store and retrieve data with a single query even if the constituent databases are heterogeneous. Principles of federated database systems were set in [4]. In the Semantic Web domain, federation of currently supported SPARQL 1.1 whose syntax and semantics are described in [5].

The top-k query answering problem has been studied in the **database** community to go beyond the naïve *materialize then sort* query execution schema. This schema retrieves all the data resources that match the boolean part of the query, then order them all according to the user defined ranking function and, finally, report the k most relevant results to the user. The state of the art in relational databases contains many algorithms to compute the top-k answer without materializing the answer to the boolean query. The key idea is to consider ranking as a first-class construct and interleave the computation of intermediate results

with their ordering. Ilyas et al. in [6] presented a survey on top-k query processing techniques in relational databases. They introduced various classifications for top-k query processing techniques based on multiple design dimensions, e.g., type of allowed data access method (sorted vs. random) or the type of operation (top-k selection query, top-k join query and top-k aggregate query).

For instance, the Threshold Algorithm (TA) [7] addresses the problem of answering top-k aggregated queries and uses both sorted and random access. The No Random Access algorithm [7] addresses the same problem but exploits only sorted access. The NRA-RJ [8], and Rank-Join algorithm [9] address the problem of top-k join using different mixes of sorted and random access.

RankSQL [10] is an example of DBMS that combines the algorithm presented in the previous paragraph. It introduces an algebraic framework to support efficient evaluation of the top-k queries in relational database systems by extending the relational algebra and query optimization. The key idea is to introduce a ranking operator and to make all other boolean operators rank-aware.

Some initial works on **top-k query answering** are also available in the **Semantic Web** community. Notably, it is possible to express top-k query in SPARQL by using projection functions together with ORDER BY and LIMIT clauses, but only few works investigated the optimization of this class of queries. Magliacane et al. [11] presented SPARQL-RANK, which is an extension of the SPARQL algebra and execution model that support ranking as a first-class SPARQL construct. The new algebra and the execution model provide the splitting of the ranking function and interleaving it with other operators. Wagner et al. [12] studied the top-k join problem in a Linked Data context by adapting the pull/bound rank join (PBRJ) [13] algorithm template for a push-based execution in the linked data setting. The authors of [14] extends SPARQL to querying RDFS annotated by bounded lattice (and thus comes with a partial ordering). Last, but not least, given that computation time is more important than accuracy and completeness, Wagner et al. addressed the problem of approximate top-k processing for the web of the data in [15].

The problem of the evaluation of top-k query in the context of ontology-based access has also been partially addressed. Straccia in [16] frames this problem in the context of relational databases generalizing the results of SoftFacts [17]– an ontology-mediated top-k information retrieval system over relational databases. [18] provides an interesting approach in the context of Web search.

6 Hypothesis

In order to operationalise my research question in hypotheses, I need to describe few classes of queries.

The basic one is the class of top-k SPARQL queries T that was shown to be optimizable in [11,12,15]. E.g., give me the top-5 authors who wrote the largest number of paper that are highly cited. This class of queries can be declared in SPARQL 1.1 and it can be evaluated faster and using less memory (compared to state of the art engines using materialize-then-sort processing schema) by

introducing ranking as first class construct in SPARQL algebra (see SPARQL-RANK algebra [11]) and by using split-and-interleave processing schema.

In my work I intend to investigate the class of top-k SPARQL queries that also include textual matching. Let me name this class top-k textual SPARQL queries T_t. E.g., give me the top-5 authors who wrote the largest number of paper whose title contains "rank", "top-k", and "query". This class cannot be expressed in SPARQL 1.1; few extension exists in proprietary systems (e.g., jena-text and virtuoso full text search).

This class can be split in two subclasses, those that include federated SPARQL and those that do not. Let me name them, centralized top-k textual SPARQL queries T_{tc} and federated top-k textual SPARQL queries T_{tf}.

Last, but not least, those classes of queries can be evaluated under different entailment regimes. In this work, I intend to investigate the cases of simple RDF entailment $T^{\emptyset}$ and the case of an extended version of OWL2QL T^{QL+eq} where it is possible to express simple equations between numerical values. E.g., we would like to express in OWL that the population density of a city is the ratio between the number of inhabitants and the area of the city, so that one can ask for cities ranked by population density even if some of the data sources to access only contain the number of inhabitants and the area of the cities.

Now that I have those classes, I can state my hypotheses as follows:

- *H.1*: Using an extended version of SPARQL, which treats ranking and textual matching as first class constructs, (namely SPARQL-rank$_{tc}$) will make the evaluation of $T^{\emptyset}_{tc}$ queries faster and less memory eager than existing SPARQL engines using materialize-then-sort processing schema
- *H.2*: Extending SPARQL-rank$_{tc}$ to include aspects of federated SPARQL (namely SPARQL-rank$_{tf}$) will make the evaluation of $T^{\emptyset}_{tf}$ queries faster and less memory eager than existing federated SPARQL engines using materialize-then-sort processing schema
- *H.3*: Users with information needs that cannot be homogeneously formulated on heterogeneous data sources, can declare such a need as a query of the class T^{QL+eq}_{tf} and SPARQL-rankT_{tf} will be able to evaluate it.

7 Approach

As the first step, I started an analysis of the state-of-the-art. Reviewing the works done in the domain of top-k query processing in database community is giving me ideas and is guiding me to use top-k query answering in Web domain. I am also becoming familiar with the concept from Web Information Retrieval. My next step is broadening my understating of federated SPARQL and OBDA. I am also working in identifying real use cases that that will be used in the evaluation phase. Finding the suitable datasets and a set of queries are the expected results of this step.

In the next step, I design the evaluation framework that is used to compare my work with the existing ones in order to investigate the hypotheses presented

above. The expected output is a benchmark for top-k SPARQL query answering and a set of the evaluation metrics for fair comparison of alternative approaches.

In parallel to the previous step, I start the main activity of my research that consists of three activities testing the three hypotheses. In the first one, I focus on top-k query and the presence of text searching (*H.1*). Then, I could evaluate *H.2* by extending the work done in testing *H.1* from local system to federated ones and finally, I will focus on the heterogeneity of the data (*H.3*).

8 Evaluation Plan

An evaluation framework is needed to compare the results of my investigations with the existing and appearing solutions. At this stage of the work, I foresee to use the following evaluation metrics and targets:

- *Query latency*: the time required to execute a query and compute the results. I aim to reduce it by two order of magnitude for accessing the first k results and two-three order of magnitude for incrementally obtaining the next results ordered by decreasing relevance.
- *Resource consumption*: I intend to focus on memory usage and I aim to reduce it by one-two order of magnitude.
- *Relevancy of the results*: as metric I indent to use the normalized Discounted Cumulative Gain (nDCG) which is widely used in information retrieval.
- Ability of user to formulate information need.

As dataset for *H.1*, I plan to use DBpedia and Wikipedia or the linked data version of DBLP and Google Scholar. For *H.2*, and *H.3*, I am considering the possibility to exploit Web Data Common[2], a project that extracts structured data from public web pages.

9 Preliminary Results

In my master thesis and in the first months of my PhD I worked on setting up the evaluation framework and I started the investigation of *H.1*.

As for the **evaluation framework**, I extended the DBpedia SPARQL Benchmark (DBPSB) [19] with the capabilities required to compare SPARQL engines on top-k queries and I proposed the Top-k DBpedia SPARQL Benchmark (namely, Top-k DBPSB) that uses the same dataset, performance metrics, and test driver of DBPSB. Top-K DBPSB was run against three SPARQL Engines (Virtouso, Jena TDB, and Sesame). The results of the extensive experimental evaluation confirms that existing solution are poorly optimized for top-k SPARQL queries.

As for the initial investigation of *H.1*, I am comparing the execution time of top-k SPARQL query involving text search between Jena ARQ and the Jena

[2]http://webdatacommons.org/

Text in Apache Jena 2.11.1. As a use case I am considering the need to find authors that have publication in a set of domains, which are defined using a set of keywords. For example, I am try to find the authors who write publications in the two domains: "RDF stream processing" (through the keywords such as "rdf stream", "continuous sparql", and "stream reasoning") and "top-k SPARQL query answering" (through the keywords such as "rdf", "sparql", "top-k", "top k", "order" and "reasoning"). As dataset I am using the dump of DBLP in a RDF store[3]. As expected, the results show that the execution time in Jena Text is one order of magnitude better than in Jena ARQ. I expect to be able to improve by another order of magnitude introducing ARQ-Rank [11].

10 Reflections

Previous work defines a SPARQL rank-aware algebra and extending operators to deal with sorted solution mappings. However, those works do not address the problem of query planning, which is also only partially solved in the relational world [10]. Combining text searching with structured query answering is an active field of research both in database and Semantic Web area, but the usage of top-k query answering methods (*H.1*) has not been explored, yet. Focusing on federated data resources and heterogeneity of the data is one of the most active fields of research in the Semantic Web and database domain, but also in this case the proposed works have not considered the top-k query processing approach (*H.2*). The combination of the top-k query with OBDA has been done in [16], but they consider the OBDA as a layer over the top-k query processing. There is not any exploration in interleaving ordering and reasoning, which require the combination of techniques in database and knowledge representation. To the best of my knowledge, there is not any proposed works that combine the OBDA and the Federation in top-k query processing (*H.3*).

References

1. Lenzerini, M.: Data integration: A theoretical perspective. In Popa, L., Abiteboul, S., Kolaitis, P.G., eds.: PODS, ACM (2002) 233–246
2. Artale, A., Calvanese, D., Kontchakov, R., Zakharyaschev, M.: The dl-lite family and relations. J. Artif. Intell. Res. (JAIR) **36** (2009) 1–69
3. Lapkin, A.: Hype cycle for big data (2012)
4. Ceri, S., Pelagatti, G.: Distributed Databases Principles and Systems. McGraw-Hill, Inc., New York, NY, USA (1984)
5. Aranda, C.B., Arenas, M., Corcho, Ó., Polleres, A.: Federating queries in sparql 1.1: Syntax, semantics and evaluation. J. Web Sem. **18**(1) (2013) 1–17
6. Ilyas, I.F., Beskales, G., Soliman, M.A.: A survey of top-k query processing techniques in relational database systems. ACM Comput. Surv. **40**(4) (2008)
7. Fagin, R., Lotem, A., Naor, M.: Optimal aggregation algorithms for middleware. In Buneman, P., ed.: PODS, ACM (2001)

[3]http://dblp.l3s.de/dblp++.php

8. Ilyas, I.F., Aref, W.G., Elmagarmid, A.K.: Joining ranked inputs in practice. In: VLDB, Morgan Kaufmann (2002) 950–961
9. Ilyas, I.F., Aref, W.G., Elmagarmid, A.K.: Supporting top-k join queries in relational databases. VLDB J. **13**(3) (2004) 207–221
10. Li, C., Chang, K.C.C., Ilyas, I.F., Song, S.: Ranksql: Query algebra and optimization for relational top-k queries. In Özcan, F., ed.: SIGMOD Conference, ACM (2005) 131–142
11. Magliacane, S., Bozzon, A., Valle, E.D.: Efficient execution of top-k sparql queries. In Cudré-Mauroux, P., Heflin, J., Sirin, E., Tudorache, T., Euzenat, J., Hauswirth, M., Parreira, J.X., Hendler, J., Schreiber, G., Bernstein, A., Blomqvist, E., eds.: International Semantic Web Conference (1). Volume 7649 of Lecture Notes in Computer Science., Springer (2012) 344–360
12. Wagner, A., Tran, D.T., Ladwig, G., Harth, A., Studer, R.: Top-k linked data query processing. In Simperl, E., Cimiano, P., Polleres, A., Corcho, Ó., Presutti, V., eds.: ESWC. Volume 7295 of Lecture Notes in Computer Science., Springer (2012) 56–71
13. Schnaitter, K., Polyzotis, N.: Optimal algorithms for evaluating rank joins in database systems. ACM Trans. Database Syst. **35**(1) (2010)
14. Lopes, N., Polleres, A., Straccia, U., Zimmermann, A.: Anql: Sparqling up annotated rdfs. In Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B., eds.: International Semantic Web Conference (1). Volume 6496 of Lecture Notes in Computer Science., Springer (2010) 518–533
15. Wagner, A., Bicer, V., Tran, T.: Pay-as-you-go approximate join top-k processing for the web of data. In Presutti, V., d'Amato, C., Gandon, F., d'Aquin, M., Staab, S., Tordai, A., eds.: ESWC. Volume 8465 of Lecture Notes in Computer Science., Springer (2014) 130–145
16. Straccia, U.: On the top-k retrieval problem for ontology-based access to databases. In Pivert, O., Zadrozny, S., eds.: Flexible Approaches in Data, Information and Knowledge Management. Volume 497 of Studies in Computational Intelligence. Springer (2013) 95–114
17. Straccia, U.: Softfacts: A top-k retrieval engine for ontology mediated access to relational databases. In: SMC, IEEE (2010) 4115–4122
18. Fazzinga, B., Gianforme, G., Gottlob, G., Lukasiewicz, T.: Semantic web search based on ontological conjunctive queries. J. Web Sem. **9**(4) (2011) 453–473
19. Morsey, M., Lehmann, J., Auer, S., Ngomo, A.C.N.: Dbpedia sparql benchmark - performance assessment with real queries on real data. In Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N.F., Blomqvist, E., eds.: International Semantic Web Conference (1). Volume 7031 of Lecture Notes in Computer Science., Springer (2011) 454–469

A Data-flow Language for Big RDF Data Processing

Fadi Maali *

Insight Centre for Data Analytics, National University of Ireland Galway
fadi.maali@insight-centre.org

Abstract. When analysing large RDF datasets, users are left with two main options: using SPARQL or using an existing non-RDF-specific big data language, both with its own limitations. The pure declarative nature of SPARQL and the high cost of evaluation can be limiting in some scenarios. On the other hand, existing big data languages are designed mainly for tabular data and, therefore, applying them to RDF data results in verbose, unreadable, and sometimes inefficient scripts. My PhD work aims at enhancing programmability of big RDF data. The gaol is to augment the existing tools with a declarative dataflow language that focuses on the analysis of large-scale RDF data. Similar to other big data processing languages, I aim at identifying a set of basic operators that are amenable to parallelisation and at supporting extensibility via user-defined custom code. On the other hand, a graph-based data model and support for pattern matching as in SPARQL are to be adopted. Giving the focus on large-scale data, scalability and efficiency are critical requirements. In this paper, I report on my research plan and describe some preliminary results.

1 Problem Statement

Petabytes and terabytes datasets are becoming commonplace, especially in industries such as telecom, health care, retail, pharmaceutical and financial services. To process these huge amounts of data, a number of distributed computational frameworks have been suggested recently [7, 13, 31]. Furthermore, there has been a surge of activity on layering declarative languages on top of these platforms. Examples include Pig Latin from Yahoo [16], DryadLINQ from Microsoft [30], Jaql from IBM [3], HiveQL from Facebook [27] and Meteor/Sopremo [11].

In the Semantic Web realm, this surge of analytics languages was not reflected despite the significant growth in the available RDF data. To analyse large RDF datasets, users are left mainly with two options: using SPARQL [10] or using an existing non-RDF-specific big data language. I argue that each of these options has its own limitations.

SPARQL is a graph pattern matching language that provides rich capabilities for slicing and dicing RDF data. The latest version, SPARQL 1.1, supports

* Supervisor: Prof. Dr. Stefan Decker. Expected graduation date: Fall 2016

also aggregation and nested queries. Nevertheless, the pure declarative nature of SPARQL obligates a user to express their needs in a single query. This can be unnatural for some programmers and challenging for complex needs [15, 9]. Furthermore, SPARQL evaluation is known to be costly [18, 23] and requires all data to be transformed into RDF beforehand.

The other alternative of using an existing big data language such as Pig Latin or HiveQL has also its own limitations. These languages were designed for tabular data mainly, and, consequently, using them with RDF data commonly results in verbose, unreadable, and sometimes inefficient scripts [21].

My PhD work aims at enhancing **programmability of big RDF data**. The gaol is to augment the existing tools with a declarative dataflow language that focuses on the analysis of large-scale RDF data. Similar to other big data processing languages, I aim at identifying a small set of basic operators that are amenable to parallelisation and at supporting extensibility via user-defined custom code. On the other hand, a graph-based data model and support for pattern matching as in SPARQL are to be adopted. Giving the focus on large-scale data, scalability and efficiency are critical requirements. Moreover, I intend to work towards relaxing the prerequisite of full transformation of non-RDF data and facilitating processing of RDF and non-RDF data together.

2 Relevancy

Data is playing a crucial role in societies, governments and enterprises. For instance, data science is increasingly utilised in supporting data-driven decisions and in delivering data products [14, 20]. Furthermore, scientific fields such as bioinformatics, astronomy and oceanography are going through a shift from "querying the world" to "querying the data" in what commonly referred to as e-science [12]. The main challenge nowadays is analysing the data and extracting useful insights from it.

Declarative languages simplify programming and reduce the cost of creation, maintenance, and modification of software. They also help bringing the non-professional user into effective communication with a database [5]. In 2008, the Claremont Report on Database Research identified declarative programming as one of the main research opportunities in the data management field [1].

My PhD work intends to facilitate analysing large amount of RDF data by designing a declarative language. The fast pace at which the data is growing and the expected shortage in people with data analytical skills [6], make users' productivity of paramount importance. Moreover, By embracing the process of RDF and non-RDF data together, my hope is to increase the utilisation of the constantly growing size of RDF data.

3 Related Work

A large number of declarative languages were introduced recently as part of the big data movement. These languages vary in their programming paradigm, and in

their underlying data model. Pig Latin [16] is a dataflow language with a tabular data model that also supports nesting. Jaql [3] is a declarative scripting language that blends in a number of constructs from functional programming languages and uses JSON for its data model. HiveQL [27] adopts a declarative syntax similar to SQL and its underlying data model is a set of tables. Other examples of languages include Impala[1], Cascalog[2], Meteor [11] and DryadLINQ [30]. [26] presented a performance as well as language comparison of HiveQL, Pig Latin and Jaql. [22] also compared a number of big data languages but focuses on their compilation into a series of MapReduce jobs.

In the semantic web field, SPARQL is the W3C recommended querying language for RDF. A number of extensions to SPARQL were proposed in the literature to support search for semantic associations [2], and to add nested regular expressions [19] for instances. However, these extensions do not change the pure declarative nature of SPARQL. There are also a number of non-declarative languages that can be integrated in common programming languages to provide support for RDF data manipulation [17, 25]. In the more general context of graph processing languages, [29] provides a good survey.

4 Research Questions

A core part of a declarative language is its underlying data model. A data model consists of a notation to describe data and a set of operations used to manipulate that data [28]. SPARQL Algebra [18] is the data model underlying SPARQL. SPARQL Algebra cannot be used as an underlying model for the declarative language I am working on for the following reasons:

- It is not fully composable. The current SPARQL algebra transitions from graphs (i.e. the initial inputs) to sets of bindings (which are basically tables resulting from pattern matching). Subsequently, further operators such as Join, Filter, and Union are applied on sets of bindings. In other words, the flow is partly "hard-coded" in the SPARQL algebra and a user cannot, for instance, apply a pattern matching on the results of another pattern matching or "join" two graphs. In a dataflow language, the dataflow is guided by the user and cannot be limited to the way SPARQL Algebra imposes.
- It assumes all data is in RDF.
- The expressivity of SPARQL comes at the cost of high evaluation complexity [18, 23].

Therefore, the main challenge is to define an adequate data model that embraces RDF and non-RDF data and strikes a balance between expressivity and complexity. Accordingly, my research questions are:

RQ1: What is the appropriate data model to adopt?
RQ2: How do we achieve efficient scalable performance?
RQ3: How do we enable processing of RDF and non-RDF data together?

[1] `https://github.com/cloudera/impala`
[2] `http://cascalog.org/`

5 Hypotheses

We introduce several hypotheses that we would like to test in our research.

H1: A new data model can be defined to underlie a dataflow language for RDF data. The expressivity and complexity of this data model can be determined.
H2: Algebraic properties of the new data model can be exploited to enhance performance.
H3: Scalable efficient performance can be achieved by utilising state-of-the-art distributed computational frameworks.
H4: Integrating transformation to RDF as part of the data processing enables processing RDF and non-RDF data together and can eliminate the need of *full* transformation to RDF.

6 Approach & Preliminary Results

We had an initial proposal for a data model and a dataflow language. Our goal is to iteratively refine the model **(H1, H2)** and our implementation **(H3)** and then extend it to include non-RDF data and data transformation **(H4)**. The next two subsections summarise our preliminary results.

6.1 RDF Algebra

RDF Algebra is our proposed data model. This algebra defines operators similar to those defined in SPARQL algebra but that are fully composable. To achieve such composability, the algebra operators input and output are always a pair of a graph and a corresponding table **(H1)**. The core set of expressions in this algebra are: atomic, projection, extending, triple pattern matching, filtering, cross product and aggregation. The syntax and the semantics of these expressions have been formally defined and their expressivity in comparison to SPARQL is captured by the following lemma.

Lemma 1. *RDF Algebra expressions can express SPARQL 1.1 basic graph patterns with filters, aggregations and assignments.*

We have also started to study some unique algebraic properties of our data model **(H2)**. Cascading triple patterns and joins in the RDF algebra results in some unique optimisation opportunities. Therefore, we defined a partial ordering relationship between triple patterns to capture subsumption among results. Consequently, evaluation plans can be optimised and intermediary results can be reused in order to enhance evaluation performance **(H3)**.

The innovative part of this model is the pairing of graphs and tables, which, to the best of our knowledge, was not reported in the literature before. This ensures full composability and can potentially accommodate tabular data (with empty graph component that can be populated via transforming the tabular data only when necessary) **(H4)**.

6.2 SYRql, A Dataflow Language

Our current dataflow language that is grounded in the algebra defined before is called SYRql. A SYRql script is a sequence of statements and each statement is either an assignment or an expression. The syntax of SYRql borrows the use of "− >" syntax from Jaql to explicitly show the data flow. Whereas pattern matching in SYRql uses identical syntax to basic graph patterns of SPARQL. SPARQL syntax for patterns is intuitive, concise and well-known to many users in the Semantic Web field. We hope that this facilitates learning SYRql for many users.

The current implementation[3] uses JSON[4] for internal representation of the data. Particularly, we use JSON arrays for bindings and JSON-LD [24] to represent graphs. SYRql scripts are parsed and then translated into a directed acyclic graph (DAG) of MapReduce jobs **(H3)**. Sequences of expressions that can be evaluated together are grouped into a single MapReduce job. Finally, the graph is topologically sorted and the MapReduce jobs are scheduled to execute on the cluster. Our initial performance evaluation showed comparative performance to well-established languages such as Pig Latin and Jaql (Figure 1).

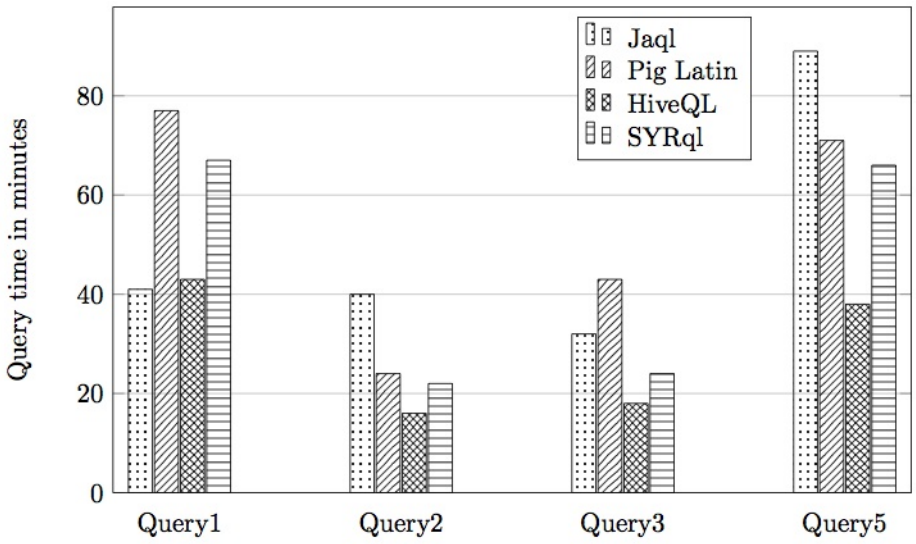


Fig. 1. Query processing times

[3] https://gitlab.deri.ie/Maali/syrql-jsonld-imp/wikis/home
[4] http://json.org

7 Evaluation Plan

We are currently conducting a formal study of the data model, its algebraic properties, its complexity and expressivity. We plan to compare it to First-Order Logic languages **(H1, H2)**.

For performance evaluation, we have started comparing response time that our implementation provides to that of SPARQL and existing big data languages **(H3)**. Figure 1 shows initial results. The benchmark we used is based on the Berlin SPARQL Benchmark (BSBM) [4] that defines an e-commerce use case. Specifically, we translated a number of queries in the BSBM Business Intelligence usecase (BSBM BI)[5] into equivalent programs in HiveQL, Pig Latin and Jaql. To the best of our knowledge, this is the first benchmark that uses existing big data languages with RDF data.

Furthermore, we plan to use some data manipulation scenarios from bioinformatics research to guide requirement collection for processing RDF and non-RDF data **(H4)**. We plan to conduct a performance evaluation and a user study to evaluate our work on this regards.

8 Reflections

We base our work on a good understanding of Semantic Web technologies as well as existing Big Data techniques and languages. The initial results we have collected are promising. Nevertheless, the current implementation leaves rooms for improvements. We plan to use RDF compression techniques such as HDT [8] and to experiment with distributed frameworks other than MapReduce such as Spark. Finally, we believe that our data model and its algebraic properties can yield fruitful results that can further be applied in tasks like caching RDF query results, views management and query results reuse.

References

1. Rakesh Agrawal et al. The Claremont Report on Database Research. *SIGMOD Rec.*, 2008.
2. Kemafor Anyanwu and Amit Sheth. P-queries: enabling querying for semantic associations on the semantic web. In *WWW*, 2003.
3. Kevin S. Beyer, Vuk Ercegovac, Rainer Gemulla, Andrey Balmin, Mohamed Y. Eltabakh, Carl-Christian Kanne, Fatma Özcan, and Eugene J. Shekita. Jaql: A Scripting Language for Large Scale Semistructured Data Analysis. *PVLDB*, 2011.
4. Christian Bizer and Andreas Schultz. The Berlin SPARQL Benchmark. *IJSWIS*, 2009.
5. Donald D. Chamberlin and Raymond F. Boyce. SEQUEL: A Structured English Query Language. In *SIGFIDET*, 1974.

[5] `http://wifo5-03.informatik.uni-mannheim.de/bizer/berlinsparqlbenchmark/spec/BusinessIntelligenceUseCase/index.html`

6. Michael Chui, Brad Brown, Jacques Bughin, Richard Dobbs, Charles Roxburgh, and Angela Hung Byers James Manyika. Big data: The next frontier for innovation, competition, and productivity. Technical report, McKinsey Global Institute, 2011.
7. Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. In *OSDI*, 2004.
8. Javier D Fernández, Miguel A Martínez-Prieto, Claudio Gutiérrez, Axel Polleres, and Mario Arias. Binary rdf representation for publication and exchange (hdt). *Web Semantics: Science, Services and Agents on the World Wide Web*, 19:22–41, 2013.
9. Stefan Hagedorn and Kai-Uwe Sattler. Efficient Parallel Processing of Analytical Queries on Linked Data. In *OTM*, 2013.
10. Steve Harris and Andy Seaborne. SPARQL 1.1 Query Language. W3C Recommendation 21 March 2013. http://www.w3.org/TR/sparql11-query/.
11. Arvid Heise, Astrid Rheinländer, Marcus Leich, Ulf Leser, and Felix Naumann. Meteor/Sopremo: An Extensible Query Language and Operator Model. In *BigData*, 2012.
12. Tony Hey, Stewart Tansley, and Kristin Tolle, editors. *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft Research, 2009.
13. Michael Isard, Mihai Budiu, Yuan Yu, Andrew Birrell, and Dennis Fetterly. Dryad: Distributed Data-parallel Programs from Sequential Building Blocks. In *EuroSys*, 2007.
14. Mike Loukides. What is Data Science? *O'Reilly radar*, 6 2010.
15. Fadi Maali and Stefan Decker. Towards an RDF Analytics Language: Learning from Successful Experiences. In *COLD*, 2013.
16. Christopher Olston, Benjamin Reed, Utkarsh Srivastava, Ravi Kumar, and Andrew Tomkins. Pig Latin: a Not-so-foreign Language for Data Processing. In *SIGMOD*, 2008.
17. Eyal Oren, Renaud Delbru, Sebastian Gerke, Armin Haller, and Stefan Decker. Activerdf: Object-oriented semantic web programming. In *WWW*, 2007.
18. Jorge Pérez, Marcelo Arenas, and Claudio Gutierrez. Semantics and Complexity of SPARQL. ISWC, 2006.
19. Jorge Pérez, Marcelo Arenas, and Claudio Gutierrez. nSPARQL: A navigational language for RDF. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2010.
20. Foster Provost and Tom Fawcett. Data Science and its Relationship to Big Data and Data-Driven Decision Making. *Big Data*, 1(1), March 2013.
21. Padmashree Ravindra, HyeongSik Kim, and Kemafor Anyanwu. An Intermediate Algebra for Optimizing RDF Graph Pattern Matching on MapReduce. In *ESWC*, 2011.
22. Caetano Sauer and Theo Haerder. Compilation of query languages into mapreduce. *Datenbank-Spektrum*, 2013.
23. Michael Schmidt, Michael Meier, and Georg Lausen. Foundations of sparql query optimization. In *ICDT*, 2010.
24. Manu Sporny, Dave Longley, Gregg Kellogg, Markus Lanthaler, and Niklas Lindström. JSON-LD 1.0. W3C Recommendation 16 January 2014.
25. Steffen Staab. Liteq: Language integrated types, extensions and queries for rdf graphs. *Interoperation in Complex Information Ecosystems*, 2013.
26. Robert J Stewart, Phil W Trinder, and Hans-Wolfgang Loidl. Comparing High Level MapReduce Query Languages. In *APPT*. 2011.

27. Ashish Thusoo, Joydeep Sen Sarma, Namit Jain, Zheng Shao, Prasad Chakka, Ning Zhang 0002, Suresh Anthony, Hao Liu, and Raghotham Murthy. Hive - a Petabyte Scale Data Warehouse Using Hadoop. In *ICDE*, 2010.
28. J.D. Ullman. *Principles of Database and Knowledge-base Systems*, chapter 2. Computer Science Press, Rockville, 1988.
29. Peter T. Wood. Query Languages for Graph Databases. *SIGMOD*, 2012.
30. Yuan Yu, Michael Isard, Dennis Fetterly, Mihai Budiu, Úlfar Erlingsson, Pradeep Kumar Gunda, and Jon Currey. DryadLINQ: a System for General-purpose Distributed Data-parallel Computing Using a High-level Language. In *OSDI*, 2008.
31. Matei Zaharia, Mosharaf Chowdhury, Tathagata Das, Ankur Dave, Justin Ma, Murphy McCauley, Michael J Franklin, Scott Shenker, and Ion Stoica. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. In *NSDI*, 2012.

A rule-based approach to address semantic accuracy problems on Linked Data
(ISWC 2014 - Doctoral Consortium)

Leandro Mendoza[1]

LIFIA, Facultad de Informática, Universidad Nacional de La Plata, Argentina

1 Problem Statement

In 2001, Berners-Lee et al. [2] defined the *Semantic Web* (SW) as an extension of the current Web in which information is given well-defined meaning through the use of common standards and technologies to facilitate the sharing and reuse of data. In 2006, the related term *Linked Data* (LD) [3, 7] was proposed as a way to identify a set of best practices for publishing data using SW tools that allow to link these isolated datasets in a large network of distributed data [17]. Since then, the number of available datasets that follow the SW and LD ideas has been considerably increasing, leading to what is currently known as the *Web of Data* (WoD).

Although this WoD provides tons of information (see the LD cloud[1]), evidence shows that it is only as usable as its quality: there is a lot of noise in current SW datasets[2] and just a few applications can effectively exploit the inherent potential of this well-defined and structured information. These SW datasets covers different domains and have different levels of quality: from "high-quality" curated SW datasets (for example, in life-science domain) to those which were extracted from unstructured and semi-structured sources or were the result of a *crowdsourcing* process (for example, DBPedia [11]). Some of the data-quality problems that affects those datasets are out-of-date values, incomplete or incorrect data, inconsistencies, etc. Most of these problems arise during the creation process of SW data, due to errors in the original data source, the tools employed to convert or create SW data, misuse of ontologies, etc.

The main problem addressed in my PhD work is about to improve existing SW datasets (also new and emerging ones) that suffer from quality problems by taking advantage of information available in other SW datasets with presumed and relatively "high" quality. This work will be mainly focused in two related quality dimensions: *"semantic accuracy"* (values that do not correctly represent

[1] http://linkeddata.org/

[2] The term "Semantic Web (SD) dataset" used in this document is also referred in others works as "Linked Data (LD) set", "RDF dataset" or generally as "dataset in the WoD".

the real state of real-world objects) [6, 20] and *"interlinking"* (datasets that are not properly linked to another datasets) [9, 20]. The aim is to develop mechanisms to detect and evaluate these quality criteria and also make suggestions to **enrich** (complete or add relevant data) and **curate** (repair wrong or inconsistent data) SW datasets. To achieve this goal, existent SW datasets (that we call "*seeds*") will be used to derive "*dependency rules*" (DRs) (relationships between attributes of a schema or ontology) that then will be applied on other dataset (that we call "*target*") to detect, measure and fix quality problems. In order to clarify the ideas behind our approach, we propose a simple SW dataset as use case scenario:

SW dataset about books and its authors. For each book we have ISBN, keywords, publication date, language, topic, etc. For each author we have personal information like country and city of residence, work place, organization, etc. This will be our *"target"* dataset on which we want to improve quality.

According to the problem that we want to address on the *"target"* dataset, specific issues need to be tackled:

- *Identify "dependency rules" (DR) using "seeds" datasets.* For example, a DR could be "`country` and `city` names determine the `zip-code` value for the author's residence location. Another DR could be, "Author's `country` and `country-language` determines the `language` of author's books".

- *Detect inconsistencies, wrong values or incomplete data on the "target" dataset.* For example, if we have information about `country`, `city` and `zip-code` (and the corresponding DRs that relate them), we want to check if values for these attributes are consistent between them.

- *Make suggestions to improve data completeness of the* "target" *dataset.* For example, if the language of a book is not established, we want to derive this information from those attributes that provides information about author's residence `country` and `country-language` (first, we must detect the DRs that relate these attributes).

- *Make suggestions to improve interlinking between "target" and "seeds" datasets.* For example, if the `country` and `city` values are just string values like "Argentina" and "Buenos Aires", how can we suggest links to connect the *"target"* dataset with the *"seeds"* datasets that provides URIs for "Argentina" and "Buenos Aires" resources (for example, DBPedia).

2 Relevancy

As mentioned above, the *WoD* provides big amounts of information distributed over a large number of diverse datasets but the usefulness of this data depends

on its quality. If I succeed, my PhD work will contribute in the SW data-quality research area and, more specifically, in the following related activities:

- ***Dataset enrichment and curation***. Enrichment refers to add relevant information to one dataset using data provided by other datasets. Curation refers to fix inconsistent or wrong data. Both activities are complementary.

- ***Link discovery and interlinking***. One of the key principles of LD is to relate datasets between them. Thus, once a set of potential external sources to relate with is detected (links discovery), the publisher must face with the decision of which one choose to link (interlinking). I expect to contribute in the Link Discovering [4] research area by developing methods to detect errors in links (incomplete, invalid, out-of-date, etc.) or suggest new links.

As a direct consequence of the potential contribution in the areas mentioned above, my PhD work will also contribute in the following activities:

- ***Data publishing***. Currently, there is a growing interest by organizations in publish data using SW and LD principles. One of the most important and complex aspects to consider during this task is to ensure data quality. It is therefore essential that publishers have mechanisms to detect quality problems and, eventually, have the tools to fix them.

- ***Development of applications and software over the WoD (Semantic Web applications)***. SW aplications developers will be hampered their task when trying to build intelligent software agents that automatically collect information of the WoD in order to get an integrated knowledge base for a certain purpose. Data quality is a critical aspect in an integration scenario where the readiness of information needs to ensure that it can be efficiently exploited by applications.

3 Related Work

As the amount and usage of SW data grew, several works have been addressed the data quality aspect of datasets. Zaveri et al. [20] present the results of a systematic review of approaches for assessing data quality of *LD* identifying a core set of twenty-six data quality dimensions (criteria). Vrandecic's work [16] focuses on ontology evaluation and provide a theoretical framework defining a set of eight ontology quality criteria and ontology aspects that can be evaluated as well as related methods and evaluations. Regarding data quality assessment methods (also known as framework or methodologies) for SW datasets, existent approaches can be classified into semi-automated, automated and manual [12, 19, 10, 1]. Besides, there is a lot of research performed extensively to assess the quality and report commonly occurring problems of the existing datasets [8,

9]. Regarding to *"semantic accuracy"* assessment, Fürber and Hepp [6] propose SWIQA, a quality framework that employs data quality rule templates to express quality requirements which are automatically used to identify deficient data and calculate quality scores for five quality dimensions. *"Semantic accuracy"* is one of these dimensions and authors proposed to identify semantically incorrect values through the manual definition of functional dependency rules. Another work that is inspired in the "functional dependency" concepts was done by Yu and Heflin [18]. In that work, authors propose a clustering-based approach to facilitate the detection of abnormalities in SW data by computing functional dependencies like, for example,"The language of a book is determined by the author's country". Fleischhacker et al. [5] give an approach oriented to enrich the schema of a SW dataset with property axioms (based on association rule mining) by means of statistical schema induction and also discuss other approaches related with the research areas of "LD mining" and "association rule learning" [14].

4 Research Questions

The research questions that I plan to address are:

- **What are the implications of learning "dependency rules" (DRs) from existent SW datasets?**

  To answer this question we need to understand the mechanisms to learn DRs from SW datasets and what kind of data do we need to perform this task (schemas, instance data, etc.). Besides, some related questions also need to be answered: Are these DRs dependent on both *"seeds"* and *"target"* datasets? Can these DRs be reused for apply in different datasets? How the amount-of-data of the involve datasets does affect the detection of DRs?.

- **How existent data quality assessment metrics can be used in my approach to measure "Semantic accuracy" and "Interlinking"?**

  To answer this question we need to understand the quality problems related to *"semantic accuracy"* and *"interlinking"*, examine its causes and consequences and study the existent methods to deal with them. In this sense, it is important to see the relation of these two dimensions and the potential of work with them together to improve quality. Finally, determine in which way DRs can be used to build procedures that allow us to detect a quality problem and measure certain information of the mentioned dimensions.

- **How to suggest recommendations to enrich and curate a SW dataset?**

  To answer this question we need to separate both activities. To enrich a dataset we need to know how to detect what information is missing or incomplete, to then suggest not only new relevant information but also the

way it should be used (completing a property value, adding a link, etc.). To curate a dataset, we need to detect wrong or inconsistent attribute values and suggest a way to correct them (deleting, replacing, etc.) giving new consistent values. For both scenarios, it is necessary to understand how DRs can be used with instance data of *"seeds"* datasets in order to make suggestions of new relevant data for the *"target"* dataset.

– **What are the methodologies issues to be considered when assessing the quality of SW datasets?**

To answer this question it is important to understand the limitations and drawbacks of current data quality assessment methodologies in order to determine how can we improve (or extend) them to fit with the needs of our approach.

5 Hypotheses

The main idea behind the approach of my PhD work is to improve the data-quality (regarding to *"semantic accuracy"*) of a SW datasets (that we will call *"target"* dataset) through a strategy that will use existing datasets (that we will call *"seeds"* datasets). Assuming a certain level of related "high-quality" for *"seeds"* datasets, we will use them to learn *"dependency rules"* (DRs). These DRs will be used to measure *"semantic accuracy"* (detecting wrong or inconsistent values), curate data (suggest new correct values) and enrich data the *target* dataset (complete missing values for attributes and suggest links to others datasets). This approach to improve data-quality leads to a cycle strategy: existent high-quality datasets can be used to improve quality of new and emerging datasets, and these in turn can also be used by future and even existent datasets with the same purpose. This general idea takes data-quality as a "transferable property": the quality of a SW dataset depends not only on the quality of their own data, but also on the quality of the external sources which are related to.

6 Preliminary results

Recently, we have been working on challenges related with the development of an application that integrates product reviews available as SW data (microformats, RDFa, rdf files, etc.) [13]. In this experimental work, we studied the architectures available to build SW applications and we focused on the data integration process. We also studied how quality problems affect the development of these applications when trying to consume and integrate data from heterogeneous SW datasets. We used a set of quality criteria which we divided in three categories: data-provider quality, schema quality and instance-data quality. Regarding data-provider quality we addressed "accessibility", "amount-of-data" and "timeliness". For schema quality we analyzed "coverage" and "mappings". Finally, for instance-data quality we analyzed "accuracy" (syntactic accuracy and

semantic accuracy) and "completeness" (property completeness and interlinking completeness). We got SW data about reviews using Sindice[3] and LOD-Cache[4] search engines. After analyze the retrieved data, we described common occurring errors for each criteria and their effects in the integration process. We found that most reviews have quality problems mainly related to incomplete data (reviews's text, language, rating or even a reference to the reviewed item is missing) and inconsistent values (for example, the text property has the value "This books is great" and rating property has value "0"). Although we did not propose a solution to the problems found, we noticed that many of them could be detected or even curated using information available in other datasets like DBPedia.

7 Approach

As mentioned in section 1, my PhD work will intend to address the data quality aspect of SW datasets by considering two quality dimensions: *"semantic accuracy"* and *"interlinking"*. The main idea behind this approach is to use existent SW datasets as *"seeds"* to learn DRs. Then, apply these DRs over a *"target"* dataset to detect incomplete, erroneous or inconsistent data and finally, make suggestions to curate and enrich the "*target*" dataset using instance values of the *"seeds"* datasets. In order to facilitate the understanding of the main problem, it was divided into more specific sub-problems. The first and most important task is related with how to get *"dependency rules"* (DRs) from *"seeds"* datatsets. *"Dependency rules"* concept is inspired in "data dependency" concept (well-known in relational databases domain and already used in [18] to detect abnormal data in RDF Graphs). With the DRs obtained, we will work on:

- *Detection and measurement of "Semantic accuracy" and "Interlinking"*. Although both dimensions will be treated separately, the idea is to take as reference quality evaluations performed by related work (see section 3) and adapt them to our approach (using DRs, *"seeds"* and *"target"* dataset).

- *Suggest recommendations to "enrich" and "curate" data.* Although both activities will be treated separately, the idea is to use a "Content-based Recommender System" approach [15] that uses DRs and *"seeds"* datasets to suggest new relevant data, either to complete or replace erroneous and inconsistent values.

The novel contribution of this work lies in extending current quality assessment methodologies, using existent SW datasets to get DRs and apply them to other datasets in order to detect and fix quality problems to increase data quality levels.

[3] http://sindice.com/
[4] http://lod.openlinksw.com/

8 Evaluation Plan

To facilitate the evaluation of my PhD approach, I will divide the task in the same way as Section 7. The proposed solutions for each sub-problem will be evaluated using offline experiments performing on pre-collected datasets that must meet certain requirements. For *"seeds"* datasets, it is neccessary to ensure a minimum level of data-quality, at least, for those attributes that will be considered in the DR, and will be used to make recommendations (for enrich and curate data). Both types of datasets, *"seeds"* and *"target"* must have a controlled size (in terms of amount-of-data) according to the complexity of the algorithms and hardware limitations. Attributes of interest of the involved schemas (or ontologies) must be mappeable. I pretend to evaluate my approach by comparing how many correct and useful DRs have been detected and how they can be used in detection and recommendations tasks:

- A DR is correct if the involved attributes represents a consistent relation according to *"seeds"* and *"target"* dataset (instance data and schema). We must check manually if a DR is correct (for example, having a set of predefined DRs we can test if our approach generates similar DRs).

- A DR is useful (for detection) if it can be used to detect wrong values (test *"semantic accuracy"*) or missing values (incomplete properties).

- A DR is useful (for prediction) if it can be used by recommendation algorithms to provide new attributes values and suggest potential relevant links to other datasets.

Note that the evaluation plan should include the test of algorithms used to derive DRs, detect wrong and incomplete values and generate recommendations. Traditional "precision", "recall" and other related approaches [15] can be used in these tasks.

9 Reflections

My PhD approach is based on the fact that there is a huge amount of information published following SW and LD principles and also that quality problems affects these diverse datasets to a greater or lesser extent. I also understand that data quality in SW datasets is an emerging research area of great interest with applications in domains like e-science, e-government and even e-commerce. Although many works have addressed the SW data-quality problem, most of them proposes methodologies to evaluate specific quality-criteria and report common occurring errors on a particular dataset. Only a few mention mechanisms to deal with incomplete or inconsistent data. The development of mechanisms and scalable tools to effectively solve these problems is still an open challenge.

References

1. Acosta, M., Zaveri, A., Simperl, E., Kontokostas, D., Auer, S., Lehmann, J.: Crowdsourcing linked data quality assessment. In: The Semantic Web–ISWC 2013, pp. 260–276. Springer (2013)
2. Berners-Lee, T., Hendler, J., Lassila, O., et al.: The semantic web. Scientific american 284(5), 28–37 (2001)
3. Bizer, C., Heath, T., Berners-Lee, T.: Linked data-the story so far. International journal on semantic web and information systems 5(3), 1–22 (2009)
4. Ferraram, A., Nikolov, A., Scharffe, F.: Data linking for the semantic web. Semantic Web: Ontology and Knowledge Base Enabled Tools, Services, and Applications p. 169 (2013)
5. Fleischhacker, D., Völker, J., Stuckenschmidt, H.: Mining rdf data for property axioms. In: On the Move to Meaningful Internet Systems: OTM 2012, pp. 718–735. Springer (2012)
6. Fürber, C., Hepp, M.: Swiqa - a semantic web information quality assessment framework. In: Tuunainen, V.K., Rossi, M., Nandhakumar, J. (eds.) ECIS (2011)
7. Heath, T., Bizer, C.: Linked data: Evolving the web into a global data space. Synthesis lectures on the semantic web: theory and technology 1(1), 1–136 (2011)
8. Hogan, A., Harth, A., Passant, A., Decker, S., Polleres, A.: Weaving the pedantic web (2010)
9. Hogan, A., Umbrich, J., Harth, A., Cyganiak, R., Polleres, A., Decker, S.: An empirical survey of linked data conformance. Web Semantics: Science, Services and Agents on the World Wide Web 14, 14–44 (2012)
10. Kontokostas, D., Westphal, P., Auer, S., Hellmann, S., Lehmann, J., Cornelissen, R., Zaveri, A.J.: Test-driven evaluation of linked data quality. In: Proceedings of the 23rd international conference on World Wide Web (2014), to appear
11. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., et al.: Dbpedia-a large-scale, multilingual knowledge base extracted from wikipedia. Semantic Web Journal (2013)
12. Mendes, P.N., Mühleisen, H., Bizer, C.: Sieve: linked data quality assessment and fusion. In: Proceedings of the 2012 Joint EDBT/ICDT Workshops. pp. 116–123. ACM (2012)
13. Mendoza, L., Zuccarelli, Díaz, A., Fernández, A.: The semantic web as a platform for collective intelligence (2014)
14. Nebot, V., Berlanga, R.: Mining association rules from semantic web data. In: Trends in Applied Intelligent Systems, vol. 6097, pp. 504–513. Springer Berlin Heidelberg (2010)
15. Shani, G., Gunawardana, A.: Evaluating recommendation systems. In: Recommender Systems Handbook, pp. 257–297. Springer (2011)
16. Vrandečić, D.: Ontology evaluation. Springer (2009)
17. Yu, L.: A developer's guide to the semantic Web. Springer (2011)
18. Yu, Y., Heflin, J.: Extending functional dependency to detect abnormal data in rdf graphs. In: The Semantic Web–ISWC 2011, pp. 794–809. Springer (2011)
19. Zaveri, A., Kontokostas, D., Sherif, M.A., Bühmann, L., Morsey, M., Auer, S., Lehmann, J.: User-driven quality evaluation of dbpedia. In: Proceedings of the 9th International Conference on Semantic Systems. pp. 97–104. ACM (2013)
20. Zaveri, A., Rula, A., Maurino, A., Pietrobon, R., Lehmann, J., Auer, S.: Quality assessment methodologies for linked open data. Submitted to SWJ (2012)

A Linked Data Application Development Framework (LDADF) *

Yusuf Mashood Abiodun

Faculty of Communication and Information Sciences,
Department of Computer Science, University of Ilorin, Ilorin, Nigeria.
yusufmashoodabiodun@gmail.com

Abstract: The launch of Linked Open Data Project in 2007 has resulted into making the Web a giant global data space hosting millions of linked RDF data triples on various domains in Linked Open Data Cloud. These data are now freely available for reuse. However, while some Web developers have developed notable Linked Data applications consuming and integrating different types of linked open data from the Linked Open Data Cloud to provide valuable services to people, many Web developers are yet to understand Linked Data principles and standards due to difficulties they face in learning semantic and linked data technologies. This research therefore proposes a Linked Data Application Development framework to help Web developers in overcoming challenges associated with designing and implementing linked data applications.

Key Words: linked data applications, Spring framework, development, web developers, semantic web, object-oriented frameworks

1. Problem Description

The launch of Linked Open Data Project [1] in 2007 has resulted into a giant global data space consisting of reusable linked RDF data on various domains.

However, while there are developers building Linked Data applications that are consuming data from Open Linked Data Cloud to provide useful services to people, there are still many web developers that are unable to apply the linked data principles in developing linked data applications. This is evident in the slow pace adoption of linked data among many web application developers. World Wide Web Consortium confirmed the slow pace of adoption by setting up a new Working Group called Data Activity with a mission to support data providers in publishing their data and also providing supports that will ease the development of linked data applications for application developers. In its inaugural minutes [2] of meeting held on 26th February, 2014, members of the group unanimously agreed that there is lack of awareness of Linked Data standards among the web application developers which make them to be put off when they are informed of Linked data.

Some of the contributing factors discouraging developers from adopting linked data technology include lack of necessary competencies, tools, methodological & frameworks supports, difficult in learning linked data standards, lack of guidelines, reusable libraries etc [5,8, 9]. Though some of these problems have been addressed. For instance, W3C has published a lot recommendations and best practices for publication of RDF/Linked Data.

---

* Research Supervisor: Dr. R. G. Jimoh

Also, there are now various tools and libraries that could be used in developing linked data applications. With the progresses made so far, developers are still having challenges.
While developers new to the field of semantic/linked data are facing problem of complexity in learning the technology, the earlier adopters are also having challenges in implementing of linked data applications. The authors of [8] suggested these problems could be addressed through the adoption of conventional software engineering practices with the linked data design approaches such as (i) guidelines, best practices and design patterns; (ii) software libraries; and (iii) software factories to provide a ready-made infrastructure for the developers. However, the authors only briefly described each of the design approach with identification of relevant tools, but failed to develop the desired infrastructure.

It is in view of the aforementioned above that this research seeks to adopt object-oriented software engineering practices in developing a framework for linked data application. The proposed framework will identify and integrate the existing Linked Data/Semantic libraries & APIs such as Jena, Sesame, Silk, RDR, etc. The framework will also be integrated with a popular object-oriented framework already familiar to developers. This will enable the developers to easily adopt linked applications and also reusing the beneficial features of spring framework which simplify development of enterprise applications.

2. Research Questions and Hypotheses

The goal of this research is to develop a framework that will simplify the learning curve of linked data technology for developers and thereby making it easy for them in reusing linked data design models and codes for designing and implementing linked data applications. In order to achieve the stated goal, the research will investigate the following research questions:

(i) What are the main features of Linked Data applications that distinguish them from conventional web applications?
(ii) What are the main software components that usually constitute the architectures of linked data applications?
(iii) How can the use of framework ease the learning curve for new developers coming into the world of semantic web/linked data applications?
(iv) How can the use of frameworks help developers who are earlier adopters in overcoming challenges associated with the design and implementation of linked data applications?
(v) Of what benefits will the use of framework be for developing linked data applications?

The research hypotheses are as follows:

(i) Defining the features of linked data applications will assist developers in deciding when to apply linked data technology for developing web applications.
(ii) Using of software engineering practice and frameworks will enable more developers to embrace linked data technology.
(iii) Integrating a linked data application framework with a popular object-oriented framework will simplify the design and implementation of linked data application for the developers.

(iv) Use of framework for implementing linked data applications will enable developers to deliver application within a short period through extensive reuse of analysis model, design model and codes.

(v) Integrating a linked data application with an existing object-oriented framework will enrich the features of linked data applications.

3. Relevancy:

Open Data Movements globally have been leading campaigns calling for the adoption of open data by governments of the world in order to make them more transparent and accountable to their people. Governments such as UK, US and some others have responded by launching data portals hosting thousands of datasets on various public sectors with the aim of promoting transparency and also improving their economies. Through the Open Linked Data project launched in 2007, many datasets from government data portals have been transformed and republished as RDF data in line with the principles of Linked Data. The efforts have resulted into a huge cloud of linked datasets. Web developers are expected to reuse the published datasets from the governments' portals and Linked Open Data Cloud in building Linked Data applications that will provide valuable services to people as envisaged by open data movements. However, while some developers have used the these opportunities to build linked data applications, many developers are yet to adopt and apply linked data principles and standards due to earlier stated reasons above. We hope this proposed framework will help more web developers in adopting Linked Data principles and standards to build more valuable applications for the betterment of people. In addition, the research will also contribute to the on-going efforts of truly making the Web a Web of documents and data.

4. Related Work:

A group of researchers [8] did a good work to design a conceptual architecture for linked data application and also identified all the interacting components that make up the architecture. They also advocated for the use of software engineering and design approaches i.e (1) guidelines, best practices and design patterns; (2) software libraries; and (3) software factories to provide ready-made solutions for developing linked data applications. They only briefly described each of the design approach with identification of relevant tools, but did not build a ready-made solution as envisaged.

A related work to this research is the development of a flexible integration framework for semantic 2.0 applications [9]. The research was successful in developing a framework and also integrated its with with an object-orient framework i.e Rubby on Rail framework. However, the framework is not suitable for linked data applications because it was developed as at the time the linked data technology was just evolving.

In the Linked Data book [6], the authors laid the foundation guide for the development of linked data applications by providing architectural patterns and components for implementing linked data applications. They went further to describe the main tasks and techniques needed for developing linked data applications. However, descriptions provided are difficult to comprehend and apply by web application developers that are just coming to the world of semantic web and linked data communities.

Also following the same step for Linked Data book was EUCLID project [7], a two years project funded under the EU Seventh Framework Programme for the purpose of providing comprehensive educational curriculum to the real needs of data practitioners. One of its lecture deliverables was a guide on building linked data applications. Also, the architecture of linked data application based on patterns, layers and components was described. The guide only captured information on the Linked Data application development frameworks

Another research [3] performed an empirical survey of 98 Semantic Web applications, in order to identify the most common shared components and the challenges in implementing these components. In their findings, the authors observed that though not explicitly stated, most of the applications surveyed applied principles of Component-based software engineering (CBSE) in implementing their applications. The authors only recommended the use of CBSE in building semantic applications.

There are also various life cycles that have been developed for publishing linked data on the web of data. These life cycles only focus on publishing linked data; neglecting processes to be carried out by developers to develop linked data application that can consume and manipulate data. Developers therefore found it difficult to comprehend and apply the life cycles within the context of software engineering.

5. Proposed Approach & Preliminary Results:

The research will adopt process for developing Object-Oriented Frameworks. In addition, refactoring method will also be used to capture aggregation and reusable components for linked data application development framework. The proposed approach will include the following steps:

1. **Definition of characteristics usually possessed by semantic web/linked data application:** In order to define the characteristics, we are adopting the NeOn methodological [5] process which set questionnaires for deriving semantic application characteristics based on the following three dimensions.
   i. Questionnaire about Ontologies: help the application developers to determine the characteristics of the ontologies that the application will make use of.

   ii. Questionnaire about Data: help the application developers to determine the characteristics of the data that the application will consume or manipulate and its relation with the ontologies or data schemas which data may conform to.

   iii. Questionnaire about Reasoning: help the application developers to determine the characteristics of the reasoning that the application will apply to the ontologies and data.

   The questionnaire about the data dimension will be extended to capture more requirements based on Linked data principles because they are not fully captured in NeOn.

2. **Domain Analysis for Linked Data application domain:** in oder to identify and characterize the problem domain, the research will adopt the following steps as stipulated in [10]:
   i. Outline the situation and the problem: the problem domain area is semantic web/linked data applications.
   ii. Examine existing solutions: this involves selection of semantic web/linked data applications using the defined characteristics in order to identify and extract the general functionalities common to all of them..
   iii. Identify key abstractions: applying component based software engineering process to obtain software components in line with the identified functionalities.
   iv. Identify what parts of the process the framework will perform
   v. Ask for input from clients and refine the approach

   The above steps will result into obtaining domain analysis model as presented in figure 1 below:

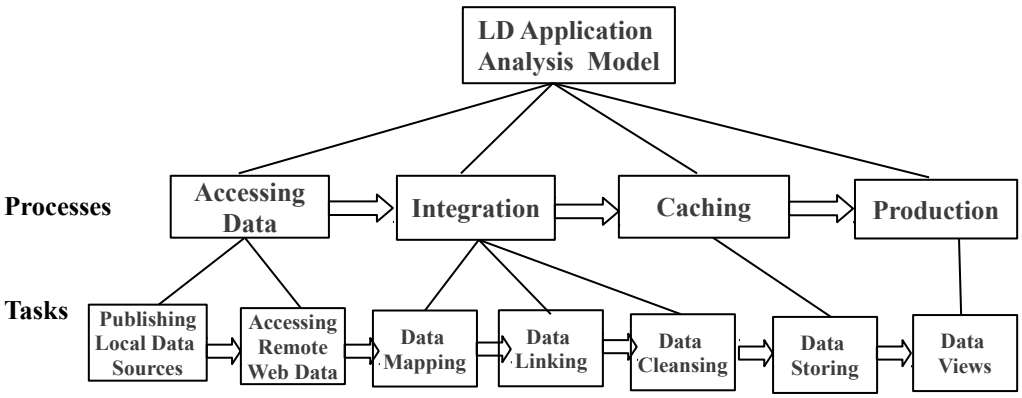
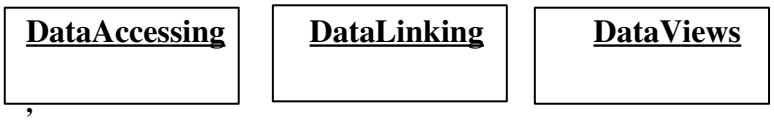


Figure 1: Analysis Model for Linked Data Application Framework

3. **Framework design:** this will be carried out following the steps illustrated in [11]. A framework design is a software design that, when implemented, provides the general and abstract functionality identified in analysis model [11].

   The two main sub-processes are:

   i. **Architectural Design:** the objective is to identify the objects/components that constitute the system and how they collaborate using the analysis model as input. Activities involved are:
      (a) **Identify Abstractions**: refining the analysis model tasks to obtain high-abstracted classes from which the clients instantiate. The derived three main classes' Names from the analysis model which client instantiate are:



,

Figure 2: The Derived Main High-Level Abstracted Classes.

for effective design and implementation of the proposed framework, the identified three above High-abstracted Classes will be classified as sub-frameworks.

(b) **Identify Generic Design Solutions:** studying the sub-frameworks and reuse design patterns to provide solutions. For instance, Strategy Design Pattern through use of Composition [12] will be reused to provide solution for DataAccessing Class/Sub-framework as shown in figure 3 below:

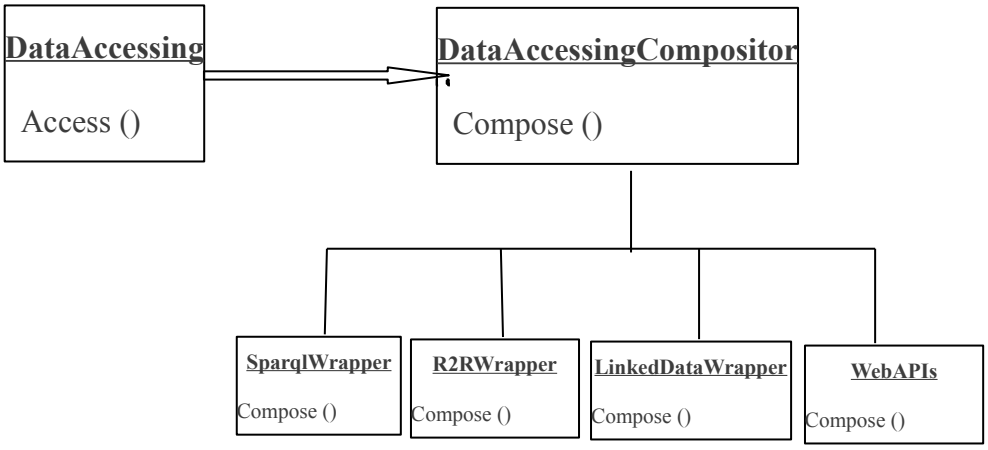


Figure 3: Use of Strategy Design Pattern for DataAccessing Sub-framework

(c) **Assigning of Classes'/Objects' Responsibilities:** these under construction

ii. **Detailed Design:** involves detailed definition of collaborations among the objects of the architectural design. Under construction.

Composition of Linked Data Application Framework with Spring Object-oriented Framework:

The proposed linked data application framework will be integrated with the Spring Object-oriented framework in order to make the linked data application framework accessible to many developers who are already familiar to Spring Framework. The integration will enable the linked data application to draw on the strengths and benefits of Spring Framework which include :

i. Spring framework is java based platform and also most of the existing RDF libraries are implemented using java programming language.
ii. Spring Core Container Infrastructure for managing the life cycle of application objects.
iii. Spring MVC Framework for achieving the separation of Concerns.
iv. Spring Aspect Programming Infrastructure

4. **Framework Implementation Phase:** the objective of the implementation phase is to implement the objects, the relationships and the collaborations identified in the design phase [11].

Core activities to be carried out according to [10], are:

i. Implement the core classes
ii. Test framework
iii. Ask client to test the framework
iv. Iterate to refine the design and add features

7. Evaluation Plan:

In order to evaluate the research hypotheses, the developed linked data application framework will be tested by the researcher using it to develop a linked data application for Nigerian National Petroleum Corporation. The agency currently publishes on its website the Oil & Gas Statistical Data (Monthly, Quarterly & Annually) on various activities ranging from upstream, midstream and downstream in PDF format. This data will be converted to RDF data and also integrated with other relevant data from the web to derive new valuable linked data for the organization. The domain expert in the industry will be involved in the development process .In addition, the framework together with its documentation will be made publicly available to developers to access and use it in developing linked data applications. Thereafter, questionnaires will be designed for developers to answer to measure research hypotheses such as shortening of development period with the use of framework, ease of learning linked data technology through the use of the framework etc.

8. Reflection:

Developing the proposed linked data application framework following the conventional software engineering and existing object-orient framework development process will enable the successful development of the proposed framework. In addition, the integration of the framework with a popular object-oriented framework, Spring framework will make linked data application technology accessible to more developers.

Acknowledgements

I sincerely wish to express my deep gratitude to my Supervisor, Dr. R. G. Jimoh for his constructive and supportive critique of my work. His mentoring role is enabling me to build my research skills.

References

1. *State of the Open Linked Data Cloud.* Retrieved April 18, 2014 from http://lod-cloud.net/state/
2. World Wide Web Consortium: *Data Activity Working Group Inaugural minute of meeting.* Retrieved April 18, 2014 from http://www.w3.org/2014/02/26-dacg-minutes.html.
3. Benjamin H, Sheila K, Conor H, & Stefan D.: Implementing Semantic Web applications. In: Proceedings of the 5th International Workshop on Semantic Web Enabled Software Engineering (SWESE 2009).
4. Leigh D., & Ian D. (2012). *Linked Data*, Retrieved on May 23, 20014 from http://patterns.dataincubator.org/book/
5. *NeOn Methodology for Building Semantic Applications (2009).* Retrieved February 9, 2014 from : http://www.neon-project.org/nw/Deliverables
6. Heath T., & Bizer C. (2011). *Linked Data Book: Evolving the Web into a Global Data Space.* Retrieved February 12, 2014 from http://linkeddatabook.com/editions/1.0/
7. *Educational Curriculum for the usage of Linked Data.* Retrieved February 12, 2014 from http://euclid-project.eu/
8. Benjamin H, Richard C, Conor H, & Stefan D: An An empirically-grounded conceptual architecture for applications on the Web of Data . Published in Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions (2012).
9. Eyal O, Armin H, Manfred H, Benjamin H, & Stefan D,: A Flexible Integration Framework for Semantic Web 2.0 Applications. *ieeexplore.ieee.org*
10. [Tal94a] Building object-oriented frameworks, Taligent, Inc., 1994
11. Niklas L. & Axel N,: Development of Object-Oriented Frameworks Authors http://www.researchgate.net/publication/245912789_Development_of_Object-Oriented_Frameworks
12. [Gam94] Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides. Design Patterns - Elements of Reusable Object-Oriented Software, Addison-Wesley, Reading, MA, 1994.

Mapping, enriching and interlinking data from heterogeneous distributed sources

Anastasia Dimou
supervised by Rik Van de Walle, Erik Mannens, and Ruben Verborgh

Ghent University iMinds Multimedia Lab
Gaston Crommenlaan 8 bus 201, 9050 Ghent, Belgium
anastasia.dimou@ugent.be

Abstract. As Linked Open Data is gaining traction, publishers incorporate more their data to the cloud. Since the whole Web of Data cannot be semantically represented though, data consumers should also be able to map any content to RDF *on-demand* to answer complicated queries by integrating information from multiple heterogeneous sources distributed over the Web or not. In both cases, the quality and integrity of the generated RDF output affects the performance of traversing and querying the Linked Open Data. Thus, well-considered and automated approaches to semantically represent and interlink, already during mapping, the domain level information of distributed heterogeneous sources is required. In this paper, we outline a plan to tackle this problem: We propose a uniform way of defining how to map and interlink data from heterogeneous sources, alternative approaches to perform the mappings and methods to assess the quality and integrity of the resulting Linked Data sets.

1 Problem Statement

Efficiently extracting and integrating information from diverse, distributed and heterogeneous sources to enable rich knowledge generation that can more accurately answer complicated queries and lead to effective decision making, remains one of the most significant challenges. Nowadays, Semantic Web enabled technologies become more mature and the RDF data model is gaining traction as a prominent solution for knowledge representation. However, only a limited amount of data is available as Linked Data, because, despite the significant number of existing tools, acquiring its RDF representation remains complicated.

Deploying the five stars of the Linked Open Data schema[1] is still the de-facto way of incorporating data to the Linked Open Data (LOD) cloud. Approaching though the stars as a set of consecutive steps and applying them to separately individual sources, disregarding possible prior definitions and links to other entities, leads in failing to reach the uppermost goal of publishing interlinked data. Manual alignment to their prior appearances is often performed by redefining their semantic representations, while links to other entities are defined after the

[1] http://5stardata.info/

data is mapped and published. Identifying, interlinking or replicating, and keeping them aligned is complicated and the situation aggravates the more data is mapped and published. Existing solutions tend to generate multiple Unique Resource Identifiers (URIs) for the same entities while duplicates can be found even within a publisher's own datasets. Hence, demand emerges for a well-considered policy regarding mapping and interlinking of data in the context of a certain knowledge domain, either to incorporate the semantically enriched data to the LOD or to answer a query on-the-fly.

So far, there is neither uniform mapping formalisation to define how to map and interlink heterogeneous distributed sources into RDF in an integrated and interoperable fashion nor complete solution that supports the whole mapping and interlinking procedure together. Apart from few domain specific tools, none of the existing solution offer the option to automatically detect the described domain and propose corresponding mapping rules. Except for the field of plain text analysis where again the main focus is on semantically annotating the text rather than describing a domain and the relationships between its entities. Moreover, there are are no means to validate and check the consistency, quality and integrity of the generated output, apart from manual user-driven controls, and no means to automate these tests and incorporate them in the mapping procedure.

2 Relevancy

The problem is directly relevant to *data publishing* and *data consumption* with an emphasis on *semantically-enabled data integration*. In the *data publishing* end of spectrum, domain level information can be integrated from a combination of heterogeneous sources and published as Linked Data, using the RDF data model. In the *data consumption* end of spectrum, the relevancy is two-fold: (i) On the one hand, the quality and integrity of the resulting RDF representation is reflected at the dataset' s consumption. (ii) On the other hand, data extracts can be mapped and interlinked *on-demand* and *on-the-fly* from different heterogeneous sources, since not all data can be represented as Linked Data. On the whole, the problem is relevant to the alignment and synchronisation of data's semantic and non-semantic representations; modifications (inserts, updates and deletions) need to be synchronised over data's semantic and non-semantic representation.

The problem is emphasized in cases of knowledge acquisition, searching or query answering that information integration is required from a combination of distributed and heterogeneous (semantic and/or non-semantic) data sources. Especially when it is taken into consideration data that cannot be easily traversed else, for instance the deep Web or large volumes of published data files. Semantic Web technologies together with the RDF data model allows to deliberately concatenate the extract of data that is relevant.

There are several stakeholders that could take advantage of such information integration enhanced with semantic annotation. Such key stakeholders are those who publish and consume large volumes of data that might be distributed and appear in heterogeneous formats. For instance, governments that publish and

consume, at the same time, Open Data, scientists that combine data from different sources and re-publish processed information or (data) journalists that need extracts of data from several sources to acquire knowledge and draw conclusions.

3 Related Work

Several solutions exist to execute mappings from different file structures and serialisations to RDF. Different mapping languages beyond R2RML were defined [6] in the case of relational databases and several implementations already exist[2]. Similarly, mapping languages were defined to support conversion from data in CSV and spreadsheets to the RDF data model. For instance, the XLWrap's mapping language [10] that converts data in various spreadsheets to RDF, the declarative OWL-centric mapping language Mapping Master's M2 [11] that converts data from spreadsheets into the Web Ontology Language (OWL), Tarql[3] that follows a querying approach and Vertere[4] that follows a *triple-oriented* approach as R2RML does too. The main drawback in the case of most *row-oriented* mapping solutions is the assumption that each row describes an entity (*entity-per-row assumption*) and each column represents a property.

A larger variety of solutions exist to map data from XML to RDF, but to the best of our knowledge, no specific languages were defined for this, apart from the W3C standardized GRDDL[5] that essentially provides the links to the algorithms (typically represented in XSLT) that maps the data to RDF. Instead, tools mostly rely on existing XML solutions, such as XSLT (e.g., Krextor [9] and AstroGrid-D[6]), XPATH (e.g., Tripliser[7]), and XQUERY (e.g., XSPARQL [1]).

In general, most of the existing tools deploy mappings from a certain source format to RDF (*per-source approaches*) and only few tools provide mappings from *different* source formats to RDF. Datalift [12], The DataTank[8], Karma[9], OpenRefine[10], RDFizers[11] and Virtuoso Sponger[12] are the most well-known. But those tools actually either employ separate *source-centric* approaches for each of the formats they support, for instance Datalift, or rely on converting data from other formats to a *master* which in most cases is *table-structured*, for instance *Karma* or *Open Refine*. Furthermore, none of them provides an approach where the mapping definitions can be detached from the implementation.

Beyond pure execution of mappings to RDF, most of the existing tools do not provide any recommendations regarding how the data should be mapped,

[2] http://www.w3.org/2001/sw/rdb2rdf/wiki/Implementations
[3] https://github.com/cygri/tarql
[4] https://github.com/knudmoeller/Vertere-RDF
[5] http://www.w3.org/TR/grddl/
[6] http://www.gac-grid.de/project-products/Software/XML2RDF.html
[7] http://daverog.github.io/tripliser/
[8] http://thedatatank.com
[9] http://www.isi.edu/integration/karma/
[10] http://openrefine.org/
[11] http://simile.mit.edu/wiki/RDFizers
[12] http://virtuoso.openlinksw.com/dataspace/doc/dav/wiki/Main/VirtSponger

namely how to model the domain described. Only Karma offers mapping recommendation, however it relies on a training algorithm that improves when several domain-relevant data sources are mapped. Among the other tools, only Open Refine supports recommendations to a certain extend, but its recommendations have the form of disambiguating named entities appearing in the LOD.

As described, existing tools are solely focused on mapping data to the RDF model, rather than interlinking the entities of the source to existing entities appearing on the Web. Only Open Refine allows to reconcile and match entities to resources published as Linked Data and Datalift which incorporates interlinking functionality but only as a subsequent step executed after the mapping is completed. Overall, till nowadays, mapping and interlinking are considered two steps that are executed consecutively. A lot of work has been done in the field of text analysis, natural language processing (NLP) and named entity recognition (NER) to identify and disambiguate entities with resources appearing in the LOD cloud. However such techniques are mainly focused on semantically annotating the text rather than modelling the domain described. Moreover these techniques are not applied in the case of (semi-)structured mappings.

Last but not least, none of the existing tools offer a complete solution that allows to refine the executed mappings based on the users' feedback, the results of data cleansing tools, reasoning over the ontologies used or studying the integrity and connectedness of the resulting dataset considering it as a graph. A summary of existing approaches for assessing data quality that could be incorporated for refining the mappings according to the result of a mapping can be found at [13]. Among the pioneer tools for RDF data cleansing are the user-driven TripleCheckMate [8] and the test-driven RDFUnit [7]. Again only Karma is capable of refining its proposed mapping according to users' intervention.

4 Research Questions

The main question in my doctoral research is:

- *How can we access and represent domain level information from distributed heterogeneous sources in an integrated and uniform way?*

On the one hand, the accessing aspect needs to be investigated:

- *How can we enable querying distributed heterogeneous sources on the Web in a uniform way?*

On the other hand, the representation aspect needs to be investigated:

- *How can we identify if entities of a source have already been assigned a URI and enrich this unique representation with new properties and links?*
- *How can we interlink newly generated resources with existing ones already during mapping considering the available domain information we have?*

And the overall result raises the following questions:

- *How can we assure that if we map some sources the domain is accurately modelled?*
- *How well the entities of the dataset are linked with each other?*
- *How well the dataset is linked with the* LOD *cloud?*

Mapping, enriching and interlinking data from heterogeneous distributed sources

5 Hypotheses

The main hypotheses related to my research are:

- *Integrated mapping and interlinking of data in heterogeneous sources generates fewer overlapping entities and models better the domain's semantics.*
- *Reusing Unique Resource Identifiers (*URI*s) leads to more robust and uniform datasets that have higher integrity and connectedness.*
- *Interlinking such datasets raises the integrity and connectedness of the whole* LOD *and improves the performance of its consumption.*
- *Not all media can be published as Linked Open Data, thus mapping extracts of multiple heterogeneous data to* RDF *might occur on demand.*

6 Approach

At this PhD, we propose a generic mapping methodology, that maps the data independently of the source structure (*source-agnostic*), puts the focus on mappings and their optimal reuse and considers interlinking already during mappings. Therefore, the initial learning costs remain limited, the potential for the custom-defined mapping's reuse augments and a richer and more meaningful interlinking is achieved. This is a prominent advancement compared to the approaches followed so far. As a result, the per-source mapping model followed so far gets surpassed, leading to contingent data integration and interlinking. Beyond the language that facilitates the mapping rules' definition and is the core of our solution, we propose a complete approach that aims to facilitate and improve the mappings definition and execution.

In our proposed approach we aim to maximize the reuse of existing unique identifiers (URIs) and rely on the links between them and the newly generated entities to achieve the interlinking of the new dataset with the LOD. The disambiguated entities are assigned the corresponding URIs and their representation is enriched with properties and relationships of the newly incorporated dataset. In contrast to the approaches followed so far, custom-generated URIs are only assigned to the entities that were not identified in the LOD cloud (not disambiguated). Based on the relationships between the newly generated entities and the disambiguated ones, the interlinking of the newly generated resources with the LOD is achieved. In order to identify such entities, we propose applying NER techniques to the sources and use them against datasets of the LOD.

Besides increasing the integrity of the dataset and reinforcing its interlinking with the LOD cloud, the whole domain needs to be modelled. Recommendations based on vocabularies used for the description and for the relationships of the disambiguated entities or other entities that are identified to model the same domain and those appearing in a vocabularies' repository, such as LOV[13], can be taken into consideration. The domain can be further refined after the execution of the mappings and the assessment of the output dataset using tools for evaluating

[13] http://lov.okfn.org

the data quality or taking into considerations the users' feedback. In these cases, the mapping rules can be adjusted to incorporate the emerging rules.

7 Preliminary results

We already defined a generic language adequate for defining rules to map heterogeneous sources into RDF in a uniform and integrated way [3]. This language is the RDF Mapping Language (RML) [14], defined as a superset of the W3C standardized mapping language R2RML. RML broadens R2RML's scope and extends its applicability to any data structure and format. RML came up as a result of our need to map heterogeneous data to RDF. Initially, R2RML was extended to map data from hierarchically structured sources e.g., XML or JSON, to RDF. Details about how we extended the row-oriented R2RML to deal with hierarchy, and other structures in general, are described in detail at our previous work [5].

Even though the language's extensibility is self-evident as RML relies on an extension over R2RML, its scalability was also proven by further extending it to map data published as HTML pages to the RDF data model. Results of the mappings from HTML to RDF using RML were presented at the Semantic Web publishing challenge of the 11th Extended Semantic Web Conference (ESWC14) [2]. At the moment, in total, RML and the prototype processor support, but are not limited, mappings from data in CSV, XML, JSON and HTML to the RDF data model.

A prototype processor[15] was designed and implemented as a *proof-of-concept* to accompany the RML mapping language. As RML extends R2RML, the processor is implemented using an existing open-source R2RML processor[16]. The RML processor was designed to have a modular architecture where the extraction and mapping modules are independently executed and the extraction module can be instantiated depending on the possible inputs. Short discussion regarding alternative approaches for processors supporting RML were discussed at [5].

Finally, some preliminary work on mapping rules' refinements by incorporating data consumers' feedback was presented at [4]. We showed how provenance generated during mapping can be used later on to identify the mapping rules that should be adjusted to incorporate data consumers' feedback.

8 Evaluation plan

There are different aspects of the proposed solution which need to be assessed and we are aiming to evaluate: the RML mapping language itself, the semantic annotations and the entities interlinking, the quality and integrity of the resulting dataset and the performance of the mapping execution.

– the language's potential in regard to (i) the range of input sources supported and their possible combinations for providing integrated mappings, namely

[14] http://rml.io
[15] https://github.com/mmlab/RMLProcessor
[16] https://github.com/antidot/db2triples

the language's *scalability* and *extensibility*; (ii) the language's *expressivity*, namely the coverage of possible alternative mapping rules, mainly in comparison to other languages (or approaches) and (iii) last, how *reusable* and *interoperable* the mapping descriptions are.
- the *validity*, *consistency* and *relevance* (especially when the domain is modelled according to automated recommendations) of the vocabularies used by the mapping rules to describe the domain knowledge.
- the *quality* of the output. To achieve this, both automated solutions assessing data quality and domain experts will be used to evaluate the resulting dataset in regard to the identified or generated entities, the provided semantic annotations, the interlinking and the overall modelling of the domain.
- the *accuracy* and the *precision* and *recall* of the retrieved, identified and enriched entities in conjunction with the *confidence* for the interlinked entities.
- the *integrity* of the resulting dataset and the overall *analysis* of the output's datasets in respect to its graph-based representation, for instance in and out degree, its connectivity, its density, bridges, paths etc.
- the *impact* of the resulting dataset's structure and interlinking in respect to its subsequent *consumption*. To be more precise, how *traversing* and *querying* the dataset is affected by the choices taken while modelling the knowledge domain. In the case of querying, we aim to examine both the *complexity* of the queries definition and the *time* and *overload* to execute them.
- finally, while the *performance* is important to verify that the mappings can be executed in reasonable time, the performance of an RML processor is not the main focus of this work. However, the two fundamental ways of executing the mappings (mapping-driven or data-driven) will be evaluated and compared to identify best use-cases. The execution planning of the mapping rules though is more interesting and will be deeper investigated and evaluated.

9 Reflections

The main difference of our approach compared to existing works on mapping data is that we (i) introduce the idea of a uniform way of dealing with the mapping of heterogeneous sources and (ii) introduce the aspect of interlinking while we perform the mapping of data to the RDF data model. We approach the mapping from a domain modelling perspective where the data is either incorporated to a partially described domain or is mapped combined, forming their own domain. This way, we achieve generating datasets with higher integrity that are already interlinked among each other and with the LOD and thus we reduce the effort for subsequent interlinking of resources and offer better conditions for their subsequent consumption.

Acknowledgement

The research described in this paper is funded by Ghent University, the Flemish Department of Economy, Science and Innovation (EWI), the Institute for the

Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research-Flanders (FWO-Flanders), and the European Union.

References

1. S. Bischof, S. Decker, T. Krennwallner, N. Lopes, and A. Polleres. Mapping between RDF and XML with XSPARQL. *Journal on Data Semantics*, 1(3):147–185, 2012.
2. A. Dimou, M. Vander Sande, P. Colpaert, L. De Vocht, R. Verborgh, E. Mannens, and R. Van de Walle. Extraction and semantic annotation of workshop proceedings in HTML using RML. In *Semantic Publishing Challenge of the 11th Extended Semantic Web Conference*, May 2014.
3. A. Dimou, M. Vander Sande, P. Colpaert, R. Verborgh, E. Mannens, and R. Van de Walle. RML: A generic language for integrated RDF mappings of heterogeneous data. In *Proceedings of the 7th Workshop on Linked Data on the Web*, Apr. 2014.
4. A. Dimou, M. Vander Sande, T. De Nies, R. Verborgh, E. Mannens, and R. Van de Walle. RDF mapping rules refinements according to data consumers feedback. In *2nd International World Wide Web Conference, Poster Track Proceedings*, 2014.
5. A. Dimou, M. Vander Sande, J. Slepicka, P. Szekely, E. Mannens, C. Knoblock, and R. Van de Walle. Mapping hierarchical sources into RDF using the RML mapping language. In *Proceedings of the 8th IEEE International Conference on Semantic Computing*, 2014.
6. M. Hert, G. Reif, and H. C. Gall. A comparison of RDB-to-RDF mapping languages. In *Proceedings of the 7th International Conference on Semantic Systems*, I-Semantics '11, pages 25–32. ACM, 2011.
7. D. Kontokostas, P. Westphal, S. Auer, S. Hellmann, J. Lehmann, R. Cornelissen, and A. Zaveri. Test-driven evaluation of linked data quality. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, pages 747–758. International World Wide Web Conferences Steering Committee, 2014.
8. D. Kontokostas, A. Zaveri, S. Auer, and J. Lehmann. Triplecheckmate: A tool for crowdsourcing the quality assessment of linked data. In *Knowledge Engineering and the Semantic Web*, volume 394 of *Communications in Computer and Information Science*, pages 265–272. Springer Berlin Heidelberg, 2013.
9. C. Lange. Krextor - an extensible framework for contributing content math to the Web of Data. In *Proceedings of the 18th Calculemus and 10th international conference on Intelligent computer mathematics*, MKM'11. Springer-Verlag, 2011.
10. A. Langegger and W. Wöß. XLWrap – Querying and Integrating Arbitrary Spreadsheets with SPARQL. In *Proceedings of the 8th International Semantic Web Conference*, ISWC '09, pages 359–374. Springer-Verlag, 2009.
11. M. J. O'Connor, C. Halaschek-Wiener, and M. A. Musen. Mapping Master: a flexible approach for mapping spreadsheets to OWL. In *Proceedings of the 9th International Semantic Web Conference on The Semantic Web - Volume Part II*, ISWC'10, pages 194–208. Springer-Verlag, 2010.
12. F. Scharffe, G. Atemezing, R. Troncy, F. Gandon, S. Villata, B. Bucher, F. Hamdi, L. Bihanic, G. Képéklian, F. Cotton, J. Euzenat, Z. Fan, P.-Y. Vandenbussche, and B. Vatant. Enabling Linked Data publication with the Datalift platform. In *Proc. AAAI workshop on semantic cities*, 2012.
13. A. Zaveri, A. Rula, A. Maurino, R. Pietrobon, J. Lehmann, and S. Auer. Quality assessment for linked open data: A survey. Submitted to the Semantic Web Journal., 2013.