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

RENDER

Deliverable D1.1.1

Initial collection of data

Editor:	Atanas Kiryakov, Ontotext
Deliverable nature:	Prototype (P)
Dissemination level: (Confidentiality)	Public (PU)
Contractual delivery date:	December 2010
Actual delivery date:	January 2011
Suggested readers:	Example: Technical and research stuff working on the data collection and management; developers working on use cases.
Version:	0.3
Total number of pages:	23
Keywords:	

Abstract

This deliverable reports on the initial work on data collection and management in RENDER. We have identified the different types of data which need to be managed within the project and prepared a questionnaire to facilitate the gathering of concrete requirements for data collection and management.

Brief descriptions of pre-existing resources and technologies is provided, namely JSI's facilities for collection of news and blog content and Ontotext's linked data exploration service FactForge. Initial work has been also performed by KIT on collection of corpus of Wikipedia articles from different language versions.

To handle the vast diversity and the dynamicity of the data we came up with a data organisation approach based on the so-called Reference Knowledge Stack, which includes several components of different size and nature: the PROTON, OpenCyc, UMBEL and few of the central LOD datasets. A new version of UMBEL was developed, which interlinks 21 000 concepts from OpenCyc and with the entities in DBPedia. The proposed data organisation approach will be modified in accordance with the diversity modelling principles which will be developed in T3.1.

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]	D1.1.1 Initial collection of data
[Editor: Name, company]	Atanas Kiryakov, Ontotext AD
[Work-package leader: Name, company]	Atanas Kiryakov, Ontotext AD
[Estimation of PM spent on the Deliverable]	3.1

Executive summary

The work within WP1 started with building an initial version of the data management concept for the project, which was required to structure the activities on initial data collection. We have identified the different types of data which need to be managed within the project (news, blogs and collaborative wiki publications, tweets, enterprise communication, ontologies, linked data and other factual knowledge). A questionnaire was prepared to facilitate the solicitation of more concrete requirements for data collection and management.

The deliverable provides brief descriptions of pre-existing resources and technologies relevant to the subject, namely JSI's facilities for collection of news and blog content and Ontotext's linked data exploration service FactForge. Only minor efforts have been invested to extend and update those in order to make best fit in the context of RENDER. Initial work has been also performed by KIT on collection of corpus of Wikipedia articles from different language versions of the encyclopaedia.

Major efforts went in developing data organisation and management concept, which can handle the vast diversity and the dynamicity of the data. A key requirement was to allow for interlinking of all sorts of data relevant to the project and to facilitate different types of access to them from users with different background, technical skills, and level of familiarity with the data. We came up with a data organisation approach based on the so-called Reference Knowledge Stack, which includes several components of different size and nature: the PROTON upper-level ontology, OpenCyc common sense ontology, UMBEL and few of the central LOD datasets, including DBPedia, Freebase, Geonames, Wordnet, MusicBrainz. The proposed data organisation approach is mean to facilitate data management in the initial phase of the project; it will be modified in accordance with the diversity modelling principles which will be developed in T3.1.

A concrete technical contribution is the latest version (0.8) of the UMBEL reference ontology, which includes 21 000 concepts, derived from OpenCyc and interlinked with the entities in DBPedia. Still, most of the ontology alignment work required for the implementation of the Reference Knowledge Stack are still in progress and will be reported in deliverable D1.2.1; this is the case with mappings from PROTON to UMBEL and to the specific schemata end ontologies of DBPedia, Geonames and Freebase. The resources and the mappings form the Reference Knowledge Stack will become publicly available in the forthcoming major update to FactForge. The latter, together with relevant sample content (e.g. blog posts, news, and Wikipedia articles) will be made publicly available for searching, querying and exploration through the Forest web UI framework (already proven at FactForge.net and LinkedLifeData.com).

List of authors

Organisation	Author
Ontotext	Atanas Kiryakov
Ontotext	Maurice Grinberg
Ontotext	Mariana Damova
JSI	Delia Rusu

Table of Contents

Executive summary 3
List of authors 4
Table of Contents 5
Abbreviations 6
Definitions 7
1 Introduction 8
1.1 Linked Data 8
1.2 DBPedia 9
1.3 Reason-able Views: Manageable Linked Data 10
2 Data Collection Requirements Solicitation 11
3 Initial Data Collection 12
3.1 Collection of Blogs and News 12
3.2 FactForge: Linked Data Collection 13
3.3 Parallel Corpus of Wikipedia Articles 13
4 Data Organisation Approach 14
4.1 Reference Data and Ontologies 14
4.2 Reference Data-Based Management of Diverse Data 14
4.3 Reference Knowledge Stack 17
4.4 UMBEL – Linking DBPedia instances to OpenCyc Classes 19
5 Conclusion and Future Work 21
References 22

Abbreviations

DBMS – database management systems, such as the relational database engines;

DBPedia – an RDF dataset derived from Wikipedia, aiming to provide as complete as possible coverage of the factual knowledge that can be extracted with high precision from there. DBPedia is one of the most central LOD datasets; more information is available in section 1.2.

LOD – the Linking Open Data project is a W3C SWEO Community project and is an initiative for publishing "linked data"; more details are provided in section 1.1 and at [31];

RDF – Resource description framework, a basic specification determining the data model of the Semantic Web, [20];

SPARQL – a query language for RDF specified in [25];

UMBEL: an upper-level ontology defining about 20 thousand concepts (classes and predicates) derived from OpenCyc and clustered into 30 Super Types. UMBEL is equipped also with a mapping of the DBPedia entities with respect to the classes from OpenCyc. Description is provided in section 4.4.

Definitions

This material assumes prior knowledge of the basic semantic web standards, namely RDF, [21], RDFS, [8], and OWL, [13].

Linked data: Linked data represents a set of principles for publishing of structured data they can be explored and navigated in a manner analogous to the HTML WWW. The linked data concept is an enabling factor for the realization of the Semantic Web as a global web of structured data around the Linking Open Data initiative. The notion of "linked data" is defined by Tim Berners-Lee in (http://www.w3.org/DesignIssues/LinkedData.html) and prescribes that data should be published on the WWW as RDF graphs. It is viewed as a method for sharing and connecting pieces of data, information, and knowledge on the Semantic Web using URIs and RDF;

Master data: Master data, which may include reference data, is information that is key to the operation of business and is the primary focus of the Information Technology (IT) discipline of Master Data Management (MDM). This key business information may include data about customers, products, employees, materials, suppliers, etc. which often turns out to be non-transactional in nature, [29];

OpenCyc: OpenCyc is the open source version of the Cyc technology, the world's largest and most complete general knowledge base and commonsense reasoning engine;

PROTON: an upper-level schema ontology which defines about 300 classes and 100 properties relevant for entity classification, description and relation across multiple domains;

Reason-able view: Reason-able views, [17], represent an approach for reasoning and management of linked data. It can be obtained by grouping selected datasets and ontologies in a compound dataset, clean-up and post-processing and enriching the datasets if necessary for each new version of the dataset. The compound dataset is loaded in a single semantic repository.

Reference master data: reference data shared over a number of systems, [30].

Reference data: data describing a physical or virtual object and its properties. Reference data is used in data management to define characteristics of an identifier that are used within other data centric processes, [30].

Reference Knowledge Stack: a data organisation approach combining several types of ontologies and datasets, that can be used together as master reference data. See section 4.3 for further details.

RDF Molecule: the description of an URI node in an RDF graph, including only the minimal information that describes the URI, as defined in section 3.2 of [19]. Technically, the molecule of node S is the part of the graph that you can reach following all paths in the graph, starting from S, until you reach non-blank nodes. This notion is close to the one defined in [23], but quite different from the triple-centric definition provided in [12].

1 Introduction

The work within WP1 started with building an initial version of the data management concept for the project, which was required to structure the activities on initial data collection. We have identified at a high level the different types of data that need to be managed within the project, namely: news articles, blog posts, tweets statuses, ontologies, linked data and other factual knowledge. More specific information about the data that should be collected and managed within the project, its volumes, processing and access requirements will be available at the end of the next project period (M4-M6), based on the requirements collection form presented in section 2. Section 3 provides description of news and blog data which is already collected by JSI, as well as linked data collection by Ontotext and some initial work on Wikipedia article collection from KIT.

The most challenging part of the work in the work package was to develop a concept for data organisation and integration, which allows for easy management and access to the data, while at the same time respecting the diversity and the dynamicity of the different types and pieces of data relevant to the project and its use cases. We define a data organisation approach based on the so-called reference knowledge stack, including: PROTON, UMBEL, OpenCyc and FactForge. This concept is presented in section 4, along with some of the the work performed towards its realisation, namely: developing of updated version of UMBEL (mapping DBPedia to OpenCyc).

The reminder of this section provides in introduction to the so-called “linked data”, DBPedia (the most popular linked data datasest) and an approach for linked data management called “reason-able” views, which can be seen as a fundament to the data organisation within the project.

1.1 Linked Data

The notion of “linked data” is defined by Tim Berners-Lee, [4][6], as RDF graphs, published on the WWW so that one can explore them across servers by following the links in the graph in a manner similar to the way the HTML web is navigated. It is viewed as a method for exposing, sharing, and connecting pieces of data, information, and knowledge on the Semantic Web using URIs and RDF. “Linked data” are constituted by publishing and interlinking open data sources, following principles:

1. Using URIs as names for things;
2. Using HTTP URIs, so that people can look up those names;
3. Providing useful information when someone looks up a URI;
4. Including links to other URI, so people can discover more things.

In fact, most of the RDF datasets fulfil principles 1, 2, and 4 by design. The piece of novelty in the design principles above concerns the requirement for enabling Semantic Web browsers to load HTTP descriptions of RDF resources based on their URIs. To this end, data publishers should make sure that:

- the “physical” addresses of the published pieces of data are the same as the “logical” addresses, used as RDF identifiers (URIs);
- upon receiving an HTTP request, the server should return a set of triples that describe the resource.

Linking Open Data (LOD, [31]) is a W3C SWEO community project aiming to extend the Web by publishing open datasets as RDF and by creating RDF links between data items from different data sources. Linked Open Data provides sets of referenceable, semantically interlinked resources with defined meaning. The central dataset of the LOD is DBPedia (see section 1.2). Because of the many mappings between other LOD datasets and DBPedia, the latter serves as a sort of a hub in the LOD graph assuring a certain level of connectivity. LOD is rapidly growing – as of September 2010 it contains more than 200 datasets, with total volume above 25 billion statements, interlinked with 395 million statements as illustrated on Figure 1 (the figure presents an older pictured of the dataset map as the new one is too detailed to be readable in this format).

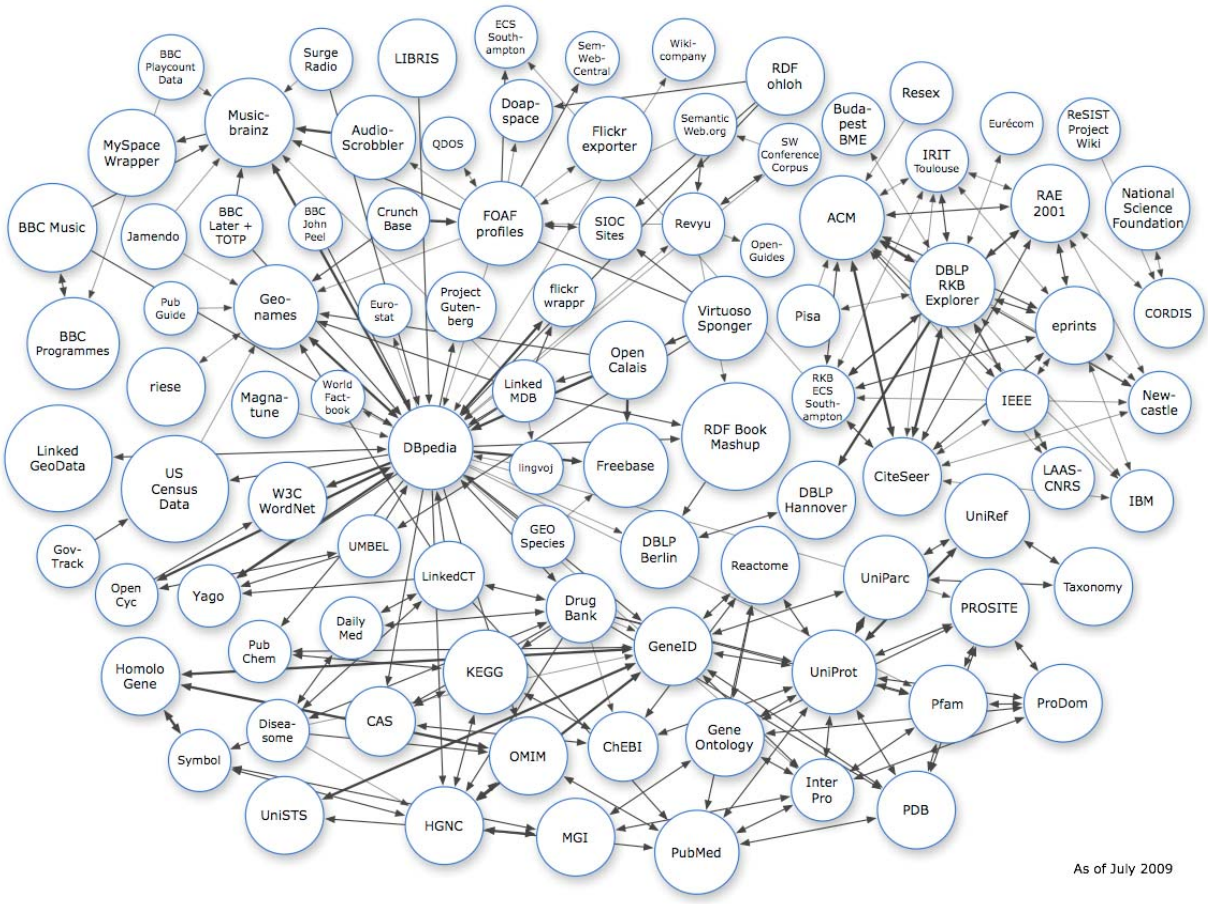

Figure 1. *Map of the Datasets in Linking Open Data (LOD) Project, [31]*

1.2 DBPedia

DBPedia (http://dbpedia.org) is an RDF dataset derived from Wikipedia, aiming to provide as full as possible coverage of the factual knowledge that can be extracted with high precision from there. Because of the vast amount of mappings between other LOD datasets and DBPedia, it serves as a sort of a hub in the LOD graph assuring a certain level of connectivity. Description of the major objectives and design principles behind DBPedia can be found in [7].

Currently, DBPedia version 3.6 consists of about 672 million explicit statements, of which 286 million are extracted from the English version of Wikipedia. It contains descriptions of 3.5 million things, of which 1.7 million are classified in the DBPedia's own ontology. DBpedia includes also 6.5 million links to external datasets, which means almost 2 links per described thing.

It should be noted that most of the factual knowledge, in terms of attribute-value pairs and relations between objects in DBPedia, is derived from the so-called Info-boxes of Wikipedia. As a result, DBPedia is characteristic for the fact that about 100 thousand different properties (attribute or relation types) are used with little degree of reuse and modelling consent. Often one and the same type of relationship (e.g. employment) is represented in tens of hundreds of different ways, which makes querying DBPedia with structured query languages such as SPARQL, [25], very challenging and unrewarding exercise.

1.3 Reason-able Views: Manageable Linked Data

Using linked data (see section 1.1) for data management is considered to have great potential. On the other hand there are several challenges to be handled to make this possible, namely:

- LOD is hard to comprehend – the fact that multiple datasets are interlinked and accessible in the same data format does not on itself help in dealing with hundreds of data schemata, ontologies, vocabularies and data modelling patterns;
- Diversity comes at a price – often there are tens of different ways of expressing one and the same information in even in a single dataset such as DBPedia (see section 1.2);
- LOD is unreliable – many of the servers behind LOD today are slow and have down times higher than the one acceptable for most of the data management setups;
- Dealing with data distributed on the web is slow – a federated SPARQL query that uses, say, 3 servers within several joins can be very slow;
- No kind of consistency is guaranteed – low commitment to the formal semantics and intended usage of the ontologies and schemata.

Reason-able views, [17], represent an approach for reasoning with and management of linked data defined at Ontotext and implemented in two systems, namely, FactForge (presented in section 3.2) and LinkedLifeData (http://www.linkedlifedata.com). *Reason-able view* is an assembly of independent datasets, which can be used as a single body of knowledge with respect to reasoning and query evaluation. The key principles can be summarized as follows:

- Group selected datasets and ontologies in a compound dataset;
- Clean up, post-process and enrich the datasets if necessary. Do this conservatively, in a clearly documented and automated manner, so that (i) the operation can easily be performed each time a new version of one of the datasets is published and (ii) the users can easily understand the intervention made to the original dataset;
- Load the compound dataset in a single semantic repository and perform inference with respect to tractable OWL dialects;
- Define a set of sample queries against the compound dataset. These determine the “level of service” or the “scope of consistency” contract offered by the reason-able view.

Each reason-able view is aiming at lowering the cost and the risks of using specific linked data datasets for specific purposes. The design objectives behind each reason-able view are as follows:

- Make reasoning and query evaluation feasible;
- Lower the cost of entry through interactive user interfaces and retrieval methods such as URI auto-completion and *RDF search* (a search modality where RDF molecules are being retrieved and ranked by relevance to a full-text style query, represented as set of keywords)*;*
- Guarantee a basic level of consistency – the sample queries guarantee the consistency of the data in the same way in which regression tests do for the quality of software;
- Guarantee availability – in the same way in which web search engines are usually more reliable than most of the web sites; they also do caching;
- Easier exploration and querying of unseen data – sample queries provide re-usable extraction patterns, which reduce the time for acquaintance with the datasets and their interconnections.

2 Data Collection Requirements Solicitation

The requirements and the needs for data collection will be gathered from the consortium, paying specific attention to the needs of the use cases. The task for soliciting these requirements was scheduled to start in M4 because in the very beginning of the project the partners are still not prepared to provide such information – those requirements should emerge from the initial analytical and planning activities in each of the RTD and use cases work packages.

The process of soliciting data collection and management requirements will be driven by a questionnaire, addressing the following aspects:

- What data is need?
    - from what sources and what portion of the information;
    - how often it should be collected and what would be the update policy;
    - expected size or volume of the data;
- What processing and integration are required?
    - What other datasets are related to this one?
    - Is there a need for some sort of (pre/post)processing of the data?
    - How should the data be linked to the reference knowledge stack (seen section 4.3)?
    - Is there a need for automated integration of the data with other pieces of data?
- What this information will be used for?
    - What structured queries/patterns will be evaluated against it?
    - Are the specific ordering and filtering criteria applicable?
    - Is there specific need of clustering or summarization?
- What are the publishing requirements?
    - Shall the data be made available to the general public or it is only to be used for auxiliary purposes?
    - Shall the data be published as “linked data”?
    - What other publishing formats and access/retrieval methods are desired?

Each partner will be asked to fill in several copies of this questionnaire – one for each kind of data that should be collected.

3 Initial Data Collection

The data collection work still needs to be further specified and implemented. Still, some efforts were made to collect data which is needed for the research and the technology work in the first phase of the project. Those efforts are described in the following subsections.

3.1 Collection of Blogs and News

JSI has prepared two data collections. One data collection is based on Spinn3r, while the other is based on the results provided by a news crawler developed by JSI. JSI's Spinn3r-based application as well as the news crawler will be available as a web service, available to project partners for the purposes and the duration of project, providing the following functionality:

- Allow retrieval and search over the database
- Allow publish-subscribe functionality for pushing new articles

Spinn3r, [26], is a web service for indexing the blogosphere, provided by a start-up company from the United States. Spinn3r is focused on crawling and streaming mainstream news and social media. It monitors around 40 million blogs and 10,000 news outlets, resulting in up to 30 million daily items. Any new items on the monitored sites are automatically crawled, (partly) cleaned, time-stamped and provided to their users as a continuous RSS feed.

Spinn3r API is using ProtoBuffer protocol, [15], defined by Google for efficient transfer of structured data, that can be seen as a simplified analogue of XML. It can be accessed using a command-line Java client provided by Spinn3r. The client connects to the stream and downloads new items in a local directory using the following loop:

```
1. LAST_RECIEVED_ITEM_TIME = Now – 10 minutes
2. NUMBER_OF_ITEMS = 1000
3. while (true) {
4.   NEW_ITEMS = receive(LAST_RECEIVED_ITEM_TIME, NUMBER_OF_ITEMS)
5.   save_as_xml(NEW_ITEMS, LOCAL_DIRECTORY + guid() + ".xml.gz")
6.   LAST_RECEIVED_ITEM_TIME = NEW_ITEMS.LAST.TIME
7. }
```

Each item crawled by Spinn3r is described by, among others, the following fields:

- **Publisher type** – type of the source; possible types are mainstream news, blog, micro blog (e.g. Twitter) and social media (e.g. Facebook status updates);
- **Publisher URL** – address of the website which published the item;
- **Language** – Language of the text in the item;
- **Item URL** – URL from where the item was retrieved;
- **Item Title** – Extracted from HTML Title field, if available;
- **Item Date** – date and time when the item was crawled.

The client was tested at Jozef Stefan Institute for more than a year and is proven to provide real-time news stream access. Technology was also developed for parsing and indexing the items in real-time. The whole system requires a high-end workstation to handle the load.

Spinn3r offers special research licenses for their service. It is free when used non-commercial research use and costs a small monthly fee (at the time of writing it was 500 USD) when used inside projects with grant money.

In addition, JSI developed its own news crawler, which monitors:

a) RSS feeds of sites that were heuristically determined to be news media outlets and
b) Google News

For Google News, the whole hierarchical structure is stored, in particular the grouping of articles into stories. For both data sources, we search the crawled pages for any additional RSS feeds and include those in future crawls as well, thus ever increasing the amount of reachable content. Both sources combined amount to about 55 000 news articles per day. The articles that we crawl are stripped of redundant mark-up and content (menus, advertisements, etc.) for all news outlets with over 95% precision and recall. A rule-based algorithm is currently being used; however, we plan to replace it in the near future with an implementation based on [24], which exhibits similar quality but higher efficiency.

3.2 FactForge: Linked Data Collection

FactForge (http://www.factforge.net/, [5], previously known as LDSR) represents a reason-able view, to the web of linked data (see sections 1.1 and 1.3). It provides efficient mechanisms to query data from multiple datasets and sources, considering their semantics. FactForge includes several of the most central datasets of LOD: DBPedia, Freebase, Geonames, UMBEL, Wordnet, CIA World Factbook, Lingvoj, MusicBrainz (RDF from Zitgist). Along with the dataset specific schemata and ontologies the following ones have been loaded in FactForge: Dublin Core, SKOS, RSS, FOAF; those were referred or imported, so, they were necessary to allow for proper interpretation of the semantics of the data.

OWLIM semantic repository is used to load the data and "materialize" the facts that could be inferred from it. FactForge is probably the largest and most heterogeneous body of general factual knowledge that was ever used for logical inference. The inference was performed with respect to ruleset derived from the so-called OWL Horst dialect, [27]. The only dataset which required modification before loading in FactForge is DBPedia:

- We had to remove the YAGO module as some incorrect classifications of entities and others faults in it were causing inference of too many faulty statements in FactForge;
- A clean up of the category hierarchy was required, as discussed in section 3.4 of [18].

FactForge is in development for more than couple of years so far. Its latest developments were targeted towards its anticipated usage in RENDER project as central data organisation block (see section 4.3). Apart from the ongoing updates of the content with respect to the latest versions of the included datasets, FactForge has been extended with OpenCyc and the New York Time's dataset. Adding these datasets, apart from other effects, was aimed to further mitigate the domination of the DBPedia and Wikipedia with respect to the instance identification vocabulary, i.e. the coverage of topics and entities and their naming. This updated version is still work in progress – it will be published before M6 of the project; these update will be presented in D1.2.1.

3.3 Parallel Corpus of Wikipedia Articles

KIT started to collect Wikipedia data and to run first analyses over it, by creating an environment to easily update the data and create several statistics. One of the objectives is to collect Wikipedia articles from different language editions, this way creating a sort of "parallel corpus" that contains diverse opinions on one and the same subject (e.g. a specific historical event).

In the course of the current experiments data is collected in Croatian, Bosnian, Serbian, Slovenian, and Italian. Altogether the corpus contains more than half a billion tokens.

4 Data Organisation Approach

The mainstream paradigms for dealing with structured data (e.g. relational or XML DBMS) assume good level of command of the schemata and the patterns, used for the modelling of the data. One cannot make a SQL query without knowing the tables and columns where the data is located. One of the major challenges in dealing with linked data is that knowing the schemata of the data is practically unfeasible due to several factors:

- They are developed without centralised control and strict modelling discipline;
- They are very diverse in the sense that the heterogeneity of the factual knowledge encoded in, for instance, DBPedia and Freebase is much larger than that in most of the existing database systems;

We believe that stable reference vocabulary can considerably facilitate the access to the dynamic and diverse data. The data organisation approach proposed in RENDER is based on the understanding that comprehensive reference data constellation is needed for efficient management of linked data in combination with any other data relevant to the use cases.

4.1 Reference Data and Ontologies

In data management, “reference data” and “master data” are terms used to describe a novel data management approach based on the understanding that difference has to be made between highly dynamic transactional data and more static data, which complements the schemata in modelling important aspects of the domain. In the design of most of the information systems, there is such kind of “important” data is often referred from the descriptions of concrete transactions. Thus, changes in this kind data is almost as critical as changes in the schemata of a database as there are large volumes of data which modelling depends on it.

According to [30], “Reference data is data describing a physical or virtual object and its properties. Reference data is used in data management to define characteristics of an identifier that are used within other data centric processes... Reference data is used in data management to define characteristics of an identifier that are used within other data centric processes.” According to [29], “Master data, which may include reference data, is information that is key to the operation of business and is the primary focus of the Information Technology (IT) discipline of Master Data Management (MDM). This key business information may include data about customers, products, employees, materials, suppliers, etc. which often turns out to be non-transactional in nature”. Further [30] defines “reference master data” as reference data shared over a number of systems and claims that it should be used instead of “master data” because “Master data is also the term used for original data, like an original recording (see also: Master Tape)”

In a broader perspective, “reference data” matches very closely the notion of ontology developed within the knowledge representation and Artificial Intelligence (AI) community as a paradigm for development of interoperable and reusable knowledge bases. The most popular definition is given in [14] as follows: “An ontology is an explicit specification of a conceptualization”, where “a conceptualization is an abstract, simplified view of the world that we wish to represent for some purpose.” Another widely used extended definition is provided in [9]: “An ontology is a formal, explicit specification of a shared conceptualization.” Through the years however the term ontology got a wide variety of meanings which made it highly ambiguous. For instance, in the definition of the OWL language “ontology” is considered to be any sort of information encoded in OWL, which may be even some “short-living” piece of transactional data. We will use “reference ontology” and “reference knowledge” to denote ontologies and knowledge bases which represent reference data.

4.2 Reference Data-Based Management of Diverse Data

We will try to explain the importance of reference data for data management by analogy to forestry. Suppose data is wood and reference data represent beaten paths through the forest. Forests are dynamic

and can be considered undetermined, at least to the extent that foresters do not know most of the particular trees and even for areas they know well, they cannot be certain about the changes that can take place within several months. Forest paths are known reference points and communication facilities, which facilitate the navigation within the forest, its exploration and overall, the access to the wood resources. In the same way reference data are well determined, relatively static and predictable data structures that can facilitate access to a diverse and dynamic set of the data as the web of linked data.

To access real volumes of wood foresters should, at some point, get off the beaten track and use methods and techniques to explore wild forest areas. Still, it is the case that beaten paths allow for exploitation of large forests, by means of lowering the efforts of their exploration. In a similar way linked data management is unthinkable (and in a sense pointless) without techniques which allow for dealing with unseen data – those are all sorts of automated statistical or symbolic methods which allow for analysis, interpretation, selection and retrieval of data. Still, using reference ontologies and more general reference knowledge structures has the potential to considerably lower the cost of using linked data as well as any collection of dynamic and diverse data collection.

A concrete example of how reference ontologies can facilitate access to linked data is query formulation. Imagine a situation where query must involve information from multiple datasets, each of which coming with its own ontology. As presented on Figure 2, in such case the specification of the query needs to use the vocabulary of each of these datasets and their ontologies; which means that the person who specifies the query should be acquainted with all these vocabularies.

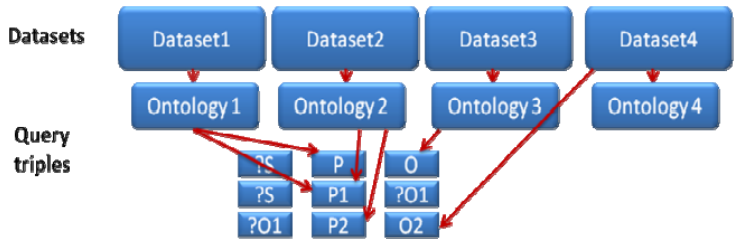

Figure 2. *Query constraints when multiple datasets are addressed directly*

Now imagine that the ontologies of each of the datasets are mapped to single reference ontology. This would allow query formulation using a single uniform vocabulary – the one of the reference ontology, as presented on Figure 3.

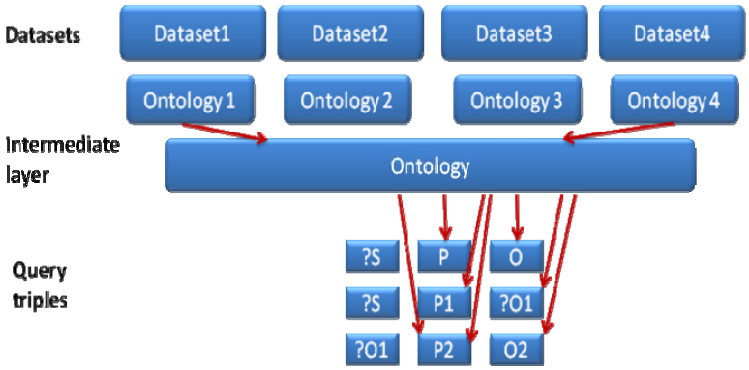

Figure 3. *Query constraints when multiple datasets are addressed via intermediate ontology*

The major advantage of using an intermediate ontology is that in order to formulate a query one needs to know a single ontology. To enable such scenario we mapped the PROTON upper-level ontology to the ontologies of several of the central LOD datasets. The schemata and the ontologies of DBPedia, Freebase and Geonames have been mapped to PROTON classes and properties as reported in [11].

To illustrate the benefits of such mapping we provide bellow two different representations of a query to FactForge (see section 3.2), which aims to retrieve cities where original paintings of Modigliani can be seen – this is the popular "Modigliani test" presented in [20]. The first version of the query below is that original one, which is addressing directly Freebase, DBPedia and UMBEL:

```
PREFIX fb: <http://rdf.freebase.com/ns/>
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX dbp-prop: <http://dbpedia.org/property/>
PREFIX dbp-ont: <http://dbpedia.org/ontology/>
PREFIX umbel-sc: <http://umbel.org/umbel/sc/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX ot: <http://www.ontotext.com/>

SELECT DISTINCT ?painting_l ?owner_l ?city_fb_con ?city_db_loc ?city_db_cit
WHERE {
      ?p fb:visual_art.artwork.artist dbpedia:Amedeo_Modigliani ;
         fb:visual_art.artwork.owners [ fb:visual_art.artwork_owner_relationship.owner ?ow ] ;
         ot:preferredLabel ?painting_l.
      ?ow ot:preferredLabel ?owner_l .
      OPTIONAL { ?ow fb:location.location.containedby [ ot:preferredLabel ?city_fb_con ] }
      OPTIONAL { ?ow dbp-prop:location [ rdf:type umbel-sc:City ; ot:preferredLabel ?city_db_loc ] }
      OPTIONAL { ?ow dbp-ont:city [ ot:preferredLabel ?city_db_cit ] }
}
```

Below follows a simplified version of the same query that can bring the desired results when FactForge gets extended with PROTON and its mappings to the LOD ontologies:

```
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX ot: <http://www.ontotext.com/>
PREFIX ptop: <http://proton.semanticweb.org/protont#>
PREFIX ploc: <http://proton.semanticweb.org/protonl#>
PREFIX p-ext: <http://proton.semanticweb.org/protonue#>

SELECT DISTINCT ?painting ?owner ?city
WHERE {
      ?p p-ext:author dbpedia:Amedeo_Modigliani ;
         p-ext:ownership [ ptop:isOwnedBy ?ow ] ;
         ot:preferredLabel ?painting .
      ?ow ot:preferredLabel ?owner ;
          ptop:locatedIn [ rdf:type ploc:City ; ot:preferredLabel ?city ].
}
```

In this case querying through PROTON brings several advantages:

- One does not need to search through the vast majority of predicates defined in both DBPedia and Freebase – even the small set of about 100 properties defined in PROTON appear to be sufficient for the formulation of this query;
- There is no need of multiple optional patterns because the most popular variations of "located in" relationships form Freebase and DBPedia are all mapped to the ptop:**locatedIn** property in PROTON;
- The query is much shorter, easier to define and understand.

It has to be noted that an upper-level ontology such as PROTON cannot cover all sorts of distinctions made in DBPedia and Freebase. Querying those datasets through PROTON will always have limited "resolution" as PROTON defines classes and properties, which are much more general than most of those in the specific datasets. On the other hand, defining queries using the fine-grained original vocabularies is simply unfeasible in many scenarios. This way, PROTON allows for an easy entry point for exploration of LOD as one can get acquainted with its few hundred of concepts with much lower efforts as compared to those required for the larger (and in the case of DBPedia, less uniform) vocabularies of the specific datasets.

We will evaluate if this mapping approach is feasible within Render – it could appear that although reference vocabularies facilitate the access to the data they somehow limit diversity. We are prepared to

extend the underlying data model according to the results of Task 3.1 in order to resolve such shortcomings of the proposed data organisation. One should note that the existence of the reference vocabulary does not mean that one cannot use the vocabulary of the original datasets. If we extend the analogy to forestry from the beginning of this section, the existence of paths in the forest, does not prevent anyone to explore the forest ignoring them.

4.3 Reference Knowledge Stack

Mapping the schemata of several datasets to a small upper-level ontology as presented in the previous sub-section already facilitates multiple retrieval scenarios. On the other hand the data management needs within RENDER require a more comprehensive approach that considers both schema- and instance-level reference structures. It is also required that one can use reference structures with higher granularity, in other words more extensive reference ontologies and datasets should be available. At the same time, this should be implemented in a way that preserves the low cost of entry provided by a relatively small upper-level ontology.

To meet the above requirements Ontotext, in cooperation with Structured Dynamics LLC, developed the concept for the so-called "Reference Knowledge Stack" (ReKS), which includes:

- PROTON – an upper-level ontology, 300 entity classes and 100 properties;
- UMBEL – 20 000 concepts extracted from OpenCyc and mapped to DBPedia instances (see section 4.4);
- OpenCyc – the largest and most comprehensive hand-crafted knowledge base, including 1.6 million statements;
- FactForge, combining the above with a "refined" version of DBPedia and the original versions Freebase, Geonames and few other LOD datasets in a compound dataset containing couple of billon explicit statements (see section 3.2).

Figure 4 represents the mappings that are available or under development between the different elements of the reference knowledge stack.

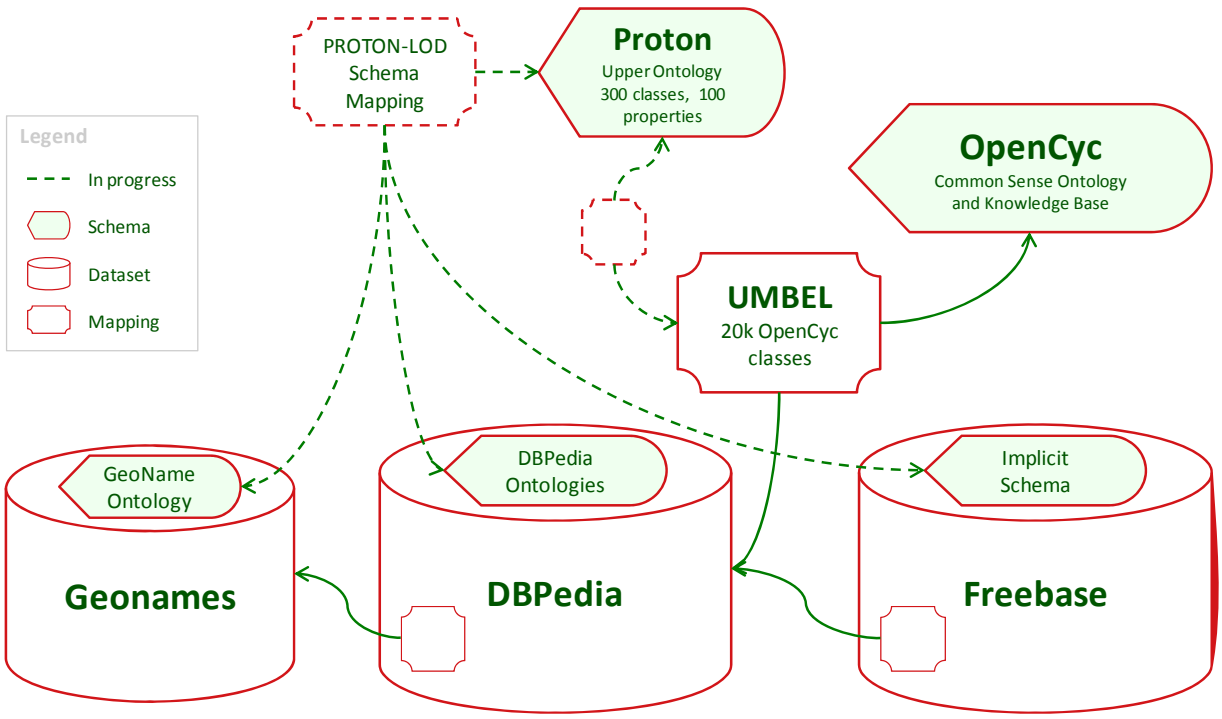

Figure 4. *Reference Knowledge Stack*

The following Table 1 describes the major features of the datasets and their intended usage.

Table 1 Reference Knowledge Stack Elements

Dataset	**Size** (approx.)	**Schema-level Vocabulary**	**Instance-level vocabulary**	**Reliable formal semantics**
PROTON	400+ concepts	+		+
UMBEL	20 000 classes	+		+
OpenCyc	2 million assertions	+	+	+
DBPedia	700 million assertions		+	
Freebase	500 million assertions			+
Geonames	100 million statements	+		+

We will illustrate a sample usage of such a reference knowledge stack with another sample query of FactForge, which aims to retrieve "the most popular entertainers born in Germany".

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbp-ont: <http://dbpedia.org/ontology/>
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX opencyc: <http://sw.opencyc.org/2008/06/10/concept/en/>
PREFIX geo-ont: <http://www.geonames.org/ontology#>
PREFIX om: http://www.ontotext.com/owlim/

SELECT * WHERE {
     ?Person dbp-ont:birthPlace ?BirthPlace ;
             rdf:type opencyc:Entertainer ;
             om:hasRDFRank ?RR .
     ?BirthPlace geo-ont:parentFeature dbpedia:Germany .
} ORDER BY DESC(?RR) LIMIT 100
```

The definition of this query involves: schema- and instance-level vocabulary from DBPedia; schema-level vocabulary of OpenCyc and schema-level vocabulary from Geonames[1]. A system predicate of the BigOWLIM semantic repository, where FactForge's compound dataset is loaded, is used to retrieve the so-called RDFRank of the corresponding node, which represents an equivalent of Google's PageRank computed on top of the RDF graph. This rank is used as a measure for "popularity". It is interesting to note that this query returns unexpected results – on top of the results set is Friedrich Nietzsche. He qualifies as an entertainer because the little known fact that he was not only philosopher, but also virtuoso piano player, was available in the MusicBrainz dataset. Although MusicBrainz vocabulary is not used in the query, this fact was considered because of the mappings between MusicBrainz classes and the OpenCyc classes which are

[1] A version of the query close to the one originally provided as answer to the Modigliani test is presented. Still, one should note that this query can also be formulated by using just DBPedia instance vocabulary and PROTON schema vocabulary. In this case we preferred to use the schemata of Geonames and DBPedia for two reasons: (i) to indicate where the corresponding facts are coming from and (ii) to demonstrate that, with the proposed approach, often there are various way to formulate one and the same query.

established through UMBEL and the mappings between instance level identifiers between MusicBrainz and DBPedia.

Technically, the Reference Knowledge Stack will become available as next version of the FactForge reasonable view, which would allow the different components to be used together or in separation.

Most of the mappings within the reference knowledge stack are developed semi-automatically, using various machine learning techniques. More details on those mappings will be provided in deliverable D1.2.1. An illustration of the potential of the automated methods for ontology alignment is presented in [16], which uses the hand-crafter mapping between PROTON and the schemata of DBPedia, Geonames and FactForge, reported in [11], as golden standard.

4.4 UMBEL – Linking DBPedia instances to OpenCyc Classes

UMBEL is the Upper Mapping and Binding Exchange Layer, designed to help content interoperability on the Web, [13]. UMBEL has two purposes:

- to provide a general vocabulary of classes and predicates (the UMBEL "vocabulary") for describing and interlinking domain ontologies;
- to provide a coherent framework of broad subjects and topics (the UMBEL "reference concepts"), suitable for mapping relevant Web-accessible content.

UMBEL has about 21,000 reference concepts drawn from the OpenCyc knowledge base, which are organized into more than 30 SuperTypes. One of its most valuable characteristics is that UMBEL is packed with mapping of all the DBPedia entities to UMBEL reference concepts. This allows one to combine the broad coverage of DBPedia with respect to popular entities with the sound class hierarchy of Cyc.

The *UMBEL vocabulary* defines some predicates – connecting verbs or properties – for linking disparate information sources together. The UMBEL Vocabulary is designed to recognize that different sources of information have different contexts and different structures. A meaningful vocabulary is necessary that can express potential relationships between two information sources with respect to their differences in structure and scope. By nature, these connections are not always exact, thus means for expressing the "approximateness" of relationships are essential.

The second step and purpose of UMBEL is to provide a fixed set of 'Reference Concepts' by which these approximate alignments can be oriented. By design, this set of fixed reference points is neither exact nor comprehensive. These reference concepts are not meant to model the world in all of its complexity and nuance. UMBEL's goal is to provide a set of fixed references by which constituent content can be oriented and navigated. The goal is to describe the constituent information in terms of what it is about and try to gather similar relevant content together.

The coherent set of UMBEL reference concepts began with the Cyc knowledge base. However, since its scope and sophistication far exceeded what was tractable for a lightweight reference structure, Cyc was pruned and cleaned to a significant degree. All of the UMBEL Reference Concepts and their relationships are derived from the OpenCyc ontology. This means that UMBEL is a clean, 100% subset of OpenCyc. The result is an UMBEL reference structure of about 21,000 concepts, broadly applicable as orienting nodes to any knowledge domain, all coherently structured and linked to one another. This winnowing produced the lightweight UMBEL Reference Concept ontology.

The UMBEL Reference Concept ontology is, in essence, a content graph of subject nodes related to one another via broader-than and narrower-than relations. In turn, these internal UMBEL Reference Concepts may be related to external classes and individuals (instances and named entities) via a set of relational, equivalent, or alignment predicates. This UMBEL Vocabulary is itself a solid basis for constructing domain ontologies that can also act as reference ontologies within their own domains.

In the beginning of the RENDER project, in a partnership between Structured Dynamics and Ontotext, the UMBEL framework has been applied and refined. Significant use cases have been tested, notably with FactForge and Proton. The UMBEL reference ontology has been better organized and made easier to

browse via the addition of 33 new SuperType classes clustered into nine dimensions. Many early vocabulary decisions have been revised, and substantial improvements across the board have been made in terms of structure and documentation. Version 0.80 is also now fully OWL 2 compliant.

5 Conclusion and Future Work

This deliverable reports on the initial work on data collection and management in RENDER. First, it outlines a questionnaire that will be used to solicit data collection requirements from the consortium – it should be noted that although the types of data relevant to the project are already clear, the specific pieces and sources of data are still to be determined. Further, the deliverable provides brief descriptions of pre-existing resources and technologies relevant to the subject, namely JSI's facilities for collection of news and blog content and Ontotext's linked data management infrastructure. Only minor efforts have been invested to extend and update those in order to make best fit in the context of RENDER.

The central novel contribution is the concept for data organisation within RENDER based on the so-called Reference Knowledge Stack, which includes several components of different size and nature: the PROTON upper-level ontology, OpenCyc, UMBEL and few of the central LOD datasets, including DBPedia, Freebase, Geonames, Wordnet, MusicBrainz and others. This reference structure is meant to allow for interlinking all sorts of data relevant to the project and to facilitate different types of access from users with different background, technical skills and level of familiarity with the data.

A concrete technical contribution is the latest version (0.8) of the UMBEL reference ontology, which includes 21 000 concepts, derived from OpenCyc and interlinked with the entities in DBPedia and Wikipedia. Still, most of the ontology alignment work required for the implementation of the Reference Knowledge Stack is still in progress and will be reported in deliverable D1.2.1; this is the case with mappings from PROTON to UMBEL on the one hand and to the specific schemata end ontologies of DBPedia, Geonames and Freebase on the other. All these reference resources will become publicly available in the forthcoming major update to FactForge (also to be reported in D1.2.1). All appropriate resources which are not already published as linked data will be added to the LOD cloud – this is particularly the case of PROTON together with its various mappings. The reference stack along with sample content and other information will be published through the Forest framework, to enable its efficient access and querying on the web (this work will be reported in D1.3.1).

The proposed data organisation approach is mean to facilitate data management in the initial phase of the project. To make sure that the proposed reference structures do not limit the diversity, the overall approach will be modified in accordance with the diversity modelling principles which will be developed in T3.1.

References

[1] Bechofer, S, van Harmelen, F., Hendler, J., Horrocks, I., McGuinness, D. L., Patel-Schneider, P. F., and Stein, L. A. *OWL Web Ontology Language Reference*. In: Dean, M., Schreiber, G. (Eds.), W3C Recommendation, February 10, 2004. http://www.w3.org/TR/owl-ref/.

[2] Bergman, M. K. *Bridging the Gaps: Adaptive Approaches to Data Interoperability*. Keynote presentation at DC-2010 Conference, Pittsburgh, PA, October 22, 2010. http://www.slideshare.net/mkbergman/dcmi-20101022. (2010)

[3] Bergman, M. *Announcing a Major, New UMBEL Release.* http://www.mkbergman.com/930/announcing-a-major-new-umbel-release/ November (2010)

[4] Berners-Lee, T.: *Design Issues: Linked Data*. http://www.w3.org/DesignIssues/LinkedData.html (2006)

[5] Bishop, B.; Kiryakov, A.; Ognyanoff, D.; Peikov, I.; Tashev, Z.; Velkov, R. *FactForge: A fast track to the web of data*. Submission for Semantic Web Journal, Special Issue: Real-World Applications of OWL. http://www.semantic-web-journal.net/content/new-submission-factforge-fast-track-web-data. To appear. (2011)

[6] Bizer, C., Heath, T., and Berners-Lee, T. *Linked Data – The Story so Far*. In: Heath, T., Hepp, M. and Bizer, C. (Eds.) Special Issue on Linked Data, International Journal on Semantic Web and Information Systems (IJSWIS), http://linkeddata.org/docs/ijswis-special-issue, (2009)

[7] Bizer, C.; Lehmann, J.; Kobilarov, G.; Auer, S.; Becker, C.; Cyganiak, R.; Hellmann, S. *DBpedia – A Crystallization Point for the Web of Data.* Journal of Web Semantics: Science, Services and Agents on the World Wide Web, Issue 7, Pages 154–165, 2009.

[8] Brickley, D., Guha, R.V, eds.: *Resource Description Framework (RDF) Schemas*. W3C Recommendation, 10 February 2004. http://www.w3.org/TR/rdf-schema/ (2004)

[9] Borst, P.; Akkermans, H.; Top, J. *Engineering Ontologies*. International Journal of Human-Computer Studies, (46)365-406, (1997)

[10] Cycorp. *OpenCyc*. http://www.cyc.com/cyc/opencyc

[11] Damova, M., Kiryakov, A., Simov, K., and Petrov, S. *Mapping the central LOD ontologies to PROTON upper-level ontology*. Ontology Mapping Workshop at ISWC 2010, http://om2010.ontologymatching.org. (2010)

[12] Ding, Li., Finin, T., Peng, Y., da Silva, P. , McGuinness, D.: *Tracking RDF Graph Provenance using RDF Molecules.* http://ebiquity.umbc.edu/_file_directory_/papers/178.pdf, (2005)

[13] Giasson, F.; Bergman, M.; eds.: *Upper Mapping and Binding Exchange Layer (UMBEL) Specification.*http://umbel.org/specifications/full-specification, version 0.8, Nov 2010. (2010)

[14] Gruber, T. R. *A translation approach to portable ontologies.* Knowledge Acquisition, 5(2):199-220, 1993. http://ksl-web.stanford.edu/KSL_Abstracts/KSL-92-71.html (1992)

[15] Google Code. *Protocol Buffers: Developers Guide.* http://code.google.com/apis/protocolbuffers/docs/overview.html. (2010)

[16] Jain, P., Yeh, P. Z., Verma, K., Vasquez, R. G., Damova, M., Hitzler, P., and Sheth, A. P. *Contextual Ontology Alignment of LOD with an Upper Ontology: A Case Study with Proton*. http://knoesis.wright.edu/library/download/prateek_tech_report_2010.pdf. (2010)

[17] Kiryakov, A., and Momtchev, V. *Two Reason-able Views to the Web of Linked Data.* Presentation at the Semantic Technology Conference 2009, San Jose. http://www.slideshare.net/ontotext/two-reasonable-views-to-the-web-of-linked-data. (2009)

[18] Kiryakov, A., Tashev, Z., Ognyanoff, D., Velkov, R., Momtchev, V., Balev, B., Peikov, I.: *Validation goals and metrics for the LarKC platform.* LarKC project deliverable D5.5.2. http://www.larkc.eu/deliverables/ (2009)

[19] Li, Y., Cunningham, H., Roberts, A., Kiryakov, A., Momtchev, V., Greenwood, M., Aswani, N., Damljanovic, D. *Selection Components (report accompanying two software deliverables).* LarKC project deliverable D2.2.1, 2.5.1, (2009)

[20] MacManus, R. *The Modigliani Test: The Semantic Web's Tipping Point*. http://www.readwriteweb.com/archives/the_modigliani_test_semantic_web_tipping_point.php, (2010)

[21] Manola F., and Miller, E. (eds.): *RDF Primer*. W3C Recommendation, 10 Feb 2004, http://www.w3.org/TR/rdf-primer/ , (2004)

[22] Momchev, V., Assel, M., Cheptsov, A., Bishop, B., Bradesko, L., Fuchs, C., Gallizo, G., Kotoulas, S., and Tagni, G. *D5.5.3 Report on platform validation and recommendation for next version*. LarKC EU-IST-2008-215535, (2010)

[23] Newman, A.; Li, Y.-F.; Hunter, J. *A Scale-Out RDF Molecule Store for Improved Co-Identification, Querying and Inferencing.* In The 4th International Workshop on Scalable Semantic Web knowledge Base Systems (SSWS) 2008, Karlsruhe, Germany, (2008)

[24] Pasternack, J., and Roth, D. *Extracting Article Text from the Web with Maximum Subsequence Segmentation*. In Proceedings of World Wide Web Conference, (2009)

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

[26] Tailrank Inc. *Spinn3r web service for indexing the blogosphere*. http://spinn3r.com/ (2009)

[27] ter Horst, H. J.: *Combining RDF and Part of OWL with Rules: Semantics, Decidability, Complexity*. In Proc. of ISWC 2005, Galway, Ireland, LNCS 3729, pp. 668-684, November 6-10, (2005)

[28] Terziev, I., Kiryakov, A., and Manov, D. *D.1.8.1 Base upper-level ontology (BULO) Guidance*. Deliverable of EU-IST Project IST – 2003 – 506826 SEKT (2005)

[29] Wikipedia. *Master data*. http://en.wikipedia.org/wiki/Master_data as of January 2011.

[30] Wikipedia. *Reference data*. http://en.wikipedia.org/wiki/Reference_data as of January 2011.

[31] World Wide Web Consortium (W3C): *Linking Open Data*. W3C SWEO community project home page, as of January 2010. http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData (2010)

