Scripting User Contributed Interlinking

Michael Hausenblas[1], Wolfgang Halb[1], and Yves Raimond[2]

[1] Institute of Information Systems and Information Management,
JOANNEUM RESEARCH, Steyrergasse 17, 8010 Graz, Austria
firstname.lastname@joanneum.at
[2] Centre for Digital Music,
Queen Mary, University of London, UK
yves.raimond@elec.qmul.ac.uk

Abstract. When building a linked-data dataset for humans and machines, a range of issues emerges. In this paper we discuss our findings regarding the implementation of riese (http://riese.joanneum.at), the RDFized and interlinked version of the Eurostat data. The contribution of our work is twofold: On the one hand we propose a new way of creating semantic links, labelled as User Contributed Interlinking, on the other hand we discuss integration issues regarding Ajax and embedded RDF-metadata.

1 Motivation

In early 2007 the Linking Open Data (LOD) community project has been launched within the W3C Semantic Web Education and Outreach (SWEO) group. The LOD community project bootstraps the Semantic Web by publishing datasets in RDF [1] on the Web. By creating large numbers of typed links between datasets [2, 3] Semantic Web application development is fostered.

Several issues emerge when building an LOD dataset; from the schema level—that is how to map from, e.g., a relational schema to an RDF Schema—to the proper and meaningful assignments of URIs to entities. One key success factor is the used interlinking method. Several approaches exist for semantically linking data.

With riese ("RDFizing and Interlinking the EuroStat Data Set Effort") [4] we have contributed to the LOD cloud by adding the Eurostat data. The implementation of riese heavily depends on scripting languages such as PHP and JavaScript. In this paper we report on our findings when implementing the riese dataset. We introduced a new way of enriching datasets called "User Contributed Interlinking" (UCI), which is a Wiki-style approach enabling users to add semantic (that is: typed) links between data items on a URI-basis.

The paper is structured as follows: In section 2 we briefly introduce the LOD principles and discuss the current state of the LOD datasets. Then, in section 3 we explain the riese implementation, including its UCI-interface and Web 2.0 issues. In 4 we discuss the generalisation of the UCI. Finally, we conclude our findings in section 5.

2 A Linking Open Data Overview

According to Tim Berners-Lee[3] it is desirable to interlink datasets, thus allowing the discovery of more data. Interlinking is one of the key factors that made the (hypertext) Web so successful. The same principle holds for the Semantic Web; rather than creating (untyped) hyperlinks between different documents, semantic links are used to interlink data items from different sources.

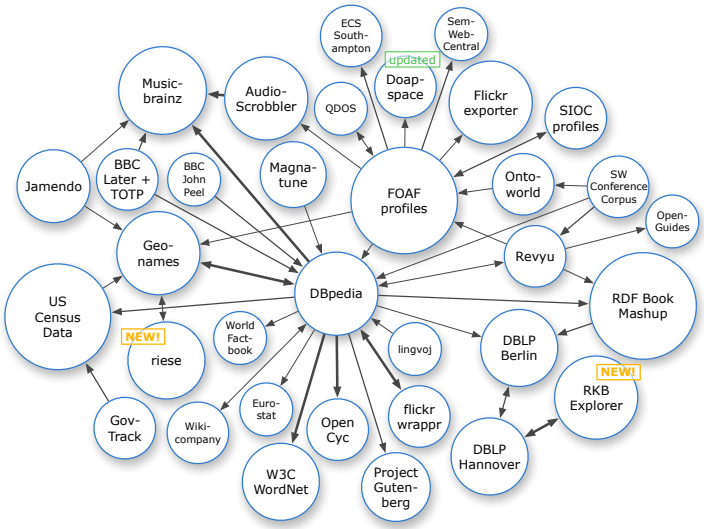

Fig. 1. The Linking Open Data dataset cloud in early 2008.

The Linking Open Data (LOD) project [2] is an open, collaborative effort carried out in the realm of the W3C SWEO[4] community projects initiative. Notable landmarks within the LOD community project include the publication of common, real-world datasets such as Wikipedia, Geonames and Musicbrainz. Currently, the project includes over 30 different datasets (cf. Fig. 1, by courtesy of Richard Cyganiak[5]), ranging from rather centralised ones (such as DBpedia) to those that are very distributed (cf. the FOAF-o-sphere).

From one billion triples and 250k links in mid-2007 the LOD dataset has grown to over two billion triples and 3 million links in early 2008, representing a steadily growing, open implementation of the Linked Data principles detailed below. It should be noted that we do not view this dataset as a fixed, delineated

[3] `http://www.w3.org/DesignIssues/LinkedData.html`
[4] `http://www.w3.org/2001/sw/sweo/`
[5] `http://richard.cyganiak.de/2007/10/lod/`

or closed entity, but rather a snapshot of a major data ecosystem within the Semantic Web at this point in time.

Linked Data Principles The linked data principles read as follows:

1. All items should be identified using *URI references* (URIrefs)[6], which implies that ideally no blank nodes are used[7];
2. All URIrefs should be *dereferenceable*—using HTTP URIs allows looking up the items identified through URIrefs; see also the so called "http-range-14 TAG finding"[8]);
3. When looking up an URIref—that is, a property is interpreted as a hyperlink—it leads to more data, which is usually referred to as the follow-your-nose principle [5];
4. Links to other URIrefs should be included in order to enable the discovery of more data.

Interlinking [6] describes how to publish linked data, and further discusses the two basic approaches for creating links to other datasets. Generally speaking, the RDF links can either be set manually or generated by automated linking algorithms for large datasets. For the latter case Raimond et.al. [7] have shown that simple interlinking algorithms produce rather poor results.

Naive approaches trying to perform a simple literal lookup are likely to fail; for instance, when trying to interlink data from the geographical domain with Geonames it is possible to do a simple literal lookup using the search facility provided by Geonames. However, when querying for the city Vienna almost 20 results will be returned as there exist that many cities named Vienna around the world. Advanced approaches such as described in [7] are needed to disambiguate similar matches and finally create appropriate interlinks. Still, there is no guarantee that the automatically generated interlinks are truly relevant. Moreover the automated process is also restricted to predefined datasets implying that only a subset of the data available on the Semantic Web is considered when looking for potential interlinks.

3 riese: Scripting LOD for humans and machines

With riese (launched in early 2008), we aim at offering a Semantic Web version of the publicly accessible data provided by the Eurostat data source, for both humans and machines. We currently serve some 3.6 million RDF triple in total, including the interlinking to Geonames data. There is ongoing work to cover the total Eurostat data (yielding some 4 billion RDF triple) and extending the interlinks to DBpedia, WordNet and other LOD data sets.

In Fig. 2 the riese system architecture is depicted. The data from Eurostat is converted into RDF/XML using SWI-Prolog, and dumped into the file system.

[6] http://www.w3.org/TR/rdf-concepts/#section-Graph-URIref
[7] http://iandavis.com/blog/2007/03/bnodes-out
[8] http://www.w3.org/2001/tag/doc/httpRange-14/HttpRange-14.html

This is to say that for each table—in tab separated values (TSV) format—from the Eurostat download[9] a corresponding RDF/XML (`content.rdf`) file, holding the statistical data, exists. The riese core schema is modelled using RDF-Schema [8] and comprises three main classes: `riese:Dataset`, `riese:Item` and `riese:Dimension`. A dataset is the logical container of either more sub-datasets (related via `skos:narrower`) or data items. We refer to [4] for further details on the modelling issues of the schema.

An Apache 2 Server along with a set of PHP scripts is used to render the pages in XHTML+RDFa [9]. The riese front-end is very light-weight; some 450 LOC in PHP and ca. 130 LOC in JavaScript were necessary to create a pleasant yet functional Web-based user interface.

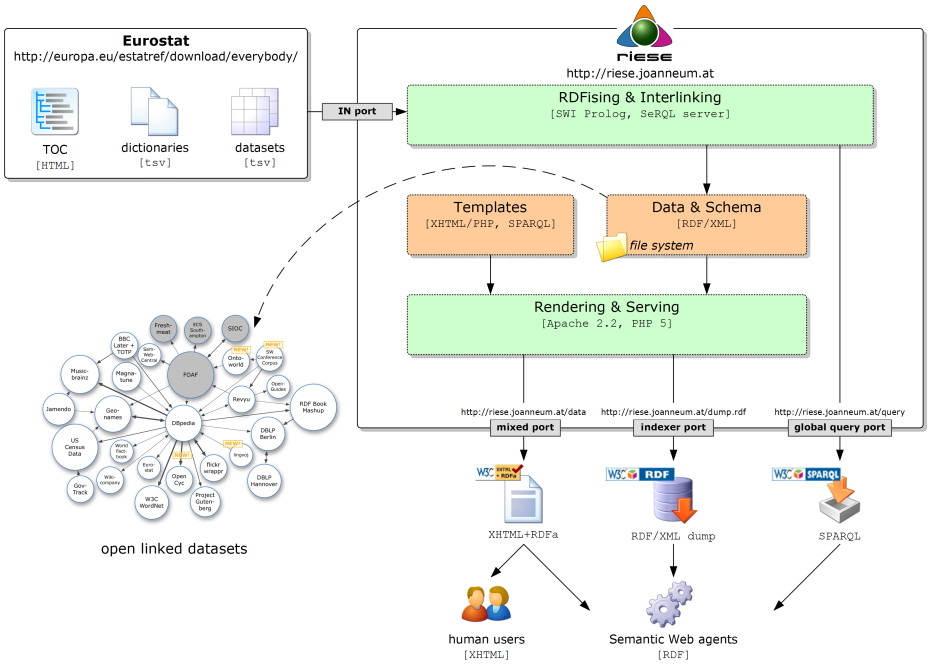

Fig. 2. The system architecture of riese.

Additional to the statistical data available for both humans and machines, riese offers another novel feature: we have implemented a User Contributed Interlinking (UCI) interface, discussed in the following.

[9] `http://europa.eu/estatref/download/everybody/`

3.1 UCI in riese

The User Contributed Interlinking (UCI) part of riese, the UCI-interface, can be understood as an agent in the sense of [10]. The UCI-interface allows to list, add, and remove user-contributed semantic links from each of the statistical data items[10].

Operation	Query String
list semantic links of the data item `sURI`	`?src=sURI`
add a semantic link to the data item `sURI`	`?src=sURI&property=pURI&target=tURI`
remove a semantic link from the data item `sURI`	`?src=sURI&property=pURI&target=tURI&remove`

Table 1. Supported operations of the UCI-interface.

The operations supported by the current version of the UCI-interface are listed in Table 1. Note that the base service URI `http://riese.joanneum.at/interlinking/uci-interface.php` is assumed. With an additional `format` parameter the output format can be controlled. The default format is XHTML, an RDF/XML representation can be obtained using `format=RDF`.

To avoid concurrent editing a simple lock mechanism has been implemented. In case two users simultaneously want to add a semantic link to a data item, an according "please-hold-the-line" message is displayed.

It has to be noted that the UCI data is kept in a separate document—that is, a separate RDF/XML document, `uci-store.rdf`, per data item—in order to allow updates independently from statistical-data updates.

```
@PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@PREFIX foaf: <http://xmlns.com/foaf/0.1/> .
@PREFIX riesed: <http://riese.joanneum.at/data/>.

<riesed:economy>
 rdfs:seeAlso <http://www.unece.org/Welcome.html> ;
 foaf:topic <http://dbpedia.org/resource/Economy> .
```

Listing 1.1. An example result from a UCI query.

To obtain, for example, an RDF representation of the UCI data for the data item `http://riese.joanneum.at/data/economy`, one would use the query

[10] `http://riese.joanneum.at/data/`

string `?src=http://riese.joanneum.at/data/economy/&format=RDF`. The result (rendered in RDF/N3 for better readability, here) would then be as shown in listing 1.1.

The HTTP-GET-interface of the UCI itself contains some 270 lines of code (LOC) in PHP, extensively making use of the RAP library[11].

UCI User Interface On the client side we have implemented an user interface that controls the UCI-interface using Ajax (the UCI-UI). The Yahoo! User Interface Library (YUI)[12] has been utilised for panels, events, etc. but also for the asynchronous communication. The UCI data is merged into the UCI user interface at rendering time. Fig. 3 shows the UCI user interface "launch pad": For each data item a user may choose to add semantic links using the "I know more" button, effectively launching the UCI-UI.

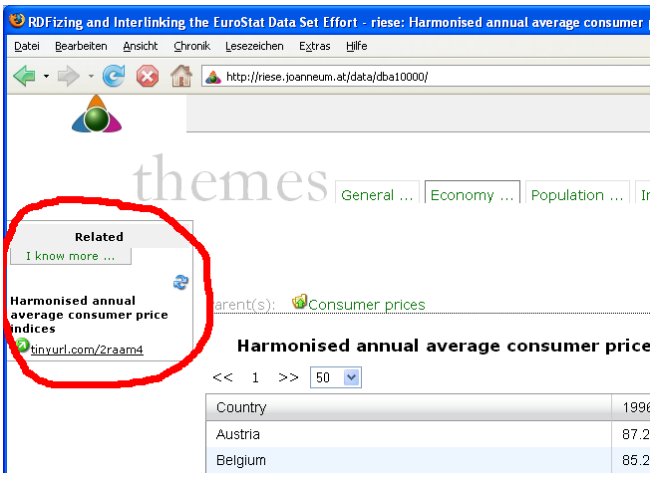

Fig. 3. UCI in riese - I.

In Fig. 4 the main UCI panel is depicted. Users can view, add, and remove semantic links with it. Note how the subject of the RDF statement is implicitly set to the data item from which it has been fired. Currently three semantic link types (properties) are supported (`owl:sameAs`, `rds:seeAlso`, and `foaf:topic`). We decided to control this part of the RDF statement as well strictly to (i) make it easier to use for the average user from the street, and (ii) to avoid issues when following-your-nose. Finally, the object of the RDF statement is the open part of the UCI data. With open we mean that is is up to the user to determine what URI to paste in. However, people are encouraged to use URIs pointing to RDF

[11] `http://www4.wiwiss.fu-berlin.de/bizer/rdfapi/`
[12] `http://developer.yahoo.com/yui/`

Fig. 4. UCI in riese - II.

(or GRDDL-able) resources. The UCI implementation is still in an early stage; further investigations both regarding scalability and usability are under way.

3.2 Issues in Web (2.0) Environments

Issues with embedded metadata and Ajax As described in [11] issues such as the do-not-repeat-yourself principle, the locality of structured data, or self-containment of descriptions need to be addressed when embedding metadata in (X)HTML. As these are generic issues, they are not limited to a specific methodology or technology, such as microformats[13], eRDF[14], or RDFa. For a deeper discussion of these issues the reader is referred to [12].

We have encountered issues with the in-place creation of RDFa in the utilised Ajax framework (YUI). Whenever rendering the metadata—expressed in RDFa, in our case—directly in the DOM, a non-DOM-based extractor is not aware of the RDF, hence unable to make use out of it. Take for example the RDFaDistiller[15], a REST-based, conforming RDFa-processor. When RDFaDistiller fetches the content from a data item it is not able to access the DOM-only parts, hence they are lost. This seems to be a general problem when using embedded metadata along with dynamic content. We are not aware of a fix allowing a generic solution. We note, however, that for example in a Last Call comment to RDFa this is has been recorded as a known issue to be addressed in future versions of this standard[16].

Access of Semantic Web data sources Another integration issue turned out to be the access of RDF-based resources. The use case in riese reads as follows: When new data is available, one way to signal this is to subscribe to a news feed. We

[13] http://microformats.org/
[14] http://research.talis.com/2005/erdf/wiki/Main/RdfInHtml
[15] http://www.w3.org/2007/08/pyRdfa/
[16] http://www.w3.org/2006/07/SWD/track/issues/114

chose Atom [13] as the news feed format, as an corresponding RDF vocabulary (AtomOwl [14]) exists.

```
<body about="http://riese.joanneum.at/updates" instanceof="awol:Feed">
 <div rel="awol:title" instanceof="awol:Content">
  <span property="awol:body">updates</span>
 </div>
 <div id="main-updates">
  <ul rel="awol:entry" instanceof="awol:Entry">
   <li rel="awol:title" instanceof="awol:Content">
    <span property="awol:body">Compensation of employees - NACE J-K -
        Current prices - Millions of euro - SA</span>:
    <span rel="awol:link" instanceof="awol:Link">
     <a rel="awol:to" href="http://riese.joanneum.at/data/na075">
     http://riese.joanneum.at/data/na075</a>
     </span>
   </li>
  </ul>
```

Listing 1.2. An AtomOwl data update example in XHTML+RDFa.

On the riese updates page (`http://riese.joanneum.at/updates/`) the data news feed is made available in AtomOwl. The AtomOwl feed in turn is serialised as XHTML+RDFa; see listing 1.2 for an excerpt of the updates page.

Using AtomOwl over XHTML+RDFa allows both humans and machines to consume the data updates properly. A human user directly accessing the page is able to view the updates, a Semantic Web agent capable of understanding XHTML+RDFa can process the feed entries for its purposes. A real-world example of how to use the AtomOwl-feed is provided in the following. In this experiment we have programmed SPARQLBot[17] to access and query the AtomOwl embedded in the riese updates page. SPARQLBot offers a Web-based interface to define commands, which in turn maps to a SPARQL query (shown in listing 1.3).

```
PREFIX aowl: <http://bblfish.net/work/atom-owl/2006-06-06/#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT DISTINCT ?headline ?feed WHERE {
?feed rdf:type aowl:Feed ;
        aowl:entry ?entry .
?entry  aowl:title ?eTitle .
?eTitle aowl:body ?headline .
}
LIMIT 10
```

Listing 1.3. A SPARQL query for data updates on riese.

[17] `http://semsol.org/semcamp/sparqlbot`

Eventually, the same procedure can be applied to other scenarios, for example, when attempting to consume news feeds in an online news-reader, such as netvibes.com. It can be seen that certain indirections are necessary, however we are confident that with the growing support of Semantic Web technologies—as recently indicated by Yahoo![18]—the burdens are likely to vanish.

4 Towards Generalising User Contributed Interlinking

With the User Contributed Interlinking (UCI) we have proposed a novel approach for creating high-quality interlinks by relying on the users. The UCI approach is motivated by the observation that generic, template-based algorithms (such as described in [7]) are limited regarding the *quality* of the typed links.

For large datasets such as riese where the entire European statistics are brought to the Semantic Web it might appear impractical at first sight to manually generate interlinks to other datasets. It is obvious that it is not feasible to have one person dedicated to manually looking for adequate related sources. However, by applying the Wiki-principle we want to initiate a crowdsourcing process that encourages users to contribute to linked datasets with similar enthusiasm as they already show in the case of Wikipedia. It has to be noted that the proposed UCI-feature is in an early stage of development and the first of its kind. The current implementation as it can be found in riese is meant to bootstrap the community-involvement in the area of linked datasets. It should be adapted to other datasets as well. Based on the experiences gained with the first release of UCI the system and the related processes will be refined. User acceptance is the critical success factor of UCI and therefore we aim at implementing as many of the best practices of Wikipedia as possible.

Sanger [15] was actively involved in the beginning of Wikipedia and has identified several factors that led to the great success of the platform such as openness and ease of editing. By inviting everybody to contribute we clearly highlight the openness of UCI. In addition we are working on enhancing the user experience by constantly improving the user interface design and keeping the user requirements at an absolute minimum as for instance no registration is required for using the UCI.

One of the disadvantages of common Wikis as identified in [16] is the limitation that "Wiki content is generally not available in a machine-processable format". With UCI we directly address this issue as the target outcome RDF is machine-processable per se. There are nevertheless still challenges left, such as reaching a critical mass of contributors by providing appropriate incentives or addressing data provenance issues. However, we would like to see a conversion of the strong community-engagement from Web 2.0 to the Semantic Web and contribute to the initiation of this transformation by providing useful tools such as the UCI.

As a next step, we have prototypically implemented a generalised UCI in a demonstrator called IRS (which is for **i**nterlinking of **r**esources with **s**emantics).

[18] `http://www.ysearchblog.com/archives/000527.html`

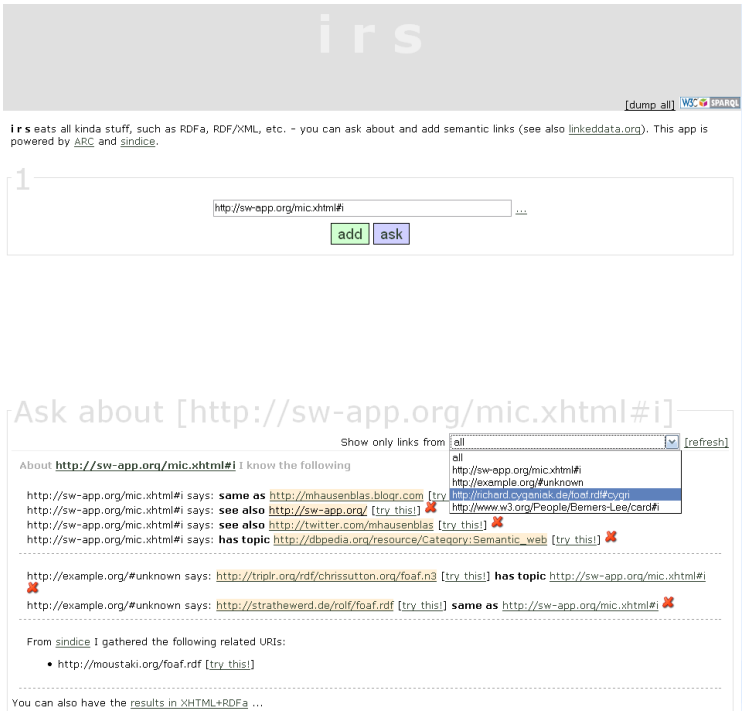

Fig. 5. A demonstrator for a generalised UCI: IRS.

The IRS demonstrator—implemented with ARC[19]—is available for testing purposes at `http://143.224.254.32/irs/`. A screen shot of IRS is shown in Fig. 5; it enables users to create semantic links (currently `owl:sameAs`, `rds:seeAlso`, and `foaf:topic`), to ask about existing links and to preview the (RDF) content. Further, a simple version of provenance tracking is offered: By placing the statements into a named graph (default is `http://example.org/#unknown`), one can track down who stated what. A simple off-the-shelf SPARQL-endpoint is also available in IRS.

5 Discussion and Conclusion

While the Semantic Web itself may be regarded as a (backbone) infrastructure, developers of Semantic Web applications have to be aware of issues arising with it.

In this paper we have presented a Wiki-style approach for user contributed (semantic) interlinking (UCI) in general, along with a discussion of tangible

[19] `http://arc.semsol.org/home`

results. First we have implemented the UCI within riese, the RDFized and interlinked version of the European statistics. We have also addressed issues emerging from using Semantic Web technologies in Web 2.0 (Ajax) environments.

With UCI we have showcased an approach potentially increasing the end-user involvement in the Semantic Web. The acceptance of such features by the community is crucial, hence we will keep working on improving the tools in order to provide an enjoyable user experience.

Due to the usage of scripting languages, an efficient and effective development of riese (and IRS, alike) was made possible. The feature-testing cycle was kept to a minimum; from a developer's perspective it was possible to focus on the important issue: functionality rather than configuration and heavy framework-study.

Acknowledgements

The research reported in this paper was carried out in two projects: the "Knowledge Space of semantic inference for automatic annotation and retrieval of multimedia content" (K-Space) project[20], partially funded under the 6th Framework Programme of the European Commission, and the "Understanding Advertising" (UAd) project[21], funded by the Austrian FIT-IT Programme.

The authors would further like to thank the following people for their lively input, discussions and support in implementation issues: Danny Ayers, Benjamin Nowack, Tom Heath, and Richard Cyganiak.

References

1. G. Klyne, J. J. Carroll, and B. McBride. RDF/XML Syntax Specification (Revised). `http://www.w3.org/TR/rdf-concepts/`, 2004.
2. C. Bizer, T. Heath, D. Ayers, and Y. Raimond. Interlinking Open Data on the Web (Poster). In *4th European Semantic Web Conference (ESWC2007)*, pages 802–815, 2007.
3. G. Tummarello, R. Delbru, and E. Oren. Sindice.com: Weaving the Open Linked Data. In *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007*, pages 552–565, 2007.
4. W. Halb, Y. Raimond, and M. Hausenblas. Building Linked Data For Both Humans and Machines. In *WWW 2008 Workshop: Linked Data on the Web (LDOW2008)*, Beijing, China, 2008.
5. L. Sauermann, R. Cyganiak, and M. Völkel. Cool URIs for the Semantic Web. W3C Editor's Draft, W3C Semantic Web Education and Outreach Interest Group., 2007.
6. C. Bizer, R. Cyganiak, and T. Heath. How to Publish Linked Data on the Web. `http://sites.wiwiss.fu-berlin.de/suhl/bizer/pub/LinkedDataTutorial/`, 2007.

[20] `http://kspace.qmul.net/`
[21] `http://www.sembase.at/index.php/UAd`

7. Y. Raimond, C. Sutton, and M. Sandler. Automatic Interlinking of Music Datasets on the Semantic Web. In *WWW 2008 Workshop: Linked Data on the Web (LDOW2008)*, Beijing, China, 2008.
8. D. Brickley and R.V. Guha. RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation, RDF Core Working Group, 2004.
9. B. Adida, M. Birbeck, S. McCarron, and S. Pemberton. RDFa in XHTML: Syntax and Processing. W3C Working Draft 21 February 2008, W3C Semantic Web Deployment Working Group, 2007.
10. D. Ayers. Graph Farming. *IEEE Internet Computing*, 12(1):80–83, 2008.
11. B. Adida. hGRDDL: Bridging microformats and RDFa. *Web Semantics: Science, Services and Agents on the World Wide Web*, 6(1):54–60, 2008.
12. M. Hausenblas, W. Slany, and D. Ayers. A Performance and Scalability Metric for Virtual RDF Graphs. In *3rd Workshop on Scripting for the Semantic Web (SFSW07)*, Innsbruck, Austria, 2007.
13. M. Nottingham and R. Sayre. The Atom Syndication Format. RFC 4287, Network Working Group, 2005.
14. D. Ayers and H. Story. AtomOwl Vocabulary Specification . Namespace Document, Atom Owl Working Group, 2006.
15. L. Sanger. The Early History of Nupedia and Wikipedia: A Memoir. In C. DiBona, M. Stone, and D. Cooper, editors, *Open Sources 2.0: The Continuing Evolution*. O'Reilly, 2005.
16. D. E. O'Leary. Wikis: 'From Each According to His Knowledge'. *Computer*, 41(2):34–41, 2008.