RENDER
FP7-ICT-2009-5
Contract no.: 257790
www.render-project.eu

RENDER

Deliverable D1.2.2

Final Data Integration

Editor:	Mariana Damova, Ontotext
Author(s):	Mariana Damova, Ontotext; Delia Rusu, JSI
Deliverable Nature:	Report (R)
Dissemination Level: (Confidentiality)	Public (PU)
Contractual Delivery Date:	March 2013
Actual Delivery Date:	March 2013
Suggested Readers:	Data integrators, users of RENDER data layer
Version:	1.1
Keywords:	data integration, reason-able view, diversity management, reasoning, RENDER data layer, RENDER data infrastructure, basic data layer, Reference Knowledge Stack, secondary data layer, KDO, PROTON, DBpedia, Freebase, Geonames, SIOC, DC, DMOZ, Schema.org, nested repositories

Disclaimer

Full Project Title:	RENDER – Reflecting Knowledge Diversity
Short Project Title:	RENDER
Number and Title of Work package:	WP1 Data collection and management
Document Title:	D1.2.2 - Final data integration
Editor (Name, Affiliation)	Mariana Damova, Ontotext
Work package Leader (Name, affiliation)	Maurice Grinberg, Ontotext

Copyright notice

Executive Summary

This deliverable reports on the constitution of the final RENDER data layer. It is based on the final data collection described in deliverable 1.1.3. The final RENDER data layer is contained in OWLIM [1] semantic repository, which plays the role of central data integration repository in RENDER architecture. The RENDER data layer is split into basic data layer and secondary data layer. The final basic data layer contains the Reference Knowledge Stack, described in deliverables 1.1.1 and 1.1.2. The new version of PROTON, e.g. PROTON 4.0 [20], is released and used in the Reference Knowledge Stack of this final basic RENDER data layer. The reduction of the number of datasets for this release is motivated by the requirements of the use cases, making the remaining 6 LOD [17] datasets unnecessary as part of it. The secondary data layer makes use of two nested repositories, which host the processed by Enrycher according to the RENDER data models, based on KDO [14], SIOC [23], DC [8], and PROTON described in deliverable 1.1.3, Google news and Twitter data respectively. The three layers have the APIs, described in deliverable 1.3.1, which allow access and modification of the data by humans by GUIs, and programmatically by standard SESAME RESTful APIs for SPARQL endpoints [26]. The final data integration consists in production of the basic data layer and in automated upload of the secondary data layers, via technical integration between Enrycher and the RENDER backend, e.g. the RENDER data layer through its APIs. This deliverable, closely related to deliverable 1.1.3, describes the results of the final data integration of RENDER data layers.

Table of Contents

Abbreviations

KDO	Knowledge Discovery Ontology
LOD	Linked Open Data
RDF	Resource Description Framework
TID	Telefónica Investigación y Desarrollo
UI	User Interface
URI	Unique Resource Identifier
OWL	Web Ontology Language

1 Introduction

The final data integration of RENDER provides the final version of RENDER data layer, which is the central data pool and integrated infrastructure for RENDER use cases[1]. The final data integration follows the first version of RENDER data layer, presented in deliverables 1.1.1, and 1.1.2 available at render.ontotext.com, and integrates the data presented in the final data collection, described in deliverable 1.1.3. It must be pointed out that the delivered data layer reduces the number of datasets covered because of the needs of the use cases, and the constraints triggered from the quality of the LOD data, to a certain degree the loading times as well. A data curation methodology has been applied in this final data integration effort, which allows distinction from the data. Section 3 discusses the Integration of the Basic Data Layer, Section 4 discusses the Integration of the Secondary Data Layer, and Section 5 explains the integration between Enrycher service and RENDER backend, e.g. OWLIM repository and RENDER data layer, which allows automated population of the secondary layers with textual data processed by Enrycher.

[1] Thanks are due to Svetoslav Petrov, Dimitar Manov, Kiril Simov for contributing to the technical work related to the final data integration.

2 RENDER Data Layer Integration Architecture

RENDER data layer is loaded in OWLIM semantic repository, OWLIM-SE, version 5.3. It consists of a basic data layer and of a secondary data layer. The basic data layer is a reason-able view of the web of data, comprising a segment from LOD with inference according to OWL-Horst. It carries FactForge data model, which is referred in RENDER as Reference Knowledge Stack, enabling access to the heterogeneous generic data via a unification ontology, PROTON. The basic data layer has access GUI for human user, and access APIs for programmatic access, which is compliant with SESAME RESTful APIs for SPARQL end points [26]. The secondary data layer of RENDER is built on top of the basic data layer. It is hosted by a mechanism of data sharing for OWLIM [1], developed within RENDER, nested repositories, described in deliverable 1.3.1. The RENDER secondary data layer makes use of two nested repositories for the two distinct datasets, e.g. the Google News cluster data and the Twitter data. Each nested repository is set up on top of RENDER basic data layer, sharing its knowledge, but being separate with each other with respect to their specific data. Each “secondary” dataset has a data model, which is based on the following ontologies: SIOC [23], DC [8], KDO [14], PROTON [20] and the RDF [19] produced from Enrycher [9] text processing service complies these two models. Enrycher text processing service is also supplied with a plug-in, which allows automatic connection to the RENDER backend in the appropriate nested repository and upload of the data through the RESTful APIs. Similarly to the RENDER basic data layer, the two nested repositories have access GUI, based on Forest framework, developed and maintained at Ontotext, and access APIs, based on SESAME RESTful APIs for SPARQL end points[2].

The component architecture of RENDER data layer, which is applied in the final data integration procedure is shown in Figure 1: RENDER Data Layer Integration Architecture.

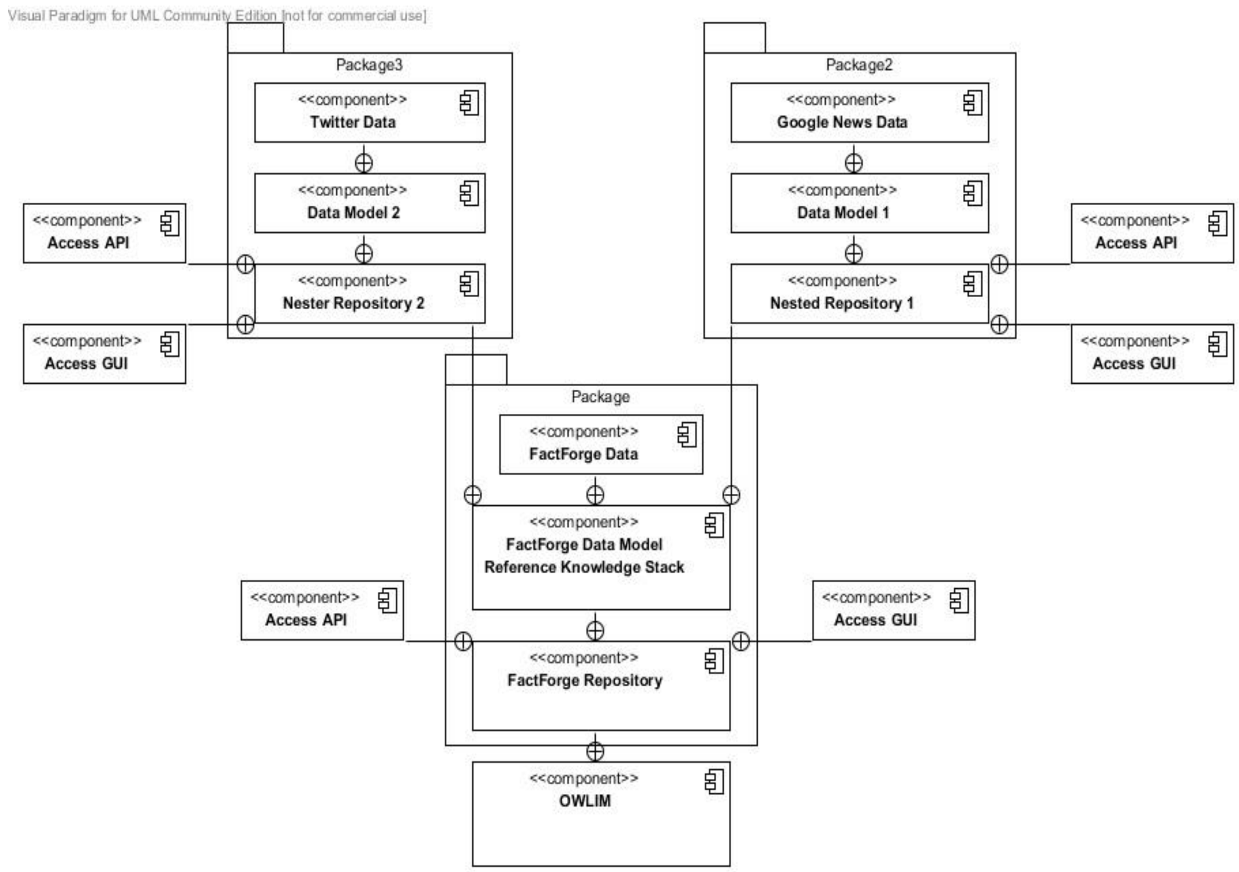


Figure 1: RENDER Data Layer Integration Architecture

The following sections 3, 4 and 5 describe the final data integration process step by step.

[2] SESAME RESTful APIs for SPARQL end points implementation. http://www.w3.org/wiki/SparqlImplementations

3 Final Integration of RENDER Basic Data Layer

3.1 Datasets

The datasets of RENDER basic data layer, FactForge, are organized in three categories:

- General knowledge
- Domain specific knowledge
    - Geography

The general knowledge datasets of RENDER basic data layer, FactForge, are:

- **DBPedia 3.8**

DBPedia 3.8 is based on Wikipedia dumps from late May/early June 2012. DBPedia ontology has 359 classes (40 more than DBPedia 3.7). Only the English version of DBpedia 3.8 is part of the final data integration dataset.

- **Freebase**

Freebase (http://freebase.com) is a large collaborative knowledge base, an online collection of structured data harvested from many sources, including individual wiki contribution. Freebase has about 20 million topics or entities and no defined ontology. The entities described in this knowledge base are in structured predicate names, which reflect a hidden class hierarchy.
The latest RDF data dump of Freebase is from 04.12.2012. Its compressed size is of 7.5 GB, and uncompressed size is of 49.5 GB. It contains approximately 736 million triples.

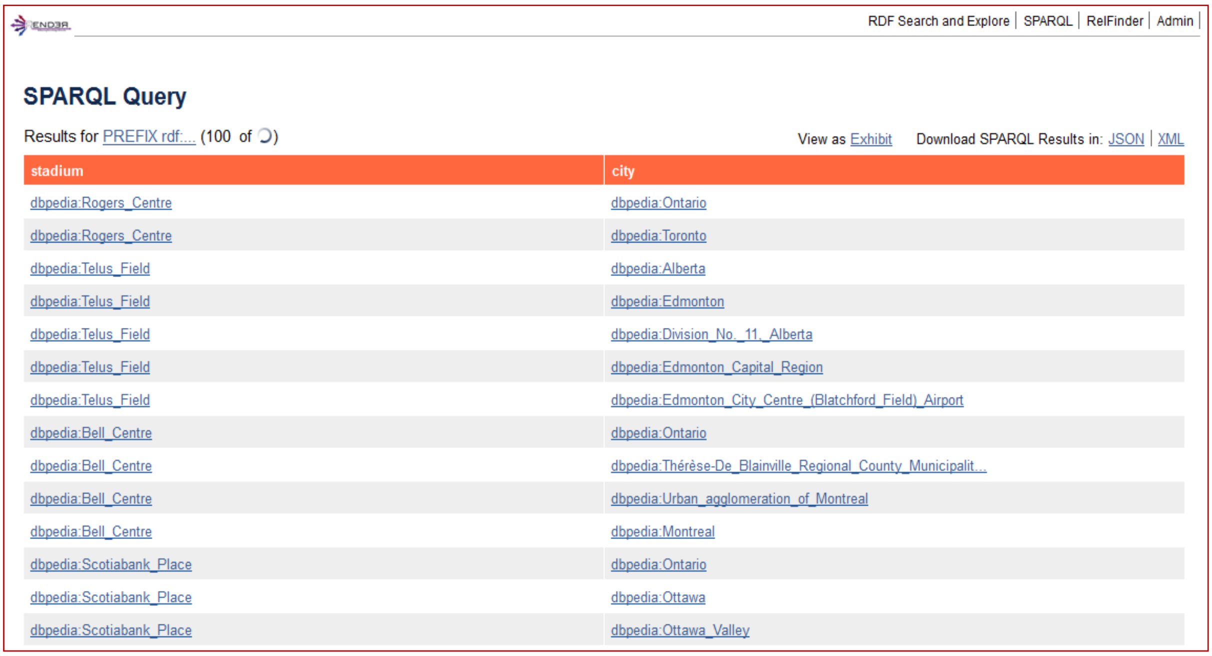
- **Geonames**

Geonames is a geographical knowledge base, containing over 8 million placenames, their geolocations and other geographical and demographic information. Its last RDF version is from October 2012.

3.2 Reference Knowledge Stack

The concept of a Reference Knowledge Stack has been introduced in the RENDER project. It refers to defining a unification ontology, which is linked to the ontologies of FactForge datasets, and serves as a common entry point to FactForge's data, making querying, accessing, managing and navigating this wealth of data more efficient and manageable. The Reference Knowledge Stack includes a unification ontology, PROTON, mapped to the schemata of DBpedia 3.8, Freebase from October 2012, and Geonames 2.0.2, which are not loaded into the final RENDER data layer. This makes the size of the repository optimal, and reduces the loading times, as the inference that would have been produced from the schemata of the LOD datasets separately is not produced. Except for the mappings, PROTON 4.0 has extended the previous version of PROTON to cover the schemata of DBpedia 3.8, Freebase, October 2011, Geonames 2.0.1. PROTON was moved to a new namespace, e.g. http://www.ontotext.com/proton/protontop#, http://www.ontotext.com/proton/protonext#. The information about the ontology is accessible at http://www.ontotext.com/proton-ontology. The reference layer has been extended in the final data integration of RENDER with mappings between PROTON and Schema.org [24]. This allows querying render.ontotext.com with predicates from Schema.org as well. For instance, the query about stadiums in Canada, can be written with Schema.org Class for Stadium, e.g. sch:StadiumOrArena, substituting PROTON's pext:Stadium, which covers DBpedia's and Geonames' descriptions, and returns the results shown in Figure 2: Results for Query "Stadiums in Canada" with the stadiums given in the left-hand side column, and their locations in the right-hand side column.


```
select  ?stadium ?city where {
          ?stadium rdf:type sch:StadiumOrArena .
          ?stadium ptop:locatedIn ?city .
          ?city ptop:subRegionOf dbpedia:Canada
  }
```


RDF Search and Explore | SPARQL | RelFinder | Admin |

SPARQL Query

Results for PREFIX rdf:... (100 of)

View as Exhibit Download SPARQL Results in: JSON | XML

stadium	city
dbpedia:Rogers_Centre	dbpedia:Ontario
dbpedia:Rogers_Centre	dbpedia:Toronto
dbpedia:Telus_Field	dbpedia:Alberta
dbpedia:Telus_Field	dbpedia:Edmonton
dbpedia:Telus_Field	dbpedia:Division_No._11,_Alberta
dbpedia:Telus_Field	dbpedia:Edmonton_Capital_Region
dbpedia:Telus_Field	dbpedia:Edmonton_City_Centre_(Blatchford_Field)_Airport
dbpedia:Bell_Centre	dbpedia:Ontario
dbpedia:Bell_Centre	dbpedia:Thérèse-De_Blainville_Regional_County_Municipalit...
dbpedia:Bell_Centre	dbpedia:Urban_agglomeration_of_Montreal
dbpedia:Bell_Centre	dbpedia:Montreal
dbpedia:Scotiabank_Place	dbpedia:Ontario
dbpedia:Scotiabank_Place	dbpedia:Ottawa
dbpedia:Scotiabank_Place	dbpedia:Ottawa_Valley

Figure 2: Results for Query "Stadiums in Canada"

This level of data integration makes the RENDER basic data layer interoperable with Google, Yahoo, and Bing.

3.3 Loading Statistics

The RENDER basic data layer has been loaded into OWLIM 5.3 with full materialization based on OWL-Horst rule set, extended with specific inference rules, that cover the structural discrepancies between the conceptual models of PROTON and the LOD schemata and ensure that the mapping is effective. The loading statistics are as follows:

Statistics after Loading DBpedia 3.8, Freebase, Geonames, PROTON 3.0, and mappings	
NumberOfStatements:	2,005,343,932
NumberOfExplicitStatements:	1,366,801,742
NumberOfEntities:	383,113,371

Table 1: Statistics after Loading DBpedia 3.8, Freebase, Geonames, PROTON 3.0, and mappings

Statistics after Loading owl:sameAs statements of DBpedia 3.8, Geonames and Freebase	
NumberOfStatements:	2,336,280,003 (+330,936,071)
NumberOfExplicitStatements:	1,370,500,605 (+3,698,863)
NumberOfEntities:	383,113,686 (+315)

Table 2: Statistics after Loading owl:sameAs statements of DBpedia 3.8, Geonames and Freebase

The constitution of RENDER basic data layer follows the methodology of creating FactForge[3], a free service maintained by Ontotext. This includes post-processing steps, providing indices for full text search, preferredLabel, RDF-Rank, generation of snipets for each entity, geo-index for retrieving geographical areas. The overall number of triples in OWLIM repository increases by about 400 mnl statements, as shown in table 3.

Statistics after Loading the results of the post-processing steps	
NumberOfStatements:	2,722,092,195
NumberOfExplicitStatements:	1,542,698,037
NumberOfEntities:	411,707,334

Table 3: Statistics after Loading the results of the post-processing steps

The loading on a server with the following characteristics: 144G Memory, RAID O array of 4 SSD drives, 2 processors XEON X5680, 3.3 GHz took 2 weeks, where most of the time has been spent on the post-processing steps, their calculation and their loading.

RENDER basic data layer can be accessed at http://render.ontotext.com, and via SESAME RESTFul APIs at http://render.ontotext.com/repositories/owlim.

The size of the dataset of the final data integration repository is affected by the number of datasets in it, but also by the undertaken data curation steps, described in the following section 3.4.

3.4 Basic Data Layer Curation

It is well known that LOD data contain a lot of noise and errors, which can become a big burden and generate a lot of wrong statements when inference is taken into consideration. The final data integration part of the RENDER basic data layer undertook a systemic approach to identify erroneous statements in the data, and removing them. Using the OWL 2 RL pie rule, described in deliverable 1.1.3 was not made use of in this data load, as a lighter reasoning rule set, OWL-Horst, has been adopted as the most scalable one. That is why a series of queries was crafted to help spot the reasons underlying the inference errors, and consequently remove them. For instance, queries were defined that carry different conceptualization in PROTON and in DBpedia, and wrongly inferred statements have been removed. The following query looks for instances that are either Activities or Sports.

```
PREFIX pext: <http://www.ontotext.com/proton/protonext#>
PREFIX onto: <http://www.ontotext.com/>

select distinct ?s
from onto:disable-sameAs {
?s a ?t1 .
?s a ?t2 .
filter(?t1 != ?t2)
filter(?t1 = pext:Activity || ?t1 = pext:Sport)
filter(?t2 = pext:Activity || ?t2 = pext:Sport)
}
```

This helps identify wrongly classified instances.

Another reason for producing wrong inference are plain mistakes in the explicit statements of the LOD datasets. For instance, we encountered an explicit statement that Germany is a Book in DBpedia, its French version. This fact lead to limiting the final RENDER basic data layer to the English DBpedia only, as a guarantee to reduce the statement pool from which errors would possibly appear.

[3] http://factforge.net; http://www.ontotext.com/owlim

Additional optimization of the data has been produced by removing LOD based triples describing higher level types of instances. For example, for all Sports of DBpedia the triples, stating that they are activities were removed. Further, to ensure that only correct types of domains and ranges of properties were present in the basic data layer, all statements that did not have the proper types were deleted using SPARQL queries. The identification of the correct types took place also by running SPARQL queries of the form shown in the beginning of the subsection. DBpedia has introduced consistently for each statement its Schema.org type. This causes interference with PROTON and the mappings with Schema.org, that is why all DBpedia statements with Schema.org types were removed from the repository.

Lastly, there have been incompatible, or rather consistently ambiguous concepts, such as University, which is classified in DBpedia as Organization and as Location. This systemic ambiguity has been addressed in RENDER basic data layer by selecting one of the two types, and removing the other.

4 Final Data Integration of Render Secondary Data Layers

The RENDER secondary data layer consists of two datasets, loaded in two nested repositories, one for each dataset. These are Twitter data, and Google News Cluster data. The detailed description of the datasets and the ontologies that are part of RENDER secondary data layer can be found in deliverable 1.1.3. Here they are repeated for clarity.

4.1 Ontologies

The RENDER secondary data layer includes additional ontologies, e.g.

- **KDO (Knowledge Diversity Ontology)** [14]

KDO is developed within RENDER project and presents RENDERs view about diversity. It models statements, sentiments and opinions.

- **SIOC (Semantically Interlinked Online Communities)** [23]

SIOC is an ontology allowing to represent and integrate information on online communities.

Except for the additional ontologies, RENDER secondary data layer involves two categories of schemata, e.g.

- **DMOZ** [7]

DMOZ is a free dictionary that categorizes segments of the world to be used as media informational units, for instance, business, sports, computers, health and medicine, etc.

- **Telefonica categories** [27]

Telefonica categories are provided by Telefonica, a partner in RENDER project, and present company internal classification of their products and services.

4.2 News

The News dataset provided by JSI is a collection of news articles crawled from the Google News web site in the period of approximately two months. The collection contains about 23,500 articles clustered (by Google) into stories 10-150 articles in size with median at 30. The articles are stripped of HTML markup and chrome (navigation, headers, footers etc.), then enriched with named entity detection and disambiguation algorithms and with full constituency parse trees. The total size of the enriched dataset is 1.5 GB (sqlite table, uncompressed). All the articles in this collection are in English, although they are gathered from publishers located all over the world. The average article is 550 words in length.

Google news articles were analyzed according to the RDF model, developed by Ontotext, c.f. deliverable 1.1.3. It includes diversity information, topic information, reference to an entity from the LOD cloud, document identification.

This nested repository is available at http://rendernews.ontotext.com and via SESAME RESTFul APIs http://rendertweets.ontotext.com/repositories/news.

The overall dataset and ontologies have been loaded for about 4 days.

This is the size of the rendernews nested repository:

Statistics after Google News Cluster RDF processed by Enrycher and the Render Secondary Layer Ontologies	
NumberOfStatements:	16,138,797
NumberOfExplicitStatements:	9,173,828
NumberOfEntities:	413,818,078

Table 4: Statistics after Google News Cluster RDF processed by Enrycher and the Render Secondary Layer Ontologies

4.3 Twits

The Twitter dataset collected by Telefonica is processed using the Diversity Mining Services [2] (Enrycher) and its RDF representation is stored in the OWLIM repositories.
The processed data comprises annotated tweets from:

- September 2010 [9 days]
- October 2010 [6 days]
- January 2011 [8 days]
- Monthly datasets June, July, August 2012

The size of a monthly unprocessed dataset is on average 5-6GB, smaller datasets can reach around 2GB. With respect to the number of twits per month, the dataset from July 2012 contained 26 million twits. After processing and enriching, the resulting RDF dataset would be around 85GB uncompressed (given an input dataset of 4.5GB). Regarding processing time, approximately 90% is required for enriching the dataset (performing topic detection, entity extraction and resolution, sentiment analysis) and the remaining 10% is required for RDF export.

Twitter data were analyzed according to the RDF model, developed by Ontotext, c.f. deliverable 1.1.3. It includes diversity information, topic information, reference to an entity from the LOD cloud, document and author account identification.

This nested repository is available at: http://rendertweets.ontotext.com, and via SESAME RESTFul API at http://rendertweets.ontotext.com/repositories/tweets

An overall of 26 GB of twitter data in RDF have been loaded for one week.

This is the size of the rendertweets nested repository after the first round of loading the data from 2010 and from June 2012 :

Statistics after Twitter data RDF processed by Enrycher and the Render Secondary Layer Ontologies	
NumberOfStatements:	1,154,042,785
NumberOfExplicitStatements:	938,277,129
NumberOfEntities:	619,895,316

Table 5. Statistics after Twitter data RDF processed by Enrycher and the Render Secondary Layer Ontologies

5 Integration with Enrycher

The component for loading RDF data in the OWLIM repositories was written in Java and is using the SESAME API[4] [1]. We start by creating the HTTP repository:

```
HTTPRepository rep = new HTTPRepository("…");
```
Next, we create a RepositoryConnection object:
```
RepositoryConnection con = rep.getConnection();
```
The RDF data is parsed via the Rio parser:
```
RDFParser parser = Rio.createParser(RDFFormat.RDFXML);
```
The repository supports inserting the entire RDF data, via the `add` method of the RepositoryConnection class, or by adding and then committing n-triples (subject-predicate-object RDF statements).

Experiments

We have experimented with loading one month of Telefonica Twitter collected data, namely June 2012. This data has been processed with the Diversity Mining Toolkit pipeline, yielding what we refer to as enriched Twitter data. We loaded approximately 530MB (compressed) of enriched Twitter data (approx. 10 000 enriched tweets in RDF).

We took two approaches to loading the data:

1. Committing few statements at a time
2. Committing in chunks of statements

1. Committing few statements at a time

We loaded approx. 50-100 statements, corresponding to one single tweet. This simulates the stream setting, where each tweet in the stream would be processed independently and then loaded in the repository. We can load approx. 400 enriched tweets in 1h.

2. Committing in chunks of statements

We committed chunks of statements, each chunk consisting of 100,000 statements. The loading took roughly 3.5 min for 100,000 statements, and 2.5 hours for 530 MB (approx. 4,300,000 statements).

[4] http://www.openrdf.org/doc/sesame/users/ch07.html, retrieved 21.03.2013

6 Conclusion

This deliverable presented the principles of building the final data integration, and described the process of producing the final RENDER data layer, including statistics and applied specific quality assurance methodologies. RENDER basic data layer for the final data integration consists of the Reference Knowledge Stack only, and just the English DBpedia, whereas previous versions contain the entire FactForge, e.g. 9 LOD datasets, including the whole DBpedia. This is due to the requirements of the use cases, and the established need to create and apply a straightforward data curation methodology, which will allow to guarantee the quality and the conceptual salience of the data to be employed by the use cases. On the other hand, the final data integration includes a second reference ontology for the reference layer of the RENDER basic data layer, e.g. Schema.org, which makes it interoperable with Google, Yahoo, and Bing. The secondary data layer has been extended with Twitter data from several full months of 2012 in English and Spanish, which are supplied with language information. This gives the possibility for the use cases to select information per language as well. Successful experiments have been conducted to automatically load the data outputted from the Enrycher service, which allowed full automation of the production of RENDER secondary data layer. The OWLIM feature nested repository requires human intervention for its setup by a system administrator.

The final data integration provides the data layer, which support RENDER use cases. The reason-able view of the web of data, supplied with two nested repositories, is a unique showcase of large scale data and reasoning with them, as the available for retrieval data are between 30 and 40 % more than the explicitly introduced into the repository data.

RENDER data layer has three access points for the three repositories, and will be used by Telefonica, Google and UIBK’s applications for their corresponding use cases. It will remain supported for one year after the end of the project.

References

1. Bishop, B., Kiryakov, A., Ognyanoff, D., Peikov, I., Tashev, Z., & Velkov, R. (2011). OWLIM: A family of scalable semantic repositories. (Hitzler, P., Ed.) *Semantic Web Journal. Special Issue: real-World Applications of OWL*.

2. Caminero, J. (ed). 2012. Initial version of diversity information extensions for Telefónica tools. RENDER Deliverable D5.3.3.

3. Damova, M., Kiryakov, A., Grinberg, M., Bergman, M., Giasson, F., Simov, K.. Creation and Integration of Reference Ontologies for Efficient LOD Management. In: Semi-Automatic Ontology Development: Processes and Resources, IGI Global, USA, Armando Stellato and Maria Teresa Pazienza (Eds.) 2012.

4. Damova, M., Kiryakov, A., Simov, K.., Petrov, Sv. Mapping the Central LOD Ontologies to PROTON Upper-Level Ontology In Proceedings of Ontology Matching workshop, International Semantic Web Conference, Shanghai, China, November 2010.

5. *Datalog*. (2012) http://en.wikipedia.org/wiki/Datalog.

6. *DBpedia.* (2011). Retrieved from Structured information from Wikipedia: http://DBpedia.org.

7. *DMOZ.* (2012). http://www.deemoz.org/.

8. *DC (Dublin Core)*. (2011). Retrieved 2011, from Dublin Core Metadata Element Set: http://dublincore.org/documents/dces/.

9. *Enrycher Service of JSI (2012).* http://ailab.ijs.si/tools/enrycher/.

10. *FactForge*. (2011). Retrieved from A Reason-able View to the Web of Data: http://factforge.net, http://www.ontotext.com/factforge.

11. *FOAF*. http://www.foaf-project.org/

12. *Freebase*. (2011). Retrieved from http://www.freebase.com/.

13. *Geonames*. (2011). Retrieved 2011, from A geographical database: http://www.geonames.org.

14. *KDO*. (2012) http://lov.okfn.org/dataset/lov/details/vocabulary_kdo.html.

15. Kiryakov, A., & Momtchev, V. (2009, June). Two Reason-able Views to the Web of Linked Data. *Paper presented at the Semantic Technology Conference* . San Jose, USA.

16. Kiryakov, A., Ognyanoff, D., Velkov, R., Tashev, Z., & Peikov, I. (2009). LDSR: Materialized Reason-able View to the Web of Linked Data. In R. H. Patel-Schneider (Ed.), *Proceedings of OWLED 2009* . Chantilly, USA.

17. *LOD*. http://linkeddata.org/

18. *OWLIM*. (2011). Retrieved from http://www.ontotext.com/owlim.

19. *RDF (Resource Description Framework).* (2012). http://www.w3.org/RDF/

20. *PROTON 4.0.* (2012). PROTON 4.0 Documentation. Sofia, Bulgaria. http://www.ontotext.com/proton-ontology.

21. Prud'hommeaux, E., Seaborne, A. (2008). *SPARQL Query Language for RDF, W3C Recommendation 15 January 2008.* http://www.w3.org/TR/rdf-sparql-query/.

22. *RSS*. http://www.rssboard.org/rss-specification.

23. *SIOC*. *(Semantically-Interlinked Online Communities)* (2012). http://sioc-project.org/.

24. *schema.org*. (2012). http://www.schema.org.

25. *SPARQL 1.1*. (2012). http://www.w3.org/TR/sparql11-query/.

26. *SPARQL End Point Specification* (2012). http://semanticweb.org/wiki/SPARQL_endpoint.

27. Telefonica categories at RENDER Wiki

28. ter Horst, H. J. Combining RDF and Part of OWL with Rules: Semantics, Decidability, Complexity . In Proceedings of The Semantic Web ISWC 2005, LNCS volume 3729 pp. 668–684. Springer Berlin / Heidelberg, 2005.

