Approaches to Visualising Linked Data: A Survey

Aba-Sah Dadzie [a,*] and Matthew Rowe [b]

[a] *OAK Group, Department of Computer Science, Regent Court, University of Sheffield, Sheffield, United Kingdom*
E-mail: a.dadzie@dcs.shef.ac.uk
[b] *Knowledge Media Institute, The Open University, Milton Keynes, United Kingdom*
E-mail: m.c.rowe@open.ac.uk

Abstract. The uptake and consumption of Linked Data is currently restricted almost entirely to the Semantic Web community. While the utility of Linked Data to non-tech savvy web users is evident, the lack of technical knowledge and an understanding of the intricacies of the semantic technology stack limit such users in their ability to interpret and make use of the Web of Data. A key solution in overcoming this hurdle is to visualise Linked Data in a coherent and legible manner, allowing non-domain and non-technical audiences to obtain a good understanding of its structure, and therefore implicitly compose queries, identify links between resources and intuitively discover new pieces of information. In this paper we describe key requirements which the visualisation of Linked Data must fulfil in order to lower the technical barrier and make the Web of Data accessible for all. We provide an extensive survey of current efforts in the Semantic Web community with respect to our requirements, and identify the potential for visual support to lead to more effective, intuitive interaction of the end user with Linked Data. We conclude with the conclusions drawn from our survey and analysis, and present proposals for advancing current Linked Data visualisation efforts.

Keywords: Linked Data, Information Visualisation, Visual Analytics, User-centred Design, Users, Consumption

1. Introduction

The Web of Linked Data provides a large, distributed and interlinked network of information fragments contained within disparate datasets and provided by unique data publishers. The interest in this Web of Data has led to a *data race* where organisations – both commercial and not – have begun publishing their data in a format which is machine-readable – in RDF (Resource Description Framework [47]) – and linking this data to other external data. The utility of such linkage has enabled, for the first time, complex queries to be answered and traversals to be made through a diverse and semantically rich information network.

Such is the popularity of the Web of Linked Data, that it has now grown to a staggering size, with 38.5 billion triples now residing in the central connected cloud[1]. Making sense of such data presents a huge challenge to the research community, a challenge which is compounded further by the drive to produce data from both government and public bodies for end-user consumption [15,27,30,49,53,67]. It is evident that the ubiquity of mobile devices and the reduction in cost to produce sensors will escalate these challenges further, as more and more data is published through a

---

*Corresponding author. E-mail: a.dadzie@dcs.shef.ac.uk

[1] http://www4.wiwiss.fu-berlin.de/lodcloud

variety of means, e.g., information about events, foot fall in buildings and city centres.

The size and scale of the Web of Data presents challenges when trying to make sense of the information contained within it. A basic visualisation of the Web of Data, focusing on a resource which has a high out-degree of relationships to other data, will present the viewer with a mass of edges linking into the resource, resulting in information overload. How does an end user *make sense* of the response? How do they understand and interpret the data in a meaningful way?

The third principle of Linked Data states that "*When someone looks up a URI, provide useful information, using the standards*" [11]; therefore when a URI (Uniform Resource Identifier) is dereferenced, a response is returned according to the requester's parameters. These parameters can request an XHTML (the eXtensible HyperText Markup Language [55]) representation of the resource – in which case the information can be displayed in a Web browser, while embedding machine-readable information in RDFa [1] – or raw RDF according to a given serialisation format (e.g., XML (Extensible Markup Language [17]), Notation3 (n3) [12], Turtle [8]). In the latter case knowledge of how to use this format and interpret the information provided using it is restricted to tech-savvy end users, and in certain cases, only those who have knowledge of Semantic Web (SW) technologies. It is clear that regular (readable) Web users, so called *lay users*, who have no knowledge of RDF, nor ontologies, are inhibited in their ability to understand data returned when looking up a URI [24,29,62, among others (a.o.)].

Automatic production of Linked Data (LD) and its interweaving into the Web of Data is regularly done at a large scale. One of the central issues with large-scale LD production is the accuracy and completeness of links with other datasets. Identifying such links using the solitary RDF format of a dataset limits the reader's ability to identify any errors and incorrect links. The LD community recognise that a complete solution to this challenge may not be possible; however, visualisation of Linked Data may help to resolve this, as it enables the identification of such errors more easily, using, for instance, a graph visualisation. This will show, clearly, links between resources where those links should not exist, and allow the viewer to identify instances where links *should* exist but are missing.

Clear and coherent visualisation of Linked Data would enable accessibility to the Web of Data and encourage its use outside the SW community. To enable such uptake therefore requires Linked Data to become *usable* also by *lay users*, by providing interfaces and browsers of the Web of Data to support sense making and information exploration and discovery. Furthermore, query composition in languages such as SPARQL [60], although useful, requires understanding of a given query language's syntax and at least a basic knowledge of data content and structure. End users should be able to implicitly compose such queries without being aware of the underlying query mechanism that is used to pose the required questions.

In this paper we present a survey of existing approaches in current use for visualising Linked Data. To enable a consistent analysis of each approach we identify challenges to visualising Linked Data and the requirements which visualisation approaches must fulfil. By analysing such approaches, our survey provides, for the first time, a comprehensive discussion of Linked Data browsers, where they succeed and the inherent limitations of such tools.

We have structured this paper as follows: section 2 describes the background and context in which this survey is set. It describes the current state of the Web of Data and the end users which we expect to utilise this information network. Section 3 defines the requirements which approaches to visualising Linked Data must fulfil. Section 4 describes current approaches to browsing and visualising Linked Data, where each approach is analysed based on the requirements laid out in section 3. Section 5 discusses the findings from our survey, what functions were common across current Linked Data browsing approaches, and what limitations were also consistent. Section 6 presents the conclusions we draw from this survey.

2. Background

2.1. A basic scenario: the LOD Cloud

The scale of the Web of Data and the sheer mass of information now available hinder the ability of end users to make sense of the data. To illustrate the challenge involved, let us stop for a moment and take a high-level look at the Linking Open Data (LOD) cloud, and the popular diagram used to visualise it [22], generated from metadata extracted from the Comprehensive Knowledge Archive Network[2] (CKAN). The September 2010 version, shown in Fig. 1, contains 203 datasets, with 25 billion RDF triples and 395 million

---

[2]http://ckan.net

links. Fig. 1b overlays colour coding on the base graph in Fig. 1a. Both graphs use relative node size and edge thickness to represent the number of triples in each dataset and the links to other connected datasets respectively. The graph is centred on the *DBpedia*[3] [16] knowledge base created by extracting structured information from the online encyclopaedia Wikipedia[4], as part of the LOD effort.

An alternative visualisation of the CKAN dataset by Ed Summers[5] is shown in Fig. 2, using the *Protovis*[6] API (Application Programming Interface). Colour is used here to encode CKAN rating. To reduce occlusion only those nodes corresponding to more than 5 million triples are labelled (hovering over a node reveals hidden labels). The visualisation shows a central cluster, surrounded by six main clusters of varying size. Five outliers can be seen at the edges of the plot – these represent nodes for which links to others in the central cluster are not found, information about the data structure hidden by the very dense cloud in Fig. 1.
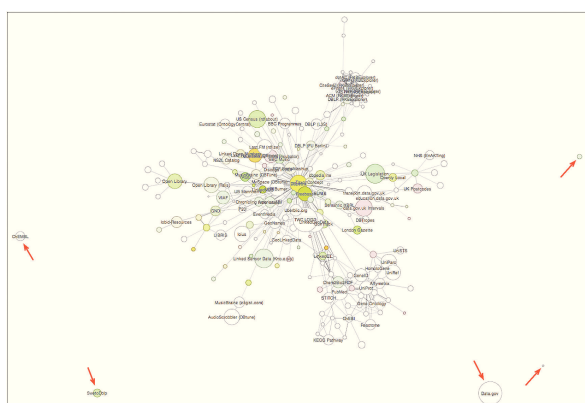

Fig. 2. The LOD cloud visualised using *Protovis*. (Credit – Ed Summers[5], update Oct 2010)

Rinke Hoekstra[7] generates an alternative view on the LOD cloud (see Fig. 3) using the *Gephi Open Graph Viz Platform* [5]. Colour and relative node size are used to encode data attributes, as is done in Figs. 1b and 2). This visualisation also displays a central cluster, again with DBpedia as the focus. Link length is also used to encode information about the data struc-

ture; two smaller clusters are drawn at a significant physical (and semantic) distance from the centre.

2.2. Challenges to consuming Linked Data

(Node-link) graph representations such as those illustrated in section 2.1 are a technique often employed for visualising inherently hierarchical and inter-Linked Data. The density of the underlying data is obvious, even with clustering of related data nodes. These high-level examples highlight one of the challenges in visualising Linked Data, as ever increasing amounts of complex data are added to the Web of Data.

Additional challenges exist: links on the Web of Data are often constructed using an automated approach. Due to their huge size false links are often created which are hard to identify – this is reflected in Hogan et al. [39] who found that applying basic reasoning over null valued, inverse functional properties for people in a dataset containing 54,836 valid RDF documents and 12,534,481 statements would infer 972,000 false `owl:sameAs` links. Manual browsing seldom reveals such links as they are buried away in the data. Effective, meaningful visualisation of the graph structure is necessary if these and other errors and noise in Linked Data are to be revealed [52]. The outliers in Fig. 2, for instance, highlight the existence of "broken links" in the LOD cloud - these may be due to missing or incorrect information in the underlying data set. Alternatively they may be legitimate islands or anomalies. Identifying such structures through manual browsing of the text is very difficult at best.

Because LOD is by definition a community effort, data heterogeneity, beyond differences in data type and content, presents yet another challenge. While encoding the actual data in RDF imposes a degree of uniformity, this does not necessarily translate to homogeneity in the quality of data capture. Fidelity and reliability of data cannot always be guaranteed or verified, especially where data provenance is not clearly specified. Differences in data type, content and granularity require a degree of translation in order to link distinct data sets correctly. A significant challenge is where data representing the same or similar concepts is encoded using different languages or terminologies [4,21,53,67,75]. SW technology provides a simple solution to this, if standard ontologies are used to translate and encode data, e.g., encoding e-mail addresses

---

[3]http://wiki.dbpedia.org
[4]http://www.wikipedia.org
[5]http://inkdroid.org/lod-graph;
See also: http://inkdroid.org/journal
[6]http://vis.stanford.edu/protovis

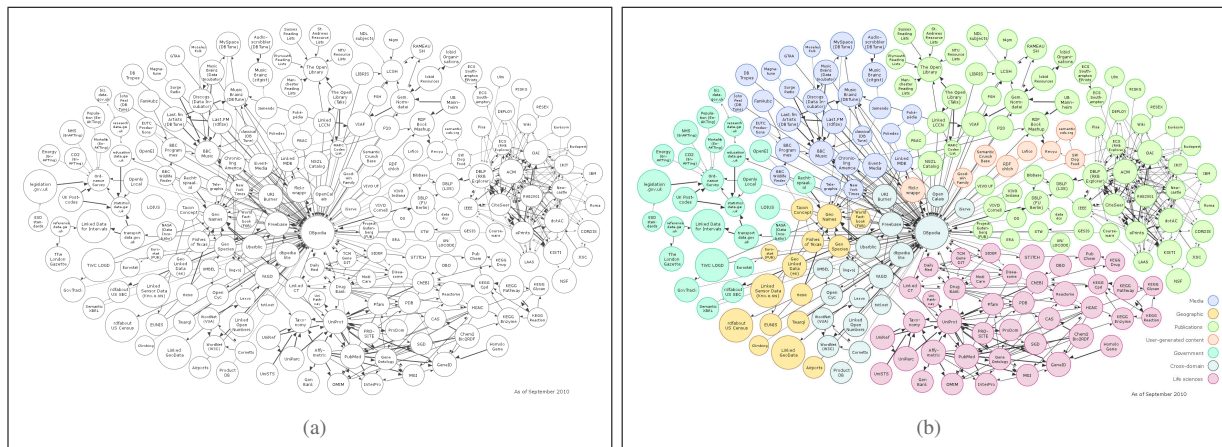[7]LOD cloud analysed with *Gephi*: http://twitpic.com/17qj1h

Fig. 1. The LOD cloud diagram [22]; the graph on the right uses colour coding to distinguish different themes in the dataset.
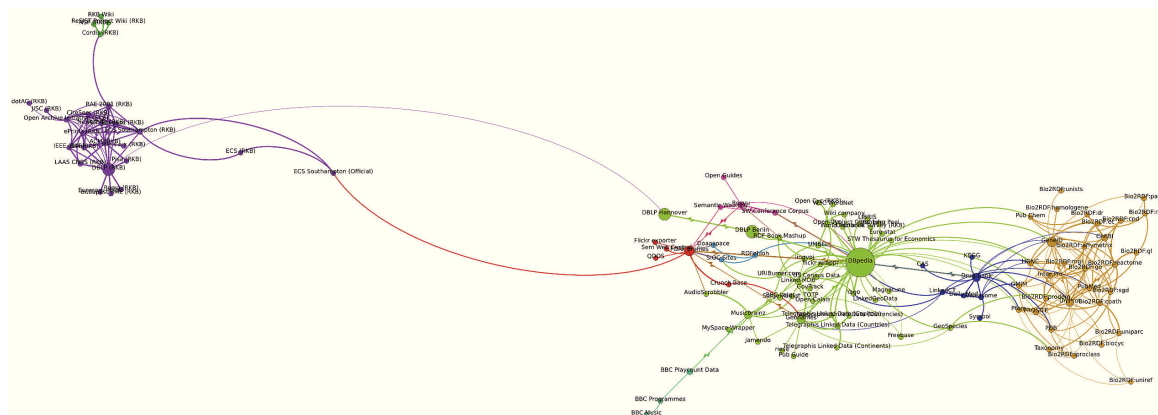

Fig. 3. Visual analysis of the LOD cloud using Gephi (Credit – Rinke Hoekstra[7], posted Mar 2010)

using `foaf:mbox`[8] rather than a custom label. Where new concepts must be defined, extending existing standards will generate a link to previously validated information, increasing the ability to reuse data.

2.2.1. Defining end users

The overall aims of Linked Data require it to be usable by both tech-savvy and mainstream end users. A number of European Union governments (e.g., the United Kingdom (UK) – *data.gov.uk*[9], *Open Government Data Austria*[10]), the United States (US) government – *Data.gov*[11], public bodies and the media (e.g., the British Broadcasting Corporation's (BBC) *Wildlife*

Finder[12] and *Music* pages[13] [49], the Guardian newspaper's *World Government data store*[14], and Link TV's *ViewChange.org*[15]) are using LOD as a means for making distributed information publicly available [29,30, 53, a.o.]. A significant challenge is found here: the (formalised) encoding of LD limits its use to those who can read and interpret RDF in the raw – a format created for machine consumption [29,33,52,57,62]. Questions posed by mainstream users would include "What is RDF?", "What is it about?", "What can I do with it?"

Different types of users will have different, albeit overlapping, requirements for the consumption of LD.

---

[8]FOAF – Friend of a Friend – Ontology: http://www.foaf-project.org;
`foaf:mbox` represents a personal Internet *mailbox*
[9]http://data.gov.uk
[10]http://gov.opendata.at
[11]http://www.data.gov

[12]http://www.bbc.co.uk/wildlifefinder
[13]http://www.bbc.co.uk/music/beta
[14]http://www.guardian.co.uk/world-government-data
[15]http://www.viewchange.org

Variety in skills, experience and environment mean end users will also face different challenges in the use of the tools available for interacting with LD. We must therefore identify the main target user groups expected to make use of this data, and what support is available to help them consume LD effectively. Shneiderman et al. [70] discuss differences in users, their skills and requirements and the tasks they carry out, and the influence this has on (perceived) usability of tools, and consequently, their adoption. Typically, broad categorisation of users breaks them down into novice, casual or intermittent users and experts [68,70]. Because no one design can pretend to meet the requirements of all user types a notion of "universal usability" [70] is adopted for tools that are expected to cater to a wide range of users; with default functionality provided to support the lowest level of expertise, but allowing some level of end user customisation of tool interfaces to better suit more advanced use. In this paper we define broadly, two main types of users, which the visualisation of Linked Data must function for:

Lay-users (mainstream) users who do not necessarily understand the intricacies of RDF and other SW technologies. Such users are computer literate and are able to find information through online resources such as Wikipedia or search engines. Lay-users will span the categories of novice to casual users, and while they may have an interest in the data they explore, only a fraction will have in-depth domain knowledge. Examples of information seeking tasks of lay-users include studying nationwide house price indices prior to buying a new home and comparing albums while buying chart music.

Tech-users expert users who understand SW and other advanced technologies, have experience in using RDF as a data format, and are able to interpret an ontological model.

A third (sub-)category of users which we must bear in mind is the *domain expert*. Such users may not necessarily have (expert) knowledge of SW technologies, but are likely to make use of sophisticated, domain-specific analysis tools to manage and interact with often very large amounts of complex, heterogeneous data. They are therefore likely to have a very good understanding of data structure and content in their domain, and bring this knowledge to guide both exploratory knowledge discovery and directed information retrieval (IR) [24,44,58,70], to enhance their ability to obtain the insight brought to bear in decision-making. For the purposes of this survey we will consider non-SW domain experts to be lay-users of Linked Data.

The sheer volume of data created on a daily basis due to advances in technology [15,37,38,44,45,71,75] means that even tech-users are severely limited in their ability to obtain a good understanding of the structure and knowledge content of these very large datasets. Another challenge is identifying suitable methods for presenting this data to all potential end users – both tech- and lay-users – so that they are able to obtain a good overview of its content and retrieve the knowledge contained with its context of use, knowledge which is necessary for confident, effective decision-making [44,58,73]. Equally important is support for intuitive navigation through the data, in order to exploit what is touted to be one of the most valuable assets of LOD – the links between diverse, heterogeneous, distributed data – to discover new, valuable and relevant information by extending existing knowledge to other related data [33].

Berners-Lee et al. [9], Karger and schraefel [43] further posit that SW data, of which LOD is a subset, enables information discovery from the user's perspective, when presented in a format suitable for the user and their task, whether this is text or visualisation-based. For example, while a software developer may need raw data in RDF or JSON[16] (JavaScript Object Notation) to feed into applications that analyse selected attributes of a data set, a formatted table is more useful to a policy analyst, and a visual representation to the lay-user browsing the results of the analysis of public health data in their community. Section 2.2.2 presents two scenarios that look, from the perspective of public data consumption, at how Linked Data presents advantages over traditional methods for collecting, encoding and sharing data. Such cases help to justify the effort required to develop more effective means for consuming Linked Data. However, we must first look at the merits the Web of Data provides over other existing large-scale data sets.

2.2.2. Why Linked Data? A public data consumption perspective

The Linked Data community[17] introduce their aims by stating:

> " *Linked Data is about using the Web to connect related data that wasn't previously linked, or us-*

---

[16] JavaScript Object Notation: http://www.json.org
[17] http://linkeddata.org

ing the Web to lower the barriers to linking data currently linked using other methods."

Data.gov is an example of a repository that provides access to heterogeneous data distributed over several areas of interest, with an aim to "empower" the US population. Data.gov currently contains more than 6.4 billion RDF triples, as part of the process of converting existing public data from traditional formats such as CSV (comma separated values) to a SW representation[18]. We illustrate the value of Linked Data to both tech- and lay-users with a subset of Data.gov that reports community health indicators visualised by county and state in the US[19], by linking to cartographic data collected by the US Census Bureau[20]. A demonstration of a set of simple methods for consuming Linked Data can be seen in a website created by Joe Meyer[21]. This site provides a good example of the differences in requirements of different types of end users in the consumption of Linked Data. To allow developers to extract the underlying data for use in other web-based applications, for example, a JSON front end is provided. We use this to extract a sub-set of the data (see Fig. 4) with the properties:

```
county,
percent age 19 to 64,
life expectancy,
percent smokers,
percent high blood pressure,
percent obese,
percent no exercise,
@state='New York'
```

Two visualisation options: *scatter plots* (Fig. 5) and *blot maps* (Fig. 6), are provided for browsing the data, in addition to (a text representation in) a sortable table (Fig. 7).

Each representation is suitable for different tasks and different end users. Linking the health data to the cartographic data set allows visualisation on a map (Fig. 6), highlighting data distribution from the perspective of geographical location. This application dis-
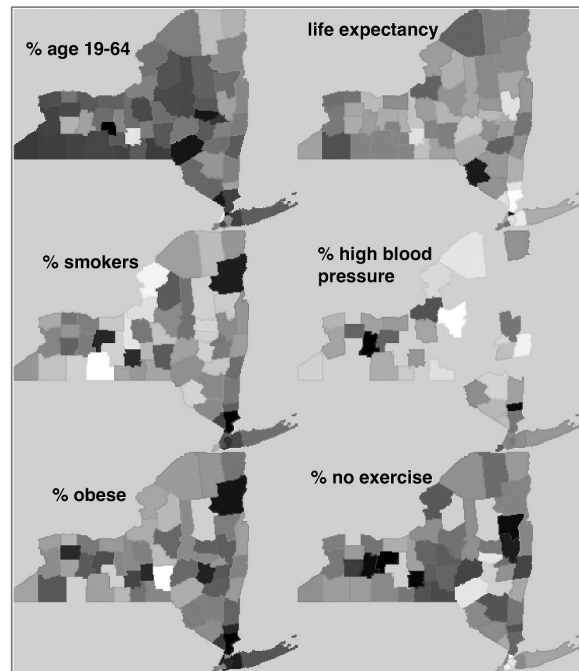
[18]See Data.gov - Semantic Web: http://www.data.gov/semantic/index

[19]See: United States Department of Health and Human Services: http://www.hhs.gov; Community Health Status Indicators Report: http://www.communityhealth.hhs.gov/homepage.aspx

[20]See: Descriptions and Metadata – U.S. Census Bureau: (Cartographic Boundary Files) http://www.census.gov/geo/www/cob/metadata.html

[21]http://health.jameyer.com/health.py/home

Fig. 6. Blot maps used to visualise data distribution, by county, of age and five health indicators (see Fig. 4) for New York state.

county	percent age 19 to 64	life expectancy	percent smokers	percent high blood pressure	percent obese	percent no exercise
Westchester	60.3	79.5	13.9	22.7	15.2	22.7
Nassau	60.3	79.2	18.1	24.5	16.6	25.3
Putnam	64.5	79.0	19.6	16.8	17.2	21.4
Tompkins	67.9	79.0	16.3	25.2	18.8	17.4
Saratoga	64.5	79.0	23.4	22.6	20.4	18.4
Queens	63.6	79.0	16.6	22.9	17.1	33.5
Rockland	59.0	79.0	17.8	28.6	18.1	28.1
Otsego	62.6	78.5	25.4		21.1	20.5
Cayuga	62.1	78.4	32.7		23.6	27.1
Tioga	61.3	78.3	24.6		21.7	21.3
Montgomery	58.9	78.2	30.4		23.5	33.6
Monroe	61.3	78.2	22.2	26.1	22.2	23.6
Madison	62.1	78.1	29.1		28.1	22.0
Dutchess	62.9	78.0	21.7	26.0	20.8	23.6
Seneca	63.2	78.0			23.2	
New York	69.0	78.0	18.7	23.1	15.9	24.5
Lewis	61.0	78.0	19.4		27.7	
Delaware	58.9	77.9	23.8		26.6	33.5
Wyoming	65.4	77.9	19.9		25.7	20.0
Suffolk	61.5	77.9	20.9	25.2	19.8	27.5
Ontario	62.1	77.9	16.1	21.3	19.3	16.9
Schenectady	60.0	77.9	21.4	30.8	25.2	23.2
Herkimer	60.6	77.8	21.1		23.6	25.7

Fig. 7. Sorted by reverse life expectancy, the data set for selected health indicators in Fig. 4 is displayed in a table.

plays only one additional attribute per map – comparison against county. The use of colour, saturation and/or patterns, and height in 2.*D (dimensions) stacked layouts, are options available for overlaying multiple attributes on a single map.

```
json_callback(
{
  "pyql" : "county,percent age 19 to 64,life expectancy,percent smokers,percent high blood pressure,percent
    obese,percent no exercise@state='New York'",
  "headers": ['county', 'percent age 19 to 64', 'life expectancy', 'percent smokers', 'percent high blood
    pressure', 'percent obese', 'percent no exercise'],
  "results" : [
    ["Erie","Cortland","Columbia","Dutchess","Delaware","Chenango","Chautauqua","Cayuga","Oswego","Putnam",
     "Allegany","Bronx","Broome","Cattaraugus","Albany","Steuben","Sullivan","Tioga","Westchester", ...
     ... "Livingston","Warren","Ulster","Wayne","Greene","Essex","Franklin","Washington","Fulton","Genesee"],
    [60.400,60.4,62.800,62.8,61.300,61.3,62.900,62.9,58.900,58.9,61.200,61.2,60.100,60.1, ... ,60.6],
    [76.900,76.9,77.100,77.1,77.400,77.4,78.000,78.0,77.900,77.9,77.000,77.0,77.700,77.7, ... ,77.8],
    [22.700,22.7,27.800,27.8,20.800,20.8,21.700,21.7,23.800,23.8,19.300,19.3,24.300,24.3, ... ,24.2],
    [26.300,26.3,null,null,26.000,26.0,null,null,30.400,30.4,null,20.300,20.3,16.800,16.8, ... ,21.3],
    [23.900,23.9,17.300,17.3,25.000,25.0,20.800,20.8,26.600,26.6,34.000,34.0,25.800,25.8, ... ,17.3],
    [23.900,23.9,22.100,22.1,27.800,27.8,23.600,23.6,33.500,33.5,22.500,22.5,25.800,25.8, ... ,26.7]
]});
```

Fig. 4. An extract of a dataset from Data.gov to JSON, to examine selected health indicators in New York state
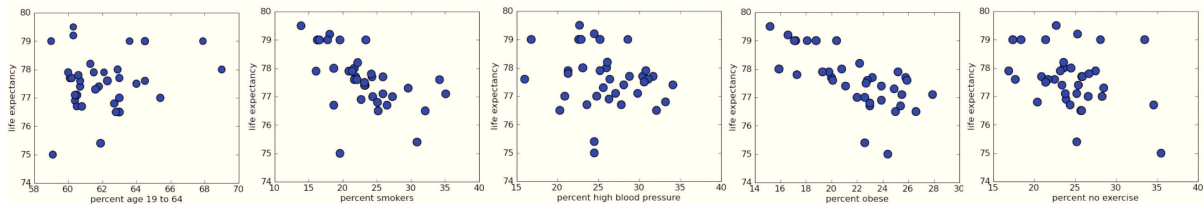

Fig. 5. The scatter plot view allows any two data attributes to be compared directly. One plot for each of the other five health indicators is compared against life expectancy for the data set in Fig. 4.

The scatter plot view allows focus on two attributes at a time. While this is useful for directly comparing two related properties it is difficult to compare multiple attributes simultaneously. We align the plots to allow the five indicators to be compared on life expectancy (see Fig. 5), to obtain a result similar to the *small multiples* technique described in [37,72]. An interactive visualisation technique that supports multi-attribute visual analysis is *parallel co-ordinates* [41] such as illustrated in [42], which are sometimes seen as an extension of a multi-attribute scatter plot. Jern et al. [42], as is done here, use multiple linked views, allowing the end user to visualise the same data from multiple perspectives, highlighting different aspects of the data and allowing a more complete overview and understanding of its content.

The table (Fig. 7) provides a quick way to inspect the underlying data. Options for sorting allow simple statistical calculations such as minimum, maximum and median values to be estimated. For the lay-user who wishes to obtain an overview of the health statistics in their community this is a simple method for obtaining hard facts for selected regions of interest (ROIs), in addition to the more general picture enabled by the visual representations of the overall data

set. Importing the table into a spreadsheet or statistical package for more detailed analysis is easily done.

While this is a very simple set of visual and text-based representations of a specific data set, the potential to extend these to more powerful analysis can be seen. The common representation for Linked Data, using a method that supports the capture also of metadata describing the data itself (as for the US Census Bureau data in the example in Fig. 4) provides not just rich data, but simplifies reuse in ways not easily achieved for data encoded in traditional formats.

For our second use case we focus on the UK government's data initiative in *data.gov.uk*: the *Research Funding Explorer*[22] (RFE) provides a currently deployed application. RDF was generated by Talis[23], working with the UK government's Department of Business Innovation and Skills[24] and other research and policy bodies in the UK. Iconomical[25] contributed

---

[22]Research Funding Explorer: http://bis.clients.talis.com
[23]http://www.talis.com
[24]http://www.bis.gov.uk
[25]http://www.iconomical.com/

to the project experience in exploring the links across different datasets, in order to identify suitable methods for visualising the Linked Data and unearthing the information contained. The aim of the project is to map the "UK's Research & Intellectual Property", to link together and make available from a single point, information about research institutions and projects, their output and funding, and companies in industry with which the research centres collaborate.

The pilot contains data on 4 research areas of excellence, displayed in the RFE using three co-ordinated views: a map that displays "investment intensity" across the different geographical regions of the UK; a bar chart that compares cumulative investment against patents published in each research area; and a stacked timeline chart showing cumulative investment. Fig. 8 shows the cumulative investment as at 2011, with a focus on regenerative medicine. The map view filters out the other domains, allowing the user to focus on the ROI. Colour coding is used to distinguish each field; this takes advantage of human perceptual ability to link information about each subject across the different views, each of which highlights a specific attribute of the data. Therefore, in the bar chart, where the other research areas are greyed out, the user automatically makes the connection to the focus ROI, based on the bar that matches the colour coding in the map. Relevant contextual information (about the other research areas) is however still available in the chart, at the periphery of the user's focus.

Fig. 8 shows a wide distribution of funding across England (for regenerative medicine), with three main clusters of high investment. Switching to the bar chart allows a more global view; significant investment is seen in three of the fields, with the highest for the focus area. The last field, RFID (Radio-Frequency Identification), however, sees very little.

The stacked timeline chart shows the investment in each domain from the year 2000, with a projection to 2018. Animation in the other two views is used to visualise the changes in trends with time for all four areas, or with a focus on a single domain. Fig. 9 shows, by zooming into the map, project activity for regenerative medicine during the peak of investment in 2009, and the drop in the number of institutions (both research and industry) by 2011, in the same geographical area. Combining the physical zoom with a semantic zoom reveals more information for the focus.

The visualisations are very useful for obtaining a high level view of the Linked Data sets and the relationships between different areas, e.g., it can be seen
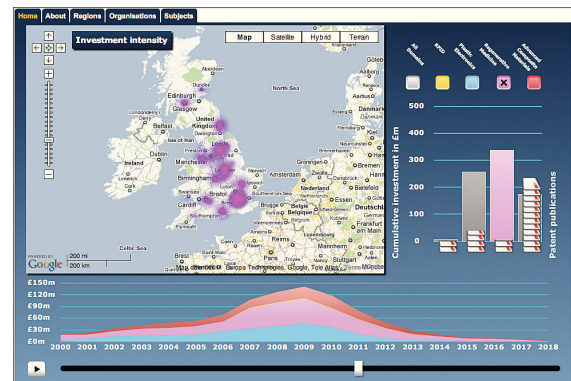

Fig. 8. The Research Funding Explorer, illustrating the use of multiple, co-ordinated views to visualise Linked Data in *data.gov.uk*

Fig. 9. Zooming into the map switches to the detail for project activity in regenerative medicine (refer Fig. 8). We examine the region of highest investment intensity – the London area. Comparing 2009 (left) to 2011 (right), a drop can be seen in institutions carrying out research in the focus technology, along with the drop in investment.

that investment does not correlate with the number of patents published in all fields. A SPARQL endpoint allows the datasets to be queried (by expert users), to extract detailed information that allows users to investigate to a greater depth trends in funding, and which topics and research areas have been successful in gaining funds over different periods of time.

2.3. Summary

Numerous examples of Linked Data usage exist; governmental organisations, public bodies and the media are among key institutions that have recognised the advantages in converting legacy data stores to and publishing new information as LD, and making explicit the links between the heterogeneous, albeit related information. We have selected from these, three main use cases, to reflect different aspects and users of LD consumption outside the SW community.

The first scenario uses simple methods to visualise data published by the US government on health, linked to census information published by another public body. The visual presentation methods make no as-

sumption of technical expertise, with the main target the mainstream user, who we find to have limited support for consuming Linked Data. The second case also illustrates the use of alternative perspectives on LD, generated by governmental and public policy bodies in the UK, that highlight links to scientific research. The target here is also the mainstream, rather than the technical user, but with a narrower focus – researchers and funding bodies. The third use case, described in section 4, looks at how a large media organisation, the BBC, is harnessing the benefits of LD as a means for storing and sharing news and other relevant information that cover a wide range of uses, users and environments. As for the other two use cases, the default presentation supports the mainstream end user; and additionally, options are provided for technical experts to query the LD repositories using formal syntax, to extract selected data sub-sets for further analysis and/or to feed into other specialised applications.

These use cases, that present data in a meaningful form, only touch the surface of what Linked Data is available and how such information can be interpreted by end users. Having presented cases that highlight both the value of LD as a means of encoding and sharing distributed data and the need to develop effective, user- and task-oriented systems for consuming this data, we must identify suitable techniques for achieving this aim. We work from these cases to ground our discussion with respect to LD visualisation.

The LD initiative is still in its infancy (see the design document written in 2006 by Berners-Lee [11], and also [15]); as we will demonstrate in the following sections of this survey, applications to date for browsing Linked Data and inspecting its structure and content are therefore, not surprisingly, predominantly targeted at tech-users [52]. Considering the requirements of such users the majority of these are text-based, often simply displaying the raw RDF data. Where available descriptive labels in the RDF (e.g., from `rdfs:label` or `rdfs:comment`) may be used to provide a more human-friendly representation of the underlying data. For tech-users such representations are useful, and often preferred [9,43]. Being domain experts, stripping away the additional layers necessary for lay-users to interpret and consume effectively the complex, interlinked data, conversely helps tech-users to make optimal use of its content.

For especially large data sets, however, obtaining a good mental model of the data places a large cognitive burden on even technical users and domain experts [52,70,79]. Where visualisation is available in

tools targeted at the tech-user they are often no-frills, node-link graphs that strictly follow the structure of the corresponding RDF graph. Such visual representations are often adequate for such users [43,56], whose focus is on inspecting the structure of a known domain and/or dataset, to validate data that is manually or automatically generated, checking for and correcting errors introduced during its generation and/or conversion [52], and/or to feed data into specialised applications for further analysis [56]. Information retrieval and filtering tasks, for such users, are typically carried out using formal query syntax, which also requires knowledge of the underlying data structure.

Linked Data, encoded in RDF and most commonly returned as XML, albeit being self-describing, because of its bent toward machine interpretation, is hard for humans to parse. As mentioned in the introduction of this paper, RDF can be served to end users via XHTML with RDFa and rendered in a more human-readable form within a browser; however delivering such a response is only possible if the hosting service chooses to expose RDF in this format. The more commonly found representations – RDF using XML serialisation, the W3C recommendation [7] [see also 18,34,51], typically used to publish LD, and alternative encoding, e.g., n3, Turtle and JSON, that may also be output from SPARQL queries – restrict consumption by lay-users, whether simply browsing data to obtain an overview, deeper exploration toward knowledge discovery, delving into selected ROIs to retrieve detail or carry out analysis toward a specific end goal, or publishing new data. We use the RDF/XML format, as the motivation for our review of current Linked Data visualisation approaches and the requirement by which we assess such work.

Making Linked Data more accessible, through the use of intuitive representations that expose its knowledge content, should promote mainstream uptake and demonstrate large-scale utility. This should in turn encourage the generation of new data such that the links to existing data are highlighted and maintained, resulting in an increasingly richer, more useful and usable Web of Linked Data. Visualisation has the potential to do so; a large body of work exists on the value it brings to data exploration, analysis and the presentation of the results of analysis and the structure of the raw data itself to different audiences [44,45], by providing a spatial representation for abstract data that aids the recognition of patterns and relationships within it. In section 3 we examine the benefits this brings to LD con-

sumption, and the requirements that must be met by tools that harness visualisation for this purpose.

3. Requirements for visualisation

Section 2.2.2 illustrates not atypical cases for the consumption of Linked Data in the public domain. In Meyer's application two distinct data sets are linked together to reveal the distribution of data for health indicators of interest to the end user, using simple but informative views. The RFE links data in multiple datasets, including research, government and industrial funding and geographical data. User skills and preference often influence tool adoption and use; providing multiple options and perspectives on data, as is done in both use cases, increases the usability of each application, as different tasks can then be carried out with the tools best suited to them. Further, this highlights different aspects of the data, allowing the end user to compare the output of each view. Methods for querying the data in each case are provided, in order to update each view based on the user's current focus. A drawback here, however, is the need to know the data attributes available to query on, dependent on prior knowledge of data content or structure.

Such applications satisfy a sub-set of the requirements for the consumption of Linked Data. In addition to these often very specific solutions there is the need also for more generic browsers, that are able to take as input Linked Data of any type and from any domain and support at least basic IR, filtering and analysis tasks. These may then be supplemented with specialised tools for more advanced analysis and knowledge discovery.

3.1. 1 Picture ≈ 1K words: Benefitting from visualisation

The power and value of visualisation is seen in its ability to foster insight into and improve understanding of data, therefore enabling intuitive, effective knowledge discovery and analytical activity. This is achieved by removing the cognitive load encountered in managing the large amounts of complex, distributed, heterogeneous data common in today's technology and information-rich society, and relying instead on advanced human perceptual ability [25,37,44,58,63,69, 71].

A number of visualisation techniques are used regularly by a wide swathe of users, for myriad purposes [19,71]. Tufte [72,74] describes a number of visualisation techniques and applications in different domains and their use for a wide variety of end purposes, citing well-known historical cases such as the space shuttle Challenger disaster in 1985 and the London cholera epidemic in 1854 [see 73]. Heer et al. [37] provide a recent survey of relatively new, interactive visualisation approaches that aim to deal with large, complex data. Pousman et al. [58] describe a more artistic approach to visual analysis, modelled on artefacts encountered in every day life, in order to better support novice and casual users.

Some of the better known visualisation techniques are simple bar and pie charts, histograms, line and scatter plots, used to carry out simple statistical analysis and to report trends and summaries. Node-link tree and graph visualisations, in both 2D and 3D, are commonly employed to display hierarchically structured data such as ontologies and networks. Other visualisation types include matrices; parallel co-ordinates [41]; timeline and topology plots; map and landscape views that build on the metaphor of cartography; space-filling visualisations such as tree maps, rose diagrams, icicle, bubble and sunburst plots; iconography, including star and glyph plots [see 19,37,74, a.o.]. Some techniques, such as parallel co-ordinates, are very well suited to high-dimensional data, while others such as pie charts and scatter plots are most useful when examining only a small number of dimensions simultaneously.

It should be noted that even novel visualisation applications often employ or build on some of the simpler, well-known and used techniques in concert, as showcased by tools and APIs such as Protovis[6], IBM's *Many Eyes*[26] and *Tableau*[27]. Improvements in technology have resulted in increased support for web-based visualisation such as these three. While still limited, compared to standalone tools, web audiences, both lay- and tech-users, have encouraged the use of new, easy to interpret, online visualisations. These include text-based visualisations such as *tag clouds* (including variants such as *Wordles*[76] and *Phrase Nets*[77]), an increasingly popular method for visualising trends such as frequency. While these techniques confirm the versatility of visualisation, usability evaluation of tag clouds indicates that their ability to support effective analysis is lower than assumed [32,77]. Phrase nets, which employ some of the features used in node-link

---

[26]http://many-eyes.com
[27]http://www.tableausoftware.com

graphs, however, are more effective at displaying the relationships between nodes (in this case, words or phrases), and therefore support more effective analysis of the underlying text.

Which technique is selected to visualise a dataset is influenced by data structure and user experience and familiarity with selected tools [28,59,71], especially for non-expert users; this however means that a less than optimal choice may be made. In the case of LD, inherent data structure means that graph visualisation is often a first choice [31,43]. While this may suffice for tasks such as obtaining an overall understanding of data structure, it may not support more in-depth analysis [2] that relies on other, less obvious perspectives of the data being revealed. For example, data attributes such as time or physical location may be better displayed using temporal and topological plots respectively, while co-relation in numerical data may benefit from presentation using statistical charts and graphs. Providing multiple, usable options for consuming LD should encourage the exploration of alternative options for visualisation and analysis, and lead to an increase in ability to interact with and discover the knowledge hidden within LD.

We discuss in detail in section 3.2 requirements for LD consumption, and guidelines derived from relevant work on visualisation and complex data analysis that should lead to the design of usable, useful tools for consuming Linked Data.

3.2. Design guidelines

Having made the case for the value of visual solutions for consuming Linked Data, we examine next requirements for the design of such tools, to overcome the challenges identified for both expert and mainstream use. Visualisation tools are most effective when designed to suit particular tasks and users [2,9,62,71]; this explains the myriad tools available even within the same application domain. However, a set of basic design features is common to the visualisation of large amounts of complex data, and must be considered in tool design for both generic and more user- and domain-specific tools. In this case we can take advantage of the common format used to encode Linked Data, reducing the additional hurdle encountered in designing tools to manage different data formats; SW-based applications only need to be able to read RDF data as a starting point for development.

The challenges defined for consuming LD in section 2.2 of this paper provide the starting point from which we derive a list of requirements for the design of effective LD visualisation. While these challenges may not cover the entire scope of the LD community, they represent issues we have identified as being of importance in relation to lay-users especially, utilising Linked Data available today. Successful design, and consequently, evaluation of the usability and utility of especially novel visualisation tools present a number of challenges [2,13,19,28,36,44,59,69,71,73,74]. While we do not address these challenges in detail in this paper we derive, from studies in Information Visualisation, Visual Analytics and Human-Computer Interaction, as well as practical application of existing guidelines to the development of SW tools, such as reported in [24,26,30,33,39,54], a set of requirements for designing visualisation-based LD browsers, in order to lead to intuitive knowledge discovery and to support LD consumption both by mainstream and tech savvy-users.

We list first high level requirements for advanced user interfaces (UIs) for visualisation tools (highlighting in bold key requirements where first specified), as defined by the *visual-information-seeking mantra* in Shneiderman [69]:

1. (a) the ability to generate an **overview** of the underlying data;
   (b) support for **filtering** out less important data in order to focus on selected ROIs;
   (c) support for visualising the **detail in ROIs**.

In addition to these Shneiderman [69] further defines different tasks that information visualisation is often used to support, and the requirements for each of these. We include here those relevant to the data and tasks involved in LD consumption, namely, support for:

2. (a) handling multi-dimensional data;
   (b) hierarchical data (or tree structures) – Linked Data is often generated as instances of (rooted, hierarchically structured) ontologies;
   (c) network data (graph structures) – high inter-linking within Linked Data may result in a graph rather than a tree; networks may not define a root and may be acyclic;
   (d) identifying/highlighting **relationships** within data;
   (e) data extraction (to a format that allows reuse by third parties (users and/or applications)).

More recent work confirms the requirements identified by Shneiderman [69], and further examines attributes of design that impact visual *presentation* and *analytics* [2,19,24,44,45,59,74, a.o.], where Amar and Stasko [2] define "representational primacy" as a focus on faithful representation of the underlying data and its structure; as opposed to (the additional benefits of) the use of different visual techniques in "analytic primacy", that overlay alternative perspectives on the data, so as to highlight different aspects of the data and guide analysis (see also [74]). To support end users, LD visualisation must go beyond simply providing yet another representation of the data; it must support the user's tasks and overall goals [19,45,74]. Amar and Stasko [2] define a set of "knowledge precepts" which they use to specify design guidelines for tools that support not just a visual presentation of data but also analytical activity that leads to an understanding of its knowledge content, allowing effective information seeking and knowledge generation, that result in confident decision-making.

Other research recognises that functionality is necessary for intuitive, exploratory navigation and also more direct guidance for especially non-domain experts. This is especially so where users have difficulty identifying and/or articulating their information seeking goals [25,28,59]. Pretorius and Wijk [59] note also the challenges that arise especially in fields where the use of visualisation, for both data presentation and to enhance analysis, is still in its early stages (such as for the LOD initiative); while end users and tool designers and developers are still defining their requirements and exploring techniques that meet these. Keim et al. [44], Thomas and Cook [71], Tufte [73] further stress the danger in claiming a "best-practice" that matches any and all situations; recommending instead the selection of visual analysis options by mapping to users and their tasks. Under such circumstances, tools that employ a "white box" approach guide users in the selection of appropriate actions during exploratory discovery and analysis, and allow the reviewing of past actions have a higher chance of successful use and adoption.

Berners-Lee [11], in his design document for LD, states the need for a standard, uniform method for identifying and retrieving data, to provide metadata describing data and its content, and to return "useful information" to end users on request, in addition to links to further, related data. Hogan et al. [39] review in detail the challenges faced in the publication of LD, by examining the design principles defined in [11]. They

examine issues that lead to errors in data and data encoding and missing information, among others. We list next, based on the design guidelines for visual information presentation and analysis, and the tasks that tech-users typically carry out, high level requirements for consuming Linked Data (indicating commonality across sets of requirements with an asterisk):

3. (a) * intuitive **navigation** through Linked Data structures;
   (b) * data **exploration**, to obtain an understanding of its structure and content;
   (c) * data exploration, to identify the links within and across data sets;
   (d) data exploration, to identify errors, noise and other anomalies in content and syntax;
   (e) * **advanced querying**, using formal query syntax, in order to obtain more direct but also more flexible, effective IR;
   (f) * **publication/syndication**, and the attendant requirements for error correction, **verification** and **validation** of new data and links to existing data;
   (g) * data extraction for reuse in other applications (without the need for a change in format for SW applications – see point 2e).

Finally, general requirements for lay-users also result in more effective visualisation design for even tech-users. Halb et al. [29,30], for instance, examine the requirements for generating both human- and machine-readable LD, especially on a very large scale, in order to provide more optimal support for especially mainstream use. Heath and Motta [35] illustrate, with a use case, design requirements for an application that supports human-friendly publication of LD (especially important for lay-users), without sacrificing the quality of the semantic data generated, and ensuring correct inter-linking to related data.

4. (a) * intuitive navigation through the large amounts of complex, multi-dimensional data;
   (b) exploratory knowledge discovery;
   (c) * support for basic to advanced querying, to support filtering and IR. This must accommodate both the end user who may have little to no knowledge of formal query syntax and more tech-savvy end users; bearing in mind that the lay-user is unlikely to have advanced knowledge of underlying data content or structure (see point 3e);
   (d) * detailed analysis of ROIs;

(e) * publication/syndication;

(f) * data extraction (to a format that allows reuse by third parties – see point 3g);

(g) **presentation** of the results of analysis to different audiences.

We will use these requirements to guide the survey of existing applications for LD (in section 4), in order to allow a clear, comparative and consistent analysis of the techniques currently in use for visualising and exploring LD, and the advantages and limitations of each. We will also use the requirements as a benchmark in the discussion of our findings (see section 5) that follows.

3.3. Data 'visualisation' vs. 'presentation'

It is necessary to distinguish between the different methods of presentation of Linked Data generally lumped under the label of (data) *visualisation*. Text-based options are commonly used, including list and tabular aggregation and presentation of data, such as found in *Sig.ma*[28] [75] and *Zitgist*[29] respectively. *Piggy Bank*[30] [40] combines faceted browsing with a textual presentation. Visual representations include pictorial, map and graph-based representations such as in *DBpedia Mobile* [6], which employs *Fresnel lenses* [57] via the *Marbles Linked Data Engine*. The *OpenLink Data Explorer*[31] provides multiple options for browsing Linked Data, including both text- and visualisation-based presentations. The *UK Parliament checker*[32] plots information about serving Members of Parliament (MPs) in a selected geographical region on a timeline overview, and provides detail for each MP using table and list views [53]. Additional information on the geographical region of interest, e.g., crime and mortality rates, is displayed using bar charts; the underlying datasets are linked via their common property – geographical location.

The aim of the survey is to review existing support for visualisation of Linked Data, and identify work still necessary to support effective, intuitive visual exploration and analysis for the different end users of Linked Data. In order to do so we must define the term '*visu-alisation*'. *Infovis.org*[33], the original repository of the *IEEE Information Visualization conferences*[34], defines '*visualisation*' as:

> "*...a process of transforming information into a visual form enabling the viewer to observe, browse, make sense, and understand the information. It typically employs computers to process the information and computer screens to view it using methods of interactive graphics, imaging, and visual design. It relies on the visual system to perceive and process the information.*"

This is in line with the definition of '*visualise*' in the online Oxford English Dictionary[35]:

> "*To form a mental vision, image, or picture of (something not visible or present to the sight, or of an abstraction); to make visible to the mind or imagination.*"

This survey focuses on the use of interactive visualisation to enable intuitive knowledge discovery from Linked Data. As discussed in section 2.3 the field has to date focused on support for tech-users, whose tasks are often best served by text-based representations [43,52, a.o.]. As a result these represent a disproportionate percentage of existing LD browsers; we will therefore also review some of the more well-known text-based tools. This will allow us also to identify where visualisation-based approaches provide additional benefits for especially lay-users, and also tech-users.

For the purposes of this survey we will refer to any application that provides interactive support for navigating through or exploring Linked Data as an LD *browser*. We will distinguish those that use a graphical, pictorial or image representation as LD *visualisations* or *visual browsers*, for static snapshots and interactive applications for exploring Linked Data respectively. As a measure of usability we will look at the ability of each representation to support the construction of a good cognitive model of the underlying data and the relationships within the data, and support the end user in carrying out their knowledge-intensive and analytical tasks.

---

[28]http://sig.ma

[29]http://dataviewer.zitgist.com

[30]http://simile.mit.edu/wiki/Piggy_Bank

[31]http://lod.openlinksw.com/ode

[32]UK Parliament checker:
http://psiusecase.enakting.org

[33]http://infovis.org

[34]IEEE InfoVis conferences: http://ieeexplore.ieee.org/servlet/opac?punumber=1000370

[35]*visualise* in the online Oxford English Dictionary: http://www.oed.com/view/Entry/224009?redirectedFrom=visualise#eid

4. Linked Data browsers

We present next a brief description of the design approach followed and/or the functionality provided by typically referenced LD browsers, as a base from which to explore the support available for end users. We group browsers into those that provide only text-based functionality (subsection 4.1) and those that include visual analysis and/or presentation options (subsection 4.2). By comparing the browsers against the requirements we list for the visualisation and analysis of Linked Data (in section 3), we will also discuss any limitations encountered.

The BBC provides another case study for the consumption of distributed, heterogeneous Linked Data [61] through the *BBC Programmes*[36] persistent web presence. The initiative aims to identify and link semantically related information owned by the BBC, to increase reusability and improve IR, both within the corporation and by its audience and third party developers. The project also aims to link BBC's legacy information to other related resources in the Web of Data. The corporation aims, through this, to improve usability of both its (readable) web pages and other resources and applications that make use of the data.

The BBC's Semantic Music Project[37], which forms a part of this initiative [49], has led to the development of *BBC Music Beta*. As a baseline from which to compare the use of different LD browsers, we attempt to resolve and retrieve information related to the URIs that point to the output of the BBC's Semantic Music project (note than when dereferenced the former is redirected to the latter):

```
http://www.bbc.co.uk/music/beta
http://www.bbc.co.uk/music
```

For those browsers that do not support browsing RDF data as unbounded sources (on the Web) or that are unable to fetch the resources pointed to by the test URIs we attempt to load (remotely, or using a local copy) the RDF at the URL (Uniform Resource Locator):

```
http://www.bbc.co.uk/music/artists/
a-z/by/a.rdf
```

If this fails we examine tool use based on the results of the closest example provided by the application developers to the topic.

4.1. Text-based presentation

We present first oft-cited LD browsers that use a textual representation of SW resources and the relationships between them. This includes tools that use presentation templates that resolve literals as part of a more human-readable, text-based layout, to embed, e.g., images in a FOAF profile (see the layout of results in the *URI Burner* in Fig. 13a). Other examples of text-based layouts include lists and tables.

4.1.1. Dipper

Dipper[38] provides a public entry point to retrieve and browse LD resources from a set of repositories stored on the Talis Platform[39]. Where available (in the RDF data) human-readable labels are used to format the results (retrieved as URIs). Dipper also enables extraction of the results of queries to RDF/XML, N-Triples[40], Turtle and JSON. Fig. 10 shows the output of a request for a sample resource in the *MusicBrainz* store.

Fig. 10. The results of a request in Dipper for a sample resource in *MusicBrainz* is used as an entry point to browsing albums in the semantic store. Labels in the RDF data are used with a basic text categorisation template to render more human-readable output.

4.1.2. Disco

The *Disco Hyperdata Browser*[41] was developed as a web browser-independent tool that browses unbounded RDF resources. Disco searches for information about a resource of interest on the SW, which is seen as "a global information space consisting of Linked Data". All processing occurs on the server, and results are presented in the client browser. The output is displayed as a table of property-value pairs, specifying the source of each resource returned. As for Dipper, where available, human-readable labels are used to format the results. The simple interface aims to support the debugging of SW sites.

The Disco public entry point was not retrievable as at the time of carrying out the review.

4.1.3. Marbles

Marbles[42] retrieves information about resources of interest by querying the SW index Sindice[43] (also used for Sig.ma), the SW search engine Falcons[44] [21], and the SW-based review site Revyu[45] [35]. Marbles improves the user experience by performing (resource-intensive) processing on the server. Fresnel lenses are used to format SW resources for presentation to end users, presented as property-value pairs in a table. Different coloured *marbles* are used to distinguish the sources of the information retrieved, which are presented as a list of URIs. Fig. 11 illustrates the output of a request for the BBC Music Beta URI, from which we browse to other interesting resources.

Marbles provides, additionally, a SPARQL endpoint. Marbles may also be installed locally, allowing information to be saved to and retrieved from local data stores.

4.1.4. Piggy Bank

Piggy Bank, part of the *SIMILE*[47] project, is a Firefox[48] web browser plug-in that enables the extraction (and conversion where required) of information from web sites, as RDF to a semantic repository [40]. Users may tag resources extracted and share the information they collect with other Piggy Bank users. This allows
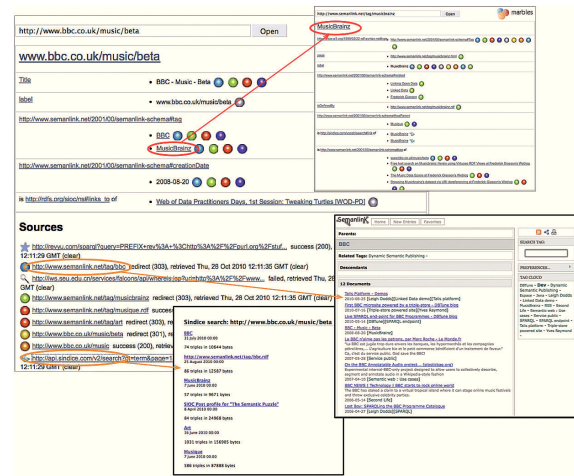

Fig. 11. The results of a request for the URI `http://www.bbc.co.uk/music/beta` are shown on the left. We then browse to the resource *MusicBrainz* (top, right). Two sources of information on the resource of interest (*BBC Music Beta*) are also retrieved from *Sindice* and *Semanlink*[46] (bottom, centre and right).

faceted search and browse across (the heterogeneous) information collected by multiple users from different data sources, based on user tags and other data properties such as date, document or resource topic. Results are presented using a list-like interface, and detail for information of interest using a table structure; Huynh et al. [40] report that presenting the information as a collection of items was more suitable for the information seeking tasks they support than would a graph representation. The reasoning behind this decision being that a graph would overload the user, thereby rendering a comprehensive view of data infeasible.

Providing the service via a web browser allows Piggy Bank to be used as part of end users' normal web browsing and IR activity. An added advantage for especially lay-users is that this allows transparent retrieval of corresponding URIs, where available, for information of interest to the user. Piggy Bank extracts information stored in non-RDF format to new RDF data, which may be stored in local or shared (remote) semantic stores, along with tags created to describe data. Users' past knowledge generation activity is therefore easily recorded and reused, e.g., as suggestions while creating new tags.

4.1.5. Sig.ma

Sig.ma – **S**emantic **I**nformation **MA**shup – views the "...Web of Data as an integrated information space" [75]. Sig.ma retrieves and integrates (often heterogeneous) Linked Data, starting from a single URI,

---

[41]Disco Hyperdata Browser: `http://www4.wiwiss.fu-berlin.de/bizer/ng4j/disco`

[42]`http://www5.wiwiss.fu-berlin.de/marbles`

[43]`http://sindice.com`

[44]`http://iws.seu.edu.cn/services/falcons/objectsearch/index.jsp`

[45]`http://revyu.com`

[46]`http://www.semanlink.net`

[47]`http://simile.mit.edu`

[48]`http://www.mozilla-europe.org/en/firefox`

by querying the Web of Data and applying machine learning and rules to the data found. Results are presented as a reorderable list of verified sources and links to potentially relevant information on the query subject; end users may confirm or reject relevance. Attributes of validated resources may also be used to filter and sort the result set, allowing users to stack information in order of highest relevance. Basic editing of the presentation templates may be carried out through a simple form.

Fig. 12. The *Sig.ma* LD browser is shown top, left (a), with query subject "BBC Music". The results of dereferencing URIs for potentially relevant information (in the right-hand pane) are shown top, right (b), and lower, left (d). Centre, right (c), shows the results of a query in Sig.ma for "BBC Music + most played artists", which corresponds to the dereferenced URI in (b). Bottom, centre, displays the results obtained for the query subject "BBC Music Beta" (e).

Support is provided for advanced querying to selected attributes of a resource. To allow reasonable response time resource retrieval is staggered. Further, repeat requests for additional information may be made – these will also take any filters in place into account. Sig.ma allows extraction of search results to RDF/XML, JSON and RSS[49] (Really Simple Syndication). Permalinks may also be saved, providing shortcuts (history) to previously executed queries. Fig. 12 shows a short browsing chain which starts with a search for the phrase "BBC Music".
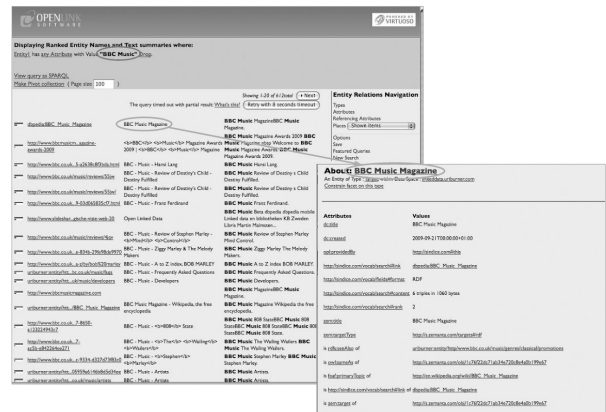
4.1.6. URI Burner

OpenLink's[50] *URI Burner*[51] service retrieves information about resources of interest by (internally) traversing the RDF graph for a resource, by matching

to standard ontologies and other relevant knowledge on the Web about the resource. The results are presented as a list of property-value pairs. Dereferenceable URIs that point to images and web pages are embedded into the HTML pages used to serve the output, in addition to displaying human-readable descriptions where stored in the RDF data.

URI Burner provides support for requests input as free text strings[52] and for looking up URIs from a text label (illustrated in Fig.13b), in addition to the norm, querying from an absolute URI (Fig.13a).
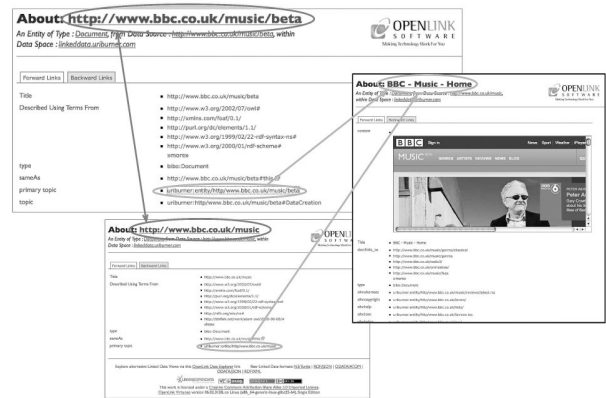
(a) Browsing from the resource URI for *BBC Music Beta*. On the right a template is used to display the result that includes the dereferenced URI (that retrieves the image) embedded in the output.

(b) Browsing the Web of data from the text query "BBC Music"; URIBurner employs *Precision Find* to find the closest matching URI and returns a ranked result set as a table of entity names with corresponding URIs.

Fig. 13. Browsing the Web of Data in *URI Burner*

A SPARQL query endpoint is available. *OpenLink ISPARQL*[53], a graphical web-based UI that demonstrates query by example, is provided as a separate tool. Additional services are also provided for application developers to extract Linked Data to raw RDF (as XML and other versions including n3 and Turtle) and other formats including HTML and JSON.

4.1.7. Zitgist DataViewer

The *Zitgist DataViewer*[29] was developed to help end users manage information overload. Zitgist collates information relevant to an entity of interest and presents it to the user, as a list of property-value pairs. Large result sets are split across multiple pages, to improve response time and help manage information load. A set of templates enables adaptation of data presentation to suit the underlying data type and presentation medium (e.g., desktop or laptop web browser vs. small screen – cf. Figs. 14 and 15).
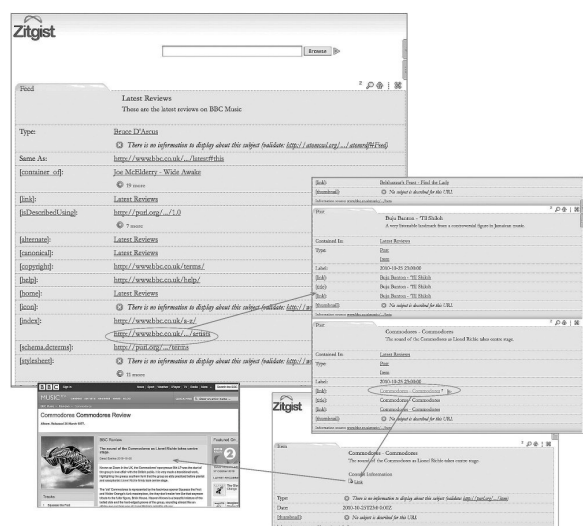

Fig. 14. The results of a request for the URI `http://www.bbc.co.uk/music/beta` is shown in the *Zitgist DataViewer*. Additional snapshots browse through the resources returned to retrieve a list of artist reviews and link finally to the *Commodores* page on the *BBC Music Beta* website.

In-built templates include for the ontologies *FOAF*[8], *DOAP*[56] (Description Of A Project), and the MO[57]

Fig. 15. A query from `http://www.bbc.co.uk/music` in the *Zitgist mini-DataViewer*[54] compares the browsing experience to that in a web browser (see Fig. 14). This time we follow the path to the review of the solo artist *Joe McElderry* and extract additional information about the entities retrieved using the *zLinks*[55] tool. This allows the MO template to be selected to format the information for the data of type `musicartist` (sub-figures b & c).

(Music Ontology). This allows information about a person, a project or a music album, for example, to be displayed in a format that is more easily interpreted by end users (see, for instance, Fig. 15 (b & c), which formats the output using the MO template). Users may edit the layout of results, by selectively embedding the resources pointed to by dereferenceable URIs in the HTML page served.

Simple filtering of sources retrieved (displayed as a list in a navigator pane) is also available. Filtering based on data attributes also provides a simple form of faceted browse/filter. Zitgist supports sharing and reuse of queries, through the use of permalinks to the HTML pages generated to serve the output; it does not however provide support for extracting the underlying (linked) data.

4.1.8. Summary

Our review found that all but one of the text-based browsers target the tech-user. As previously discussed this is not unexpected at this stage in the lifetime of the LOD initiative.

None of the tools reviewed provides a high level overview. However, support for detailed analysis of the focus is provided to at least some degree in all the browsers. Marbles additionally provides very limited highlighting of links across data, by adopting a visual paradigm – the coloured "marbles" used to distinguish different data sources in the view. This is to be expected: text does not provide a good level of support for the generation of overviews, nor the identification of relationships within data, especially on the scale dealt with in this field.

---

[53]`http://linkeddata.uriburner.com/isparql`
[54]Zitgist mini-DataViewer test on iPhone emulator:
`http://www.testiphone.com/?url=http://minidv.zitgist.com`
[55]`http://zlinks.zitgist.com`
[56]`http://trac.usefulinc.com/doap`
[57]`http://musicontology.com`

Some level of formatting and categorisation results in more human-readable representations of the RDF data output. Varying levels of (reusable) template support, such as enabled by Fresnel, result in more fine-tuned presentation of results; targeting the requirements of tech-users however means that this is fairly basic for the generic LD browsers. Their use for detailed analysis therefore still presents a challenge for the lay-user.

Genericity dominates, as expected, for the tools developed to browse the unbounded Web; with the only restriction due to the need to load and specify data repositories to search over in Dipper. Some level of support for querying via SPARQL endpoints is available, in addition to the default input via URI request. The former is the most effective way to retrieve (large amounts of) data for reuse by third parties; for tech-users this meets expectation. Support for lay users is much lower, with only two of the browsers reviewed, Sig.ma and URIBurner, providing functionality for text search, within a result set or as a starting point for retrieving/browsing Linked Data.

Tech-users are likely to generate large scale data automatically (as described in e.g., [30]), with little need for explicit support from LD browsers, beyond error identification, noise reduction and validation. Accordingly, very little support is provided for the publication and validation of new data in the tools reviewed, with only Piggy Bank, which targets lay-users, supporting intuitive, user-centred publication of Linked Data to shared repositories.

The review found that the tech-user is fairly well supported by the text-based tools that enable simple, unbounded, stepwise browsing from specified ROIs through the LOD cloud and the readable Web, with access to powerful, flexible functionality for formal querying across data sets. A significant gap however exists in support for the lay-user, especially for exploratory browsing and knowledge discovery.

We examine next a selection of visualisation-based tools, both developed for Linked Data, and generic RDF browsers able to handle Linked Data with little or no customisation, to determine the level of support available for both tech- and lay-users.

4.2. Browsers with visualisation options

The visualisation-based approaches to exploring Linked Data we review include custom applications developed specifically for Linked Data, such as *DBpedia Mobile*, *LESS*, *RelFinder* and *Tabulator*, and

browsers developed for RDF data able to support also the particular requirements of Linked Data, such as *IsaViz* and *RDF Gravity*. The discussion will highlight the application to complex information spaces (i.e., densely linked graph data), and the potential for more intuitive support by exploiting visual information seeking and knowledge discovery.

4.2.1. DBpedia Mobile

DBpedia Mobile uses geographical location as the connector between data points of interest. This allows it to support end users navigating through the real world, actively seeking or exploring resources near their physical location, to discover information about objects and events [6,16]. DBpedia Mobile uses GPS (Global Positioning System) to identify the user's position on a map and physically close objects that correspond to DBpedia resources.

A point and click interface is used to navigate to and select points of interest (available as Linked Data nodes), and the *Marbles Linked Data Engine* (see section 4.1.3) is used to retrieve related information on the selected resources. Context and type of information are used to select suitable formats (e.g., icons) for presentation. *Level of detail* functionality successively increases detail as the user moves toward an ROI; combined with filtering to hide information of low relevance to the user this allows optimal use of small screen size on mobile devices. The results are processed on the server before being returned to the user in a text overlay that contains also links to third party LD browsers.

Support is provided for keyword and concept querying via a simple query UI, and pre-specified SPARQL queries may be reused in the mobile version.

DBpedia Mobile also allows end users to publish information about their current location and any photos taken (via their FOAF profile), in addition to reviews (via Revyu) of selected resources in the vicinity (e.g., buildings, people associated with the area); this feeds into the shared DBpedia repository as new Linked Data.

4.2.2. Fenfire

Hastrup et al. [31] describe the use of interactive graph visualisation in *Fenfire* to browse Linked Data. Fenfire was developed as a generic tool, to remove the restriction to specific domains encountered in other

---

[58] DBpedia Mobile test on iPhone emulator:
`http://www.testiphone.com/?url=http://beckr.`
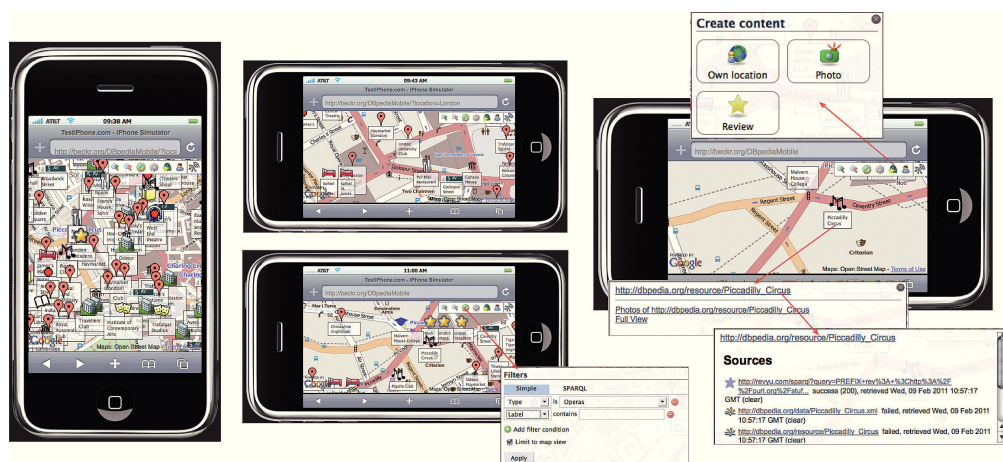`org/DBpediaMobile`

Fig. 16. Browsing central London on *DBpedia Mobile*[58]; we zoom in to an ROI and select a resource of interest. Representative icons distinguish resource types, e.g., hotels (using a bed) and music venues (musical notes). Overlaid on the snapshots are the UIs for forms-based querying and publishing new data to the LOD cloud.

RDF browsers. The application has been extended for (the more specific requirements of) browsing Linked Data, in order to support both authors and end users to verify, explore the content and make use of such data. To allow scalability to the very large amounts of data typical in this domain, Fenfire displays the detail for the focus and its immediate neighbours, and fades away the graph with distance from the focus. An alternative list view for the focus is also available. Hastrup et al. [31] note the importance of using `rdfs:labels` to provide user-readable labels for nodes in the RDF graph.

Fenfire takes input as a pointer to an RDF document(s). Where multiple documents are loaded in the view Fenfire will follow links defined between documents, and also attempt to generate dummy links between disconnected data. New graphs may be created and saved locally; changes to external documents loaded into the view may however not be published to their source, and must be saved locally.

4.2.3. IsaViz

IsaViz[59] is an interactive RDF graph browser and editor, with functionality for navigating through a 2.5D interface. Graph style sheets (GSS) [56] may be overlaid on the graph built according to the underlying RDF model, in order to provide a more intuitive view on the underlying data. Using GSS also helps to reduce the high occlusion that occurs in graph layouts for large amounts of dense, highly interlinked

data, by merging related information about entities into meaningful visual representations (e.g., using a person icon to represent a `foaf:Person` and its attributes). Fig. 17 shows the effect of applying a GSS style sheet to an RDF graph model drawn in IsaViz. IsaViz also supports the use of Fresnel lenses. Detail for ROIs is provided using text overlays in tables.

The approach taken in IsaViz provides both effective support for the tech user who is able to interact directly with the underlying RDF graph model; and the more mainstream user, on application of the presentation layers that render a more easily interpreted, albeit higher level, view.

Simple forms-based text search is provided, in addition to advanced selection and highlighting of related nodes and the links between them, using point-and-click and a form. Filters may also be applied to data attributes to remove less relevant information from the view, especially useful for reducing clutter in dense data sets. Input is a pointer to (an RDF model stored in) an RDF-encoded document (in any of the formats RDF/XML, n3 or N-Triples). IsaViz allows editing of graphs, and export to the formats above, in addition to saving the graph to an image or an SVG (Scalable Vector Graphics [23]) document. Projects may be saved and reloaded, simplifying reuse of previous analysis.

4.2.4. LESS

Auer et al. [4] aim to lower the barrier to mainstream consumption of Linked Data through the use

---

[59]`http://www.w3.org/2001/11/IsaViz`

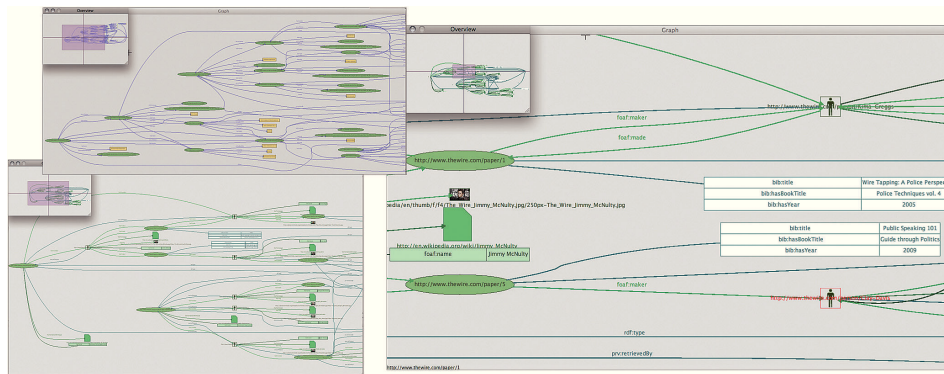[60]`http://www.w3.org/2001/11/IsaViz/gss/foaf/foaf.gss`

Fig. 17. The effect of applying the FOAF GSS stylesheet[60] to a Linked Data set describing actors in a television series using the FOAF ontology. Zooming in (right) to the formatted graph (lower left) illustrates how the template improves the ability to interpret the data content, both over the default RDF graph (top left) and the raw RDF (text).

of *LESS*[61] web-based templates to define visual representations of Linked Data. LESS templates are written using LeTL (the LESS Template Language) and rendered with PHP[62] (PHP: Hypertext Preprocessor), producing HTML, RDF, RDFa or other textual output such as JavaScript[63] or JSON. LESS templates may take as input a single data source, or synthesise multiple sources via SPARQL queries. The publicly shared web-based templates enable simplified reuse of the resulting visual output, e.g., by importing into other web pages and mobile applications' UIs. In addition to embedding as formatted HTML (as in Fig. 18), LESS may produce output in forms such as RSS.
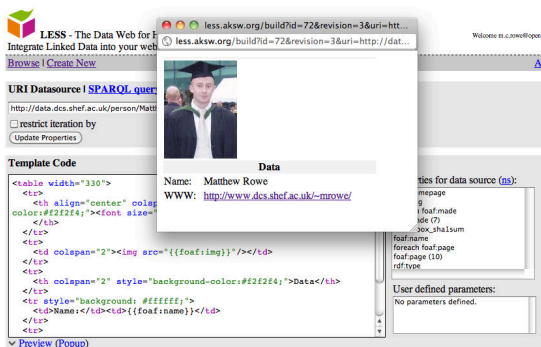

Fig. 18. Linked Data (encoded as RDF/XML) describing the `foaf:Person` "Matthew Rowe"[64] is formatted to improve readability and displayed in a web browser using a LESS template.

---

[61] http://less.aksw.org
[62] http://php.net
[63] https://developer.mozilla.org/en/JavaScript_Language_Resources
[64] http://data.dcs.shef.ac.uk/person/Matthew-Rowe

It should be noted that Auer et al. [4] do not report usability evaluation with their target (mainstream) end users. They do however indicate as future work development to provide a visual front-end for the template design and query construction, and integration with relevant web services, to better support non-technical users to both generate and consume Linked Data.

4.2.5. OpenLink Data Explorer

The OpenLink Data Explorer[31] (ODE) is a web-based RDF data browser for interacting with Linked Data. ODE requires a resolvable URI as input or a text string (for which it will attempt to locate a matching resource URI); where this does not point to a document containing RDF data, ODE attempts to extract metadata to an RDF representation and displays the result.

A number of perspectives (or views) are available: ***what:*** lists the sources from which related information has been retrieved. The result set may be grouped by resource category (e.g., `sioc:link`); ***where:*** plots data that contains geo-location information on a map view; ***when:*** plots any data with temporal attributes on an interactive timeline; ***who:*** displays information about `Person` entities using a Fresnel template based on FOAF; ***images:*** displays image data extracted from the sources retrieved; ***grid view:*** displays the triples extracted as a table of subject-predicate-object (spo) entries; ***tag cloud:*** generates a pointer to information about corresponding *delicious* tags[65] for the resource of interest; ***SVG graph:*** generates a node-link graph to display the relationships between up to the first 100 triples extracted; filters may be applied to restrict the graph to an ROI; ***navigator:*** a list of entities grouped

---

[65] http://delicious.com/tag

by type – click-through may be used to navigate to successive levels of detail for each; ***custom:*** this view allows the end user to load a Fresnel lens with which to format the display of selected resources.

Filters may be applied by selecting categories (data attributes) in the results returned. SPARQL queries generated via iSPARQL may be saved to local session files and reused in ODE.

In addition to the public web service ODE may be installed as a Firefox extension. The latter integrates the extraction of metadata and the retrieval of additional relevant sources of information into the end user's normal web browsing and IR activity (as does PiggyBank – see section 4.1.4). The user may switch between their (readable) Web browser and ODE's LD browser view, to allow direct interaction with the underlying Linked Data in the latter.
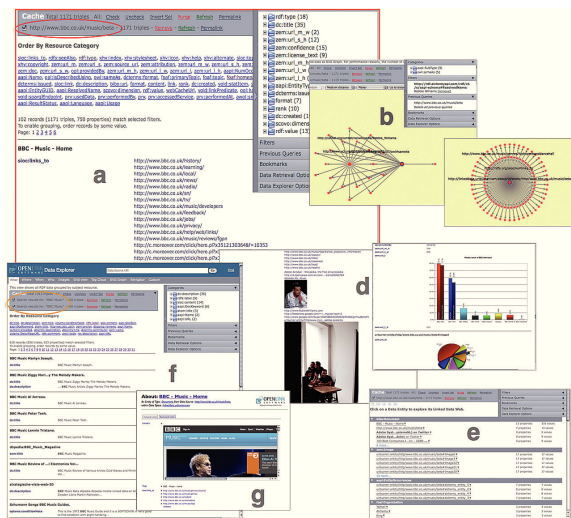

Fig. 19. The results are shown in ODE for a request for the URI `http://www.bbc.co.uk/music/beta` in (a); with alternative views – a filtered and the default RDF graphs in (b) and (c) respectively; two snapshots from the image view (d); and the navigator view in (e). The relationship to the query for the string "BBC Music" is shown bottom, left (f); and a new browsing chain is started in (g).

4.2.6. RDF Gravity

RDF Gravity[66] is a visual browser for RDF and OWL documents. It uses basic icons and colour coding to distinguish different resource types. Point and select functionality may be used to select and reposition nodes, to provide more physical space for ROIs, or

---

[66]`http://semweb.salzburgresearch.at/apps/rdf-gravity`

---

to bring distant nodes closer together to aid visual inspection. Text overlays are used to display detail about nodes related to the focus in a table. Zoom and pan are also provided, to allow a focus on the detail in ROIs; Fig. 20 shows the use of different levels of zoom for the very dense graph drawn for the list of artists with names beginning with 'A' in the BBC Music store.
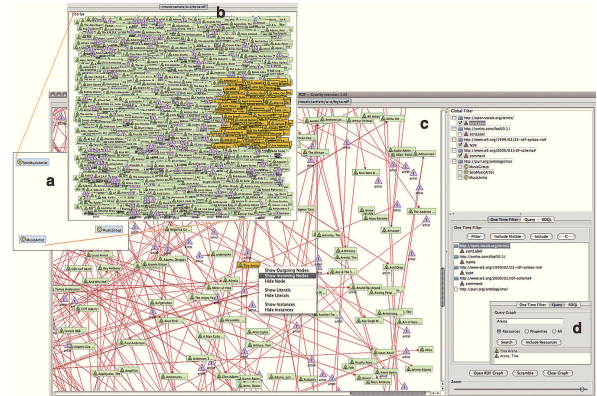

Fig. 20. Loading the URL
`http://www.bbc.co.uk/music/artists/a-z/by/a.rdf`
into RDF Gravity displays, first, the three concepts (from the MO) in the dataset – `MusicGroup`, `SoloMusicArtist` and `MusicArtist` (a). Loading the complete RDF graph (b) displays all instances of artists and groups whose names begin with 'A' (b). Even with a filter on labels to reduce occlusion (see far right), a very dense layout results. The central snapshot (c) zooms into the ROI (the nodes highlighted by selecting – yellow – in (b)) to display the links into and out of the node for the artist "Tina Arena" – see also the text query in (d) overlaid on the filters on the right.

Support for basic text search to advanced querying using formal query syntax (SPARQL and RDQL – a query language for RDF [66]) complements a visual filter used to hide information of lower relevance. This is especially useful for managing the high occlusion that occurs for very large graphs, and for analysing detail in ROIs.

4.2.7. RelFinder

RelFinder[67] aims to provide more usable interaction with Linked Data through graphical rendering of (automatically discovered) links between pairs of specified resources [38]. RelFinder focuses on the analysis of detail in ROIs, by supporting interactive restricting and/or aggregating of links drawn in the graph, based on data attributes. Colour coding is used to highlight related data properties. A data overview is maintained in a coupled table (text) view.

---

[67]`http://relfinder.dbpedia.org`

A restriction in RelFinder is that the user must supply valid entry points, a SPARQL query endpoint and the repositories to query on. This restricts use to tech-users with at least a fair knowledge of data content. Auto-completion of input labels or keywords, and an attempt to resolve these to derefenceable URI resources (that point to RDF data) in the selected repositories, coupled with validation of input URIs, support the identification of valid nodes to search from, helping to lower the entry barrier for especially lay-users, once set up for a domain. The focus on revealing interlinking within data should be useful to application developers working with domain experts, for instance, to reveal data structure and relationships across data, and highlight also incorrect linking during LD generation.

Fig. 21 illustrates a query for relationships between the entities `dbPedia:Tina_Arena`[68] and `dbPedia:Marc_Anthony`, both of whom are of type `dbPedia:MusicalArtist`.
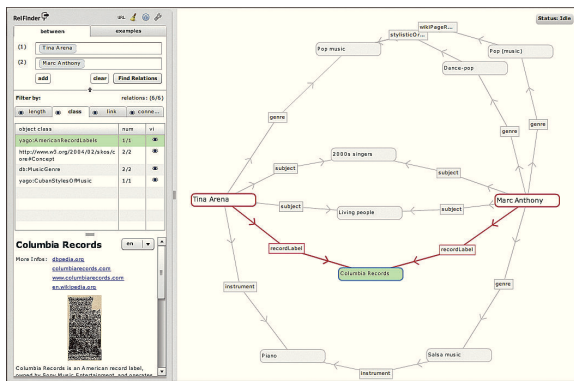

Fig. 21. RelFinder is used to search for relationships between the artists *Tina Arena* and *Marc Anthony*. The results include the `recordLabel` *Columbia Records* and the less direct relation – the `instrument` *Piano*.

4.2.8. Tabulator

Tabulator was specifically developed as a generic Linked Data browser [9]. It uses globally referenced concepts (such as FOAF) to interpret the content of Linked Data, retrieved via absolute URIs that point to RDF documents. Tabulator was developed to support, among others, the identification of effective methods for presenting the content of Linked Data resources to end users (both tech- and lay-users), via an interface that is easy to set up and learn to use. The latest ver-

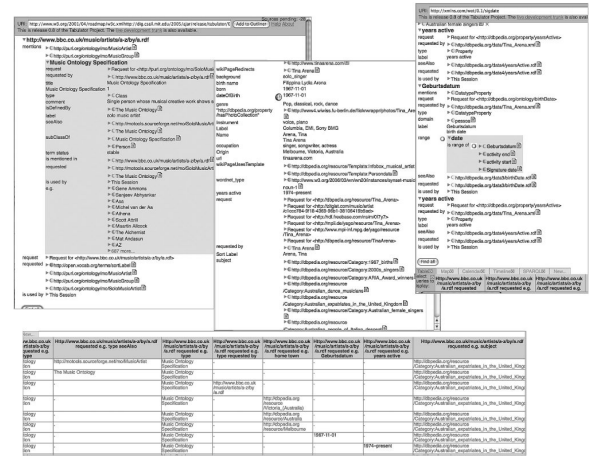sion [10] provides support for writing to the underlying data.

Fig. 22. Browsing the results returned for the URI `http://www.bbc.co.uk/music/artists/a-z/by/a.rdf` in Tabulator by clicking through the indented text index. The results for a table query are shown at the bottom.

Tabulator uses click-through a nested hierarchy (the tree view) of text labels to display increasing levels of refinement as a user browses through a dataset (see Fig. 22). Data with temporal or geo-location (encoded as latitude and longitude) properties may be displayed using a timeline or calendar view, or a map view respectively. Support for both naïve (via concept/instance selection) and expert SPARQL querying is provided; all views are updated to reflect the results of queries in any one view. Queries are automatically saved and can be reloaded.

4.2.9. Summary

Our review recognises the good level of support provided by text-based LD browsers for tech-users, but a dearth in support for lay-users (see section 4.1). The visualisation-based browsers reveal improved support for especially lay-users, for two main reasons. First of all, visualisation provides a wider range of options for encoding data attributes, making it a good candidate for effective representation of heterogeneous, multi-attribute Linked Data. The other reason for the utility and popularity of (especially well designed) visualisations is that, as discussed, they are able to take advantage of advanced perceptual ability in humans to reduce cognitive load, especially useful in complex, large-scale data analysis and knowledge retrieval. Harnessing visualisation therefore lowers the entry bar in

---

[68]Where `dbPedia:` expands to the URI for the *DBPedia* Linked Data set – `http://dbpedia.org/resource/`

the use of advanced technology for both domain experts and novice to casual (mainstream or lay) users.

Examining the top level requirements for effective visualisation support for consuming Linked Data, we find that half of the tools in this group provide a visual overview, in addition to functionality for detailed analysis of ROIs in all 8 tools reviewed. 6 out of the 8 tools provide at least some degree of functionality for highlighting links in the data, either explicitly, using graph views, and/or via simple, intuitive filtering mechanisms that highlight related data, within the context of the larger LD cloud. Availability of basic text search and/or advanced querying using formal syntax in all but 1 tool, combined with multiple views in the visualisation tools, support both guided IR and (exploratory) navigation through the data.

Although validation of the RDF model is explicitly supported only in IsaViz (which was developed as an RDF editor), the tools that provide a graph view allow end users to browse data structure, and therefore quickly recognise anomalies that would be more difficult to in large amounts of data in a textual view. Further, mapping the data to other representations that correspond to appropriate visual metaphors (e.g., geographical attributes to a map view, or temporal attributes to a timeline) reveals different perspectives on the data, in addition to highlighting any anomalies in each view.

A major limitation in the visualisation-based approaches reviewed is limited support for browsing the unbounded Web. This restricts (especially exploratory) analysis and discovery to relatively small data stores that serve as static snapshots of selected regions of the larger Web of Data. This is partly due to the fact that most of the tools available were originally designed to support the browsing of XML and/or RDF models rather than Linked Data and the Semantic Web as an unbounded resource.

Analysing requirements according to the tasks end users of LD would be expected to perform, the visualisation-based tools appear to offer more intuitive support for the exploratory knowledge discovery that is expected to guide lay-users' in consuming LD, by providing more flexible methods for (visual) navigation through the data. The interactive visualisations also provide a fair degree of support to tech-users, as advanced functionality (e.g., formal querying, inspection of the underlying data in selected ROIs using different perspectives) is provided in addition to the more intuitive but also more basic support for lay-users.

However, we recognise that even for the visualisation-based tools, support for lay-users, while largely more intuitive than for the text-based browsers, is still challenging. This is especially so where some level of technical expertise is required to set up the browsers for use for selected domains or repositories, and/or to search within data. The dearth in especially generic, basic visualisation support however reveals a gap in the field that must be filled if Linked Data is to see consumption – use, reuse and enrichment of the Linked Data cloud – by the different end users in all target communities.

5. Findings

Other RDF and Linked Data-specific browsers exist, in addition to those we specifically review. These include *SIMILE Exhibit*[69] and *Haystack*[70], *FoaF Explorer*[71], *Humboldt* [48], *LENA* [50], *Noadster* [62], *mSpace* [64], *Revyu* [35], *RKBExplorer* [27], *Semanlink* [67], and the explore mode of the *Sesame*[72] semantic store.

This survey captures the range of functionality available for browsing, querying, editing and publishing Linked Data, employing visualisation-based, as well as the more commonly available text-based UIs. Table 1 summarises the guidelines discussed in section 3 for the design of intuitive visualisation tools able to support exploratory knowledge discovery and analytic activity. For each feature we list references that present evidence (empirical and/or based on scientific theory) for its validity for and/or its contribution to design that results in tools that support intuitive, effective consumption of Linked Data.

The list is not exclusive; the ability to handle multidimensional data, for example, is implicit in LD (albeit managed in different ways), and is therefore not included. Supporting exploratory knowledge discovery is a major requirement for effective LD consumption; however this, like measuring insight or confidence in decision-making as a result of analysis, cannot be directly measured [20,25,45,71,73], but is assessed through more explicitly defined functionality such as the provision of overviews, intuitiveness of navigation, ability to examine detail in ROIs and retrieve information desired. Intuitive navigation through informa-

---

[69]http://www.simile-widgets.org/exhibit
[70]http://simile.mit.edu/hayloft
[71]http://xml.mfd-consult.dk/foaf/explorer
[72]Sesame framework: http://www.openrdf.org

tion spaces, similarly, is dependent on a number of features, including support for retracing one's steps and undoing previous actions (history), filtering out information of lower relevance in order to reduce data density and occlusion, and the use of directed querying to allow *jumping* or *porting* to a specified location in data. Exploratory navigation is also, to a large extent, influenced by the generation of effective data overviews.

The top part of Table 1 lists general guidelines for analytic activity and knowledge discovery for complex, large scale data sets such as Linked Data, while the lower section lists additional requirements derived based on tasks typically carried out in the consumption of Linked Data (note these may be applicable also to other SW data, similar domains and data types). Citations in the lower section of the table therefore focus more on empirical evidence – papers describing the implementation of tools and techniques for the use of LD and, more generally, RDF data. Bizer et al. [15], for instance, review the state of art in the LOD initiative, identifying key guidelines for the publication of new LD, and features that support the (re)use of existing LD. These include the need to maintain genericity in data encoding (i.e., maintaining the accepted standard RDF), in order to support the inclusion of any and all data, but in a form that is easily shared and reused. Other challenges that must be addressed include data verification and fusion, and the related issues in tracking provenance and ensuring data quality [15,35].

Based on what we have identified as key requirements for LD consumption (in Table 1), Tables 2 and 3 summarise functionality available in the tools we have reviewed. Tools are grouped into text-based presentation and those that are visualisation-based or provide visual representations in addition to textual output. An asterisk indicates a requirement is partially satisfied.

In assessing the extent to which these requirements are satisfied by each tool, we also look at how well they support LD consumption by lay-users. We use three main criteria to categorise a tool as targeted at the lay- or tech-user: (1) tool owners' own assessment of target user type (in publications, tool manuals and/or descriptions); (2) key tasks supported by tool design, e.g., support for information retrieval using only or predominantly formal query syntax indicates a target of tech- rather than lay-users; (3) the extent to which SW technologies are visible to the end user – tools targeted at lay-users should prevent, as far as is possible, the need for the user to understand SW technologies in order to make (optimal) use of them. For example, where a tool returns results as RDF but also provides functionality

for selecting, e.g., templates to provide more human-readable output (as in IsaViz), or the use of RDFa to embed machine-readable data within human-readable output (e.g., as described in [29]), we will categorise the tool as targeted at both tech- and lay-users.

The (relative) newness of the LOD initiative and recognised challenges in evaluating IR from semantic data[75], information visualisation and visual analytics [20,63,71], make it difficult to perform comprehensive quantitative evaluation in this field. While we do not claim our conclusions to be absolute we believe that the qualitative approach we employ – analytical inspection guided by design guidelines and best practice, where the latter has been established [see, a.o. 20,63,68,70] – reveals valuable information on usability and the potential of the techniques and approaches reviewed to enable intuitive, effective LD consumption. Our results should help to pave the way to further evaluation, which should increase in coverage as research in the field progresses, and gold standards and benchmark data and tasks become available.

In the following subsections we review, according to the distinct, high level requirements in Table 1 and section 3, whether the approaches examined are able to support intuitive exploration of Linked Data that leads to knowledge discovery and the achievement of end users' overall goals. We will discuss the support provided by the different LD browsers, and highlight gaps that must be filled in order to result in truly intuitive functionality for consuming Linked Data, especially for the lay-users who face significant challenges in the use of this rich data.

5.1. Accessing Linked Data

The tech-user is able to read and understand RDF in its raw form, and may prefer to browse long lists of URI resources and RDF triples presented with little or no formatting. Domain experts may, to a lesser extent, be able to make use of their knowledge of data in their field of expertise to browse large amounts of Linked Data in textual format, especially where labels in the RDF provide human-readable descriptions of the data. However, one of the most significant barriers to LD consumption is access by lay-users to the rich knowledge contained when presented as RDF. A challenge to exploratory discovery in the Web of Data is that most LD browsers require input in the form of absolute, dereferenceable URIs. A sub-set of these require that these also point to an RDF(-encoded) document; this requires both an understanding of SW protocols

Table 1: Summary of (visual) presentation and analytics tool design guidelines, applied to task requirements for consuming Linked Data

Design Guideline (Visual Representation & Analytics)		References
Visual presentation	the use of (often interactive) pictorial representations, graphics, images and other features that harness human perception to aid the understanding of (complex) data structure and content, to support intuitive discovery and analysis	[2,19,25,26,31, 44,45,52,56,59, 69,71]
Data overview	global, high level views on data, useful for obtaining an overall view of data structure	[19,38,44,52,56, 59,68,69,71,74]
Detail on demand	often combined with overviews, this allows the user to focus on the detail in ROIs, in order to carry out deeper analysis of data	[19,31,38,44,59, 69,71,74]
Highlight links in data	identifying the types and strength of links or relationships within a single or across multiple data sets makes a significant contribution to understanding data and knowledge discovery	[6,11,14,25,28, 29,31,34,38,44, 45,49,52,59,75]
Support for scalability	recognises the need to manage very large amounts of complex, heterogeneous, inter- and cross-Linked Data, potentially stored in remote locations	[6,19,25,29,31, 44,45,52,56,71, 74,75]
Support for querying	support for directed IR using powerful, formal query syntax such as SPARQL and RDQL, targeted at tech-users, in addition to more intuitive search employing, e.g., keyword and keyword-in-context or question-answering in a forms-based UI, or visual point-and-click UIs, that better support lay-users	[6,11,10,15,19, 28,45,52,54,70, 75]
Filtering	highlight ROIs by suppressing less relevant information. Visual, dynamic filters additionally provide a simple, intuitive method of querying, and allow previewing of more structured, formal queries	[6,19,25,44,56, 70]
History	reviewing or retracing paths, returning to specific points during visual navigation; retrieving previous analysis; undoing previous actions	[15,33,56,69]

Design Guideline (Linked Data Consumption)		References
Presentation templates	(often pre-specified) structures or designs that map data attributes or features to standard (visual) representations, e.g., a foaf:Person displayed as *Surname, Initial* with an accompanying *image/photograph*	[4,6,24,28,29, 54,56,57,75]
Entry point	default entry point into the LOD is a dereferenceable URI. Initiating browsing via keywords or other direct manipulation (DMI), e.g., clicking on a map to select a resource, lowers the entry barrier for especially lay-users and non-domain experts; this however relies on tools correctly mapping to URIs	[6,15,35,75]
Non-domain specific	generic browsers from which users can begin exploration and information retrieval, over the very wide range of LD currently available (before moving on to more sophisticated, domain- and task-specific tools)	[10,14,15,31,54, 57,75]
Faceted Search / Browse	support intuitive search/browse over multi-dimensional LD and reveal hidden inter-relationships	[40,70]
Publication / Syndication	lay-users especially require support for encoding and publishing new data into the LOD cloud; further, all end users require methods for highlighting errors, and verifying and validating new data and links to existing LD	[4,6,10,14,15, 24,29,30,31,35, 39,40,52,71]
Data Merging / Fusion	(see "Syndication") LD by definition is a community effort, with heterogeneous data from multiple sources; to reduce human cognitive load applications should link related data and fuse/merge common attributes across data, to provide an "integrated" view over the disparate data	[4,14,15,30,39, 56,71,75]
Trust & Provenance	(see "Fusion" and "Syndication") verifying data provenance influences trust, as this helps to provide measures of uncertainty, reliability and quality	[6,10,15,30,35, 54,71]
Edit underlying data	enable end users to enrich existing data with new annotations and other metadata, and correct errors	[10,14,29,31,40, 49,57]
Reusable output	encoding output using standard ontologies and vocabularies, ensuring correct interpretation and reuse, and therefore continued inter-linking and enrichment	[4,6,14,35,39, 40,49,56,57,71]

Table 2

Comparison of functionality in the text-based LD browsers.

LD Browser

Usability Criterion	Dipper	Disco	Marbles	Piggy Bank	Sig.ma	URI Burner	Zitgist
Visual presentation	–	–	–	–	–	–	–
Data overview							
Detail on demand	x*	x*	x*	x	x*	x*	x
Highlight links in data			x*	x*			
Support for scalability					x		x*
Query (formal syntax)			x*			x	
Query (forms / keyword)				x			
Filtering					x		x
History				x	x*		x*
Presentation templates	x*		x		x		x
Keyword / DMI entry point				x	x	x	
Non-domain specific		x	x	x	x	x	x
Faceted Search / Browse				x			x
Publication / Syndication				x			
Edit underlying data		x*					
Reusable output	x			x	x	x	x*
Target – Lay-users				x			
Target – Tech-users	x	x	x		x	x	x

and a knowledge of data sources and content. This explains in large part the significantly lower consumption of Linked Data outside the SW domain.

A very small portion of LD browsers however allow input as a descriptive text string, or using a DMI. DBpedia Mobile serves as a good example of the use of a custom visualisation as an entry point to the Web of Data; it infers a suitable entry point for the end user by automatically revealing resources based on geographical location. DBpedia Mobile, by making use of a familiar metaphor for navigation through unfamiliar geographical spaces, removes the technical barrier to LD consumption, enabling the (lay-)user to make use of experience in the real world to navigate the Web of Data (Yi et al. [79] explain the power in adopting metaphors in real life for visual analysis). LD browsers implemented as web browser extensions, such as Piggy Bank and the ODE Firefox plug-in, similarly infer user interest based on their current focus, and piggy-back on this to switch seamlessly to navigation through related resources in the Web of Linked Data.

We contrast the predominantly tech-oriented approach to entry into the LOD cloud to the use cases described in this paper that showcase examples of LD presentation to support access and consumption by a lay audience: first, the web-based application providing a set of perspectives from which to retrieve, browse and visualise health indicators in the US by geographical region (see section 2.2.2), by linking data in the health domain with cartographic data. The application allows end users to query the data using meaningful text labels (corresponding to data attributes) and browse the results using commonly available text (sortable table) and visual (map and scatter plot) representations.

The second example is the RFE that links UK research data, relevant information on research collected by public policy bodies, government and industrial project funding and geographical location, to provide an overview of intellectual property in research areas of excellence. Users may track research output over time and geographical location, identify collaboration across organisations, and identify potential areas of in-

Table 3

Comparison of functionality in the LD browsers with visual representations available

LD Browser

Usability Criterion	DBpedia Mobile	Fenfire	IsaViz	LESS	OpenLink	RDF Gravity	RelFinder	Tabulator
Visual presentation	x	x	x	x	x	x	x	x
RDF graph view		x	x		x	x	x	
Visual overview	x*		x			x		
Data overview							x*	
Detail on demand	x	x	x	x*	x	x	x	x
Highlight links in data		x	x		x	x	x	x*
Support for scalability	x*	x	x		x*	x*	x	
Query (formal syntax)	x*			x	x*	x	x	x
Query (forms / keyword)	x		x		x	x		x
Filtering	x	x*	x		x	x	x	
History	x		x		x			x*
Presentation templates	x		x	x	x			
Keyword / DMI entry point	x				x		x	
Non-domain specific	x*	x	x	x	x	x		x
Faceted Search / Browse								
Publication / Syndication	x	x	x					x
Edit underlying data		x*	x					x
Reusable output		x	x	x				
Target – Lay-users	x	x*	x	x*	x*	x*	x*	
Target – Tech-users		x	x	x	x	x	x	x

vestment using the DMI. Snapshots of the underlying project data may be extracted using formal query syntax.

The final use case is the BBC Music Beta project (see section 4), which is harnessing the principles of LOD to link related information across independent, previously unconnected legacy data repositories, in order to improve services to the BBC's audience. The data that has been converted to RDF may be accessed by lay-users directly from the BBC site, formatted as HTML pages, and via other services such as RSS feeds.

In each case a layer is placed over the Linked Data to present it in a format suitable for the sub-tasks (e.g., browsing, IR, analysis) and overall goals of the end user – invariably, knowledge discovery toward a specific end (e.g., review strategies for public health policies in a local council; identify industrial organisations involved in research on a specified technology; research relationships between artists classed in differ-

ent genres. Employing a more intuitive UI moreover has the advantage of lowering the cognitive load inherent in large, complex, heterogeneous data analysis, a significant factor influencing the use of LD and the adoption of tools built to consume LD in the mainstream user community.

5.2. Presentation

We have discussed the benefits of encoding Linked Data in (machine-readable) RDF. However, as also discussed, it poses significant challenges for consumption by especially lay-users (see, a.o., [62,65]). Even with processing and the application of formatting layers to provide more human-readable output, as reviewed in this paper, there still remains a large amount of output in RDF, some of which may employ unintelligible identifiers (especially to non-domain experts). Generic templates are among the most commonly used solutions, to provide an extra layer, formatting selected

data and entity types, based on, e.g., standard ontologies [4,24,54]. Fresnel lenses [57], for instance, are popular for defining implementation-independent templates for data presentation, based on ontologies such as FOAF. These, and other similar support for defining generic, reusable and extensible templates will be especially useful for the design of new, intuitive LD browsers, whether visualisation- or text-based.

Well designed, reusable and/or extensible templates that target specific users and tasks also help to manage cognitive load and support more intuitive knowledge discovery, by transforming complex data to more easily interpreted representations, by merging and clustering related information and suppressing less important data attributes [30,57,62]. We have seen that visualisation provides more options for doing so; however text-based representations also benefit from this approach.

5.2.1. Visualisation

Whether or not Linked Data is reused beyond its initial point of generation is influenced by data quality, the ability to identify links across data sets, and access to usable tools for exploratory knowledge discovery, directed IR, analysis and the presentation of the results of interaction with the data. [see, e.g., 63]. This review has found significant support for LD consumption by tech-users, using predominantly text-based presentations. However, support for lay-users is much lower, being restricted largely to custom applications and visualisations built to suit specific end goals, using selected sub-sets of the Web of Data, as seen in the case studies in section 2.2.2. Such tools tend to employ basic, commonly used techniques to generate one-off visual representations, often maps and timelines for data with geographical location and temporal attributes, respectively, and graphical plots and charts, such as scatter plots and bar and pie charts, to support statistical analysis.

Visualisation has seen lower than expected coverage in the LD application domain, especially in the development of generic LD browsers, beyond the provision of node-link graphs used to display the structure of the RDF graphs described by Linked Data. A contributory factor may be the newness of the field; a fair portion of applications are still at the prototype stage. Further, these are often built to meet the requirements of tech-users, who are still the main consumers of Linked Data, chiefly to browse data content, determine its structure, carry out error checking and validate new data. It is not surprising that functionality in LD tools is geared to these tasks.

We have however made a case for the benefits that visualisation could bring to the consumption of Linked Data, especially for mainstream use, but also for tech-users. Generic browsers are necessary to promote mainstream LD consumption. Existing visualisation-based RDF browsers such as RDF Gravity and IsaViz attempt to parse the RDF graph structure of LD and display this in node-link layouts. Such tools commonly use icons, node and edge colour and size to encode data attributes. High level data overviews are often provided, especially useful for very large data sets, in addition to support for intuitive discovery through navigation to and detailed analysis in ROIs. Clustering and level of detail filters may be provided to support interaction with especially dense data.

Another challenge in the use of LD browsers configured to browse the unbounded Web is handling responses such as "No data found in this store" or "No further information is available" for valid URIs. While the latter response implies that no information was found for the URI in question, the former corresponds to a failure to locate the resource. A particular advantage in the use of visualisation to explore unbounded data is the ability to provide a preview of the next step for not just the current focus, but also closely related information, using level of detail functionality (such as dynamic filters) that reveals more detail as a user navigates through a data space. DBpedia Mobile provides such support – as a user zooms in to an ROI or pans across the map previously hidden resources are revealed. This gives the user a preview of the knowledge in new, previously unexplored regions. If the user fails to locate expected resources in a region of the data space other information in the surrounding area may provide relevant information that may help to locate and retrieve the resources desired. Advances in technology support the generation of both simple and augmented visual layouts able to support such views, which, relying on intuitive, pre-attentive perception in humans, significantly reduce cognitive load on the end user. Greater restrictions to managing large amounts of textual information due to, a.o., screen real estate, however prevent the generation of similar detailed views in text-based representations.

5.3. Obtaining an overview

Advances in technology have resulted in ever increasing amounts of complex data; ability to manage information overload is key to the usability of data analysis and knowledge discovery tools [45]. One

technique for doing this involves the generation of data overviews. While visual overviews are most common they may take any other form suitable for the user and their task (see section 5.2, which discusses methods for general data presentation). Overviews help end users to obtain an understanding of overall data structure, and often serve as starting points for navigation to ROIs and detailed analysis [69,79]. IsaViz is the only tool that provides a dedicated (visual) overview, in a coupled pane, in addition to the central detail view (provided to some extent in all the tools reviewed). Visual overviews may be obtained in RDF Gravity and DBpedia; however, the user must switch between this and the detail in ROIs. RelFinder provides a restricted (text) data overview. A limitation of text-based browsers is the poor support in textual presentation for generating overviews – none of those we surveyed was able to provide data overviews.

Support for handling scalability, in terms of data size and dimensionality, is important for obtaining useful overviews of (large scale) Linked Data. All the tools surveyed perform some degree of processing to cluster or merge data and minimise redundancy. Where in use, presentation templates in both the text- and visualisation-based tools reviewed help to aggregate and format the information retrieved for a request, providing descriptive summaries. These use, a.o., `owl:sameAs` constructs to highlight resources that point to the same information, and `rdfs:seeAlso` to point to additional information about the resource of interest. This results in more human-readable output and ensures correct interpretation of the processed data, in addition to reducing information overload.

Multi-dimensionality in text-based tools tends to be presented in sortable tables or matrices, or grouped in (often indented) lists, allowing users to obtain, to a degree, an overview of the data. Multiple, linked views are another useful way to highlight and compare different perspectives on a data set, by focusing on a subset of the data attributes in each. Visual representations have an advantage in such cases in that they provide access to a large number of cues for encoding multiple dimensions, including variation in colour (see e.g., Fig. 1b), and representative icons (e.g., as in DBpedia Mobile, IsaViz with GSS). Icons may even be adopted as a visual cue in text-based representations – as seen in Marbles.

5.4. Navigation & exploratory discovery

In order to support discovery, especially for the lay-user who will have little knowledge about data content and structure, LD browsers must support intuitive navigation within individual data sets and across the links between data. This must be combined with strategies for managing information overload, in order that end users do not get lost, but instead are able to identify the nuggets of valuable, relevant knowledge hidden within the very large Web of highly interlinked data. The presentation of data – from high level overviews to detail in ROIs, and support for intuitive filtering, are among the most important contributors to effective navigation [62,69] through such data.

The text-based LD browsers reviewed all make use of in-built features in web browsers to provide click-through navigation from a resource to other related information. Embedding applications in a browser however often disables the history mechanism built into the forward and backward navigation buttons on their tool bars, or results in inconsistent behaviour. Dipper, LESS, Marbles, Piggy Bank and URI Burner, however, serve output as HTML pages, allowing normal use of the navigation buttons. Additionally, the tools that maintain an independent history of user actions are able to support retracing and reviewing of previous paths through the Linked Data space (e.g., Piggy Bank, ODE, by saving sessions and/or retrieving data in semantic repositories for locally installed copies).

The visualisation-based representations are able to make use of typical functionality for navigating through data layouts, with the most basic support including point and click functionality for *opening* or *jumping* to more distant resources of interest. Fenfire and IsaViz additionally support navigation using the keyboard. Pan and zoom are also features typically available for browsing to ROIs in interactive visualisation, and filters are often provided for suppressing less relevant information. Overlays may be used to display detail for ROIs or to preview related, hidden data. Such features are used to navigate through the graph layouts in Fenfire, IsaViz, RDF Gravity and RelFinder. Level of detail may be used in layouts, such as found in DBpedia Mobile, to provide increasing detail as the user moves to a previously unpopulated area of the map or zooms into ROIs.

5.5. Information retrieval

Directed information retrieval is carried out after end users obtain an understanding of data content and

structure, to enrich knowledge discovered through the initial exploration of a data space. The ability to directly query Linked Data using formal query syntax is key to successfully carrying out tech-users' normal activity. Different levels of depth may be supported, from search within local or shared, remote semantic stores, to the unbounded readable and semantic Web.

While the power and flexibility of formal querying cannot be denied, simple, intuitive filtering and IR is also important, especially for lay-users. This may be provided by, e.g., simple click-through lists of resources and data attributes used as filters, or more commonly, the use of forms. Options for ordering and grouping information by data type and property are also useful for identifying and retrieving related information, and may be used for faceted search/browse.

5.6. Detailed analysis of ROIs

Browsing of LD by definition occurs across multiple, related data sets (from a specified entry point). Each of the tools reviewed allows the inspection, and to different degrees and with different techniques, analysis of the user's (changing) ROI.

For small amounts of data, a plain text view is a popular choice for analysing detail of the unadulterated data. However, as data amount increases, and further processing and analysis become necessary, other often specialised techniques must be employed to support effective analysis of detail. Such functionality must overcome human cognitive limits that restrict the amount of information that can be processed effectively, especially where a large number of relationships must be followed to obtain a complete understanding of data.

This is where visualisation has a significant advantage (over text). It allows the use of suitable visual metaphors and a variety of visual encoding options to provide alternative perspectives that highlight selected data attributes. Another advantage is the ability to use the increased bandwidth in visualisation and perception to identify related information outside the immediate ROI, using, e.g., previews as described in sections 5.2.1 and 5.4.

5.7. Publication / syndication

Intuitive support for publishing new data into the LOD cloud is necessary to ensure high quality and reusability, and therefore promote the value and consumption of Linked Data by especially lay-users [24,

35,65]. Tech-users are likely to write custom or extend existing scripts to automatically generate and dump large-scale data into the LOD cloud, and where necessary, extract metadata from legacy data and convert it to RDF [14,16,29,39]. Lay-users on the other hand require more transparent and intuitive support to publish smaller amounts of data; weaving data syndication into users' normal IR and analysis activity is one of the more effective routes to this end [35,40]. For instance, a free text tag attached to a photograph of a cultural icon and uploaded to a shared repository as part of a tourist's review may be automatically extracted as metadata to an RDF representation and attached to the original description of tourist attractions in the geographical area, encoded as a resource in the Web of Data. DBpedia Mobile, which stands out clearly, from all the tools reviewed, as that most easily used by lay-users, takes this approach, by using the third party application Revyu [35] as a portal to the Web of Data.

Plans to enable direct publishing of LD to the Web of Data in Fenfire, and more intuitive support for template creation in LESS should provide more effective support for especially lay-users. IsaViz was developed to support editing of RDF graph models; it therefore provides extensive support for both simple and complex graphs. While lay-users are likely to encounter a relatively steep learning curve, IsaViz represents the most usable of the graph editors reviewed.

5.7.1. Data verification & validation

Both publishing new and editing existing data sets require verification and validation, i.e., checking for errors, noise and redundancy in the data. To carry out such tasks tech-users visually inspect the plain RDF or use simple graph views that display the structure of the RDF graphs described by Linked Data. Such support is available to tech-users, with tools such as IsaViz providing sophisticated functionality for viewing and editing RDF graph models, and RelFinder with support for discovering and inspecting links in small ROIs. The Sindice Inspector[73] is an example of a tool that makes use of multiple services to validate and provide information about the content pointed to by a URL. Among others, it renders an SVG graph of the RDF model returned, generates a table of triples retrieved (as spo), and presents a the response for the URL in Sig.ma.

Verification and validation tasks must however be carried out more transparently in applications targeted at lay-users; data entry forms such as provided in DB-

---

[73]http://inspector.sindice.com

pedia Mobile guide input to ensure that the free text entry is transformed to valid RDF, in addition to capturing semantic metadata describing the data input.

5.7.2. Data reuse

In addition to publishing new data to the LOD cloud, the ability to export query and analysis results (including new metadata that enriches existing LD) for reuse in other tools is a requirement for end users. A significant advantage in Linked Data is that the common data encoding format (RDF) simplifies information exchange between SW (and other) tools. Further, providing metadata about data and the reuse of standard ontologies and vocabularies increases the ability to and probability that existing LD is correctly interpreted and reused.

6. Discussion & conclusions

In this paper we have presented an analysis of current approaches used to browse and visualise Linked Data. The motivation behind this survey was to present the reader with a clear definition of the differences between such approaches, and their advantages and limitations. By identifying requirements that such approaches must address, requirements which in turn were motivated by the context of application over the Web of Data and the huge expanse of data that is to be dealt with, our analysis has identified where Linked Data may be used by both tech-savvy and lay-users.

We divided the approaches for browsing Linked Data into two types: text-based and browsers with visualisation options. Both types of approaches have their place and usage. For instance, the former allows for fine grained analysis of a given datum, whereas the latter allows the bigger picture to be seen. In essence the provision of visualisation options supports human perceptual ability [44,45,63,69]. Bombarding end users with rows of text is not effective in conveying a structural description of a resource and how it is associated with other resources in disparate datasets. However, large-scale data visualisation is not always the best solution to complex analysis; instead, presentation should be based on the requirements of the end user and their specific tasks [2,19,20,24,25,26,43,56,62,63,68,69,71,73,75].

Of the array of approaches and tools surveyed we found only two which, based on our requirements, offered a usable solution to browsing Linked Data by lay-users. The uptake of Linked Data by a mainstream audience is dependent on its utility to those outside the Semantic Web and Linked Data communities, therefore this lack of support for non-tech-savvy users could inhibit its adoption.

Analysis of the approaches did however identify several useful features that support end users – both tech-savvy and lay. Firstly, breadcrumb features, which are present in the majority of web-browser based approaches – particularly if the browser states are maintained – enable users to maintain an exploration path through the Web of Data. As a consequence, end users are able to keep track of their progress through the duration of their exploration activity, thereby providing a history of the paths that were traversed. Second, the use of templates associated with given ontological concepts transforms data returned by a URI into a legible, usable form. We found this to be particularly useful for URIs that contain many pieces of information within the instance description. When dereferenced, such URIs, depending on the Linked Data browser used, display several pieces of information, making it challenging for the end user to decipher and interpret the response.

It is evident that within the Web of Data, a chicken and egg problem exists [4,21,40,53,67]: where tools are needed to locate, retrieve and use Linked Data, and without *good quality* Linked Data, there is little motivation to design and build such tools for end users external to the Linked Data community. Mainstream consumption requires *reusability* [4,30] where data can be combined together, or the current focus of exploration saved and then re-loaded, or passed on at a later point. Heath [33] states that the sum is greater than its parts, where using the Web of Data to discover information about information enables expansion on existing knowledge, something which needs to be shared.

Formalising data using RDF principles goes a long way to obtaining a uniform representation of self-describing data, allowing the links across data to be recognised, as seen in LOD, and promoting data reuse [40], among others. This lowers the restrictions peculiar to specific domains and tasks, and increases the ability to develop more generic tools for data exploration and analysis. Using Fresnel lenses [57] and other methods for creating generic presentation templates [54,62,65], for instance, is one step toward independent declaration of the presentation of data. This will not completely remove the need for custom tools – they are better at performing specific tasks [9,59,62,70]; however the design of tools for the Semantic Web are increasingly able to provide support

for user customisation, allowing a truly user-centred perspective on analysis.

A key conclusion that we drew from our work was the limited number of Linked Data browsers currently available. We assume that this can be attributed to the infancy of this domain. Although the Semantic Web vision has been pursued for circa 10 years, it is only over the past 3 years that the production of machine-readable data has been generated at a sufficient scale that consuming it has now become a large-scale task. We therefore believe that the consumption of Linked Data has now become a major research challenge, one in which the visualisation and browsing of Linked Data will play a crucial role.

Acknowledgements

References

[1] Ben Adida, Mark Birbeck, Shane McCarron, and Steven Pemberton. RDFa in XHTML: Syntax and processing – a collection of attributes and processing rules for extending XHTML to support RDF. `http://www.w3.org/TR/rdfa-syntax`, 2008.

[2] R.A. Amar and J.T. Stasko. Knowledge precepts for design and evaluation of information visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):432–442, 2005.

[3] Lora Aroyo, Grigoris Antoniou, Eero Hyvönen, Annette ten Teije, Heiner Stuckenschmidt, Liliana Cabral, and Tania Tudorache, editors. *ESWC 2010: Proc., 7th Extended Semantic Web Conference*, 2010.

[4] Sören Auer, Raphael Doehring, and Sebastian Dietzold. LESS - template-based syndication and presentation of Linked Data. In Aroyo et al. [3], pages 211–224.

[5] Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. Gephi: An open source software for exploring and manipulating networks. In *AAAI 2009: Proc., International AAAI Conference on Weblogs and Social Media*, 2009.

[6] Christian Becker and Christian Bizer. Exploring the geospatial semantic web with DBpedia mobile. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(4):278–286, 2009.

[7] Dave Beckett and Brian McBride. RDF/XML syntax specification (revised). `http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210`, 2004.

[8] David Beckett, Tim Berners-Lee, and Eric Prud'hommeaux. Turtle – terse RDF triple language. `http://www.w3.org/TeamSubmission/turtle`, 2010.

[9] T. Berners-Lee, Y. Chen, L. Chilton, D. Connolly, R. Dhanaraj, J. Hollenbach, A. Lerer, and D. Sheets. Tabulator: Exploring and analyzing Linked Data on the semantic web. In *SWUI 2006: Proc., 3rd International Semantic Web User Interaction Workshop*, 2006.

[10] T Berners-Lee, J. Hollenbach, Kanghao Lu, J. Presbrey, E. Prud'ommeaux, and mc schraefel. Tabulator redux: Browsing and writing Linked Data. In *LDOW 2008: Proc., Linked Data on the Web Workshop at WWW'2008*, 2008.

[11] Tim Berners-Lee. Linked Data – design issues. `http://www.w3.org/DesignIssues/LinkedData`, 2006. (last updated: 2010).

[12] Tim Berners-Lee and Dan Connolly. Notation3 (N3): A readable RDF syntax. `http://www.w3.org/TeamSubmission/n3`, 2008.

[13] Enrico Bertini, Heidi Lam, and Adam Perer. Beliv'10: Beyond time and errors novel evaLuation methods for Information Visualization. In *CHI EA '10: CHI '10 extended abstracts, 28th international conference on Human factors in computing systems*, pages 4441–4444, 2010.

[14] C. Bizer. The emerging web of Linked Data. *IEEE Intelligent Systems*, 24(5):87–92, 2009.

[15] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked Data – the story so far. *International Journal on Semantic Web and Information Systems – Special issue on LInked DAta*, 5(3): 1–22, 2009.

[16] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. Dbpedia - a crystallization point for the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154–165, 2009.

[17] Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, and François Yergeau. Extensible markup language (XML) 1.0 (5th ed.). `http://www.w3.org/TR/xml`, 2008.

[18] José Manuel Cantera Fonseca. RDF next steps, Telefónica's position paper. In Wood et al. [78].

[19] Stuart K. Card, Jock D. Mackinlay, and Ben Shneiderman, editors. *Readings in information visualization: using vision to think*. Morgan Kaufmann Publishers, Inc., 1999.

[20] Sheelagh Carpendale. Evaluating information visualizations. In Kerren et al. [46], pages 19–45.

[21] Gong Cheng and Yuzhong Qu. Searching linked objects with Falcons: Approach, implementation and evaluation. *International Journal on Semantic Web and Information Systems*, 5 (3):49–70, 2009.

[22] Richard Cyganiak and Anja Jentzsch. Linking Open Data cloud diagram. `http://lod-cloud.net`, 2010.

[23] Erik Dahlström, Jon Ferraiolo, Fujisawa Jun, Anthony Grasso, Dean Jackson, Chris Lilley, Cameron McCormack, Doug Schepers, Jonathan Watt, and Patrick Dengler. Scalable vector graphics (SVG) 1.1 (2nd ed.). `http://www.w3.org/TR/SVG11`, 2010.

[24] Stephen Davies, Jesse Hatfield, Chris Donaher, and Jessica Zeitz. User interface design considerations for Linked Data authoring environments. In *LDOW 2010: Proc., LInked DAta on*

the Web Workshop at WWW'2010, 2010.

[25] Jean-Daniel Fekete, Jarke J. van Wijk, John T. Stasko, and Chris North. The value of information visualization. In Kerren et al. [46], pages 1–18.

[26] Flavius Frasincar, Alexandru Telea, and Geert-Jan Houben. Adapting graph visualization techniques for the visualization of RDF data. In Vladimir Geroimenko and Chaomei Chen, editors, *Visualizing the Semantic Web*, pages 154–171. Springer London, 2006.

[27] Hugh Glaser, Ian Millard, and Afraz Jaffri. RKBExplorer.com: A knowledge driven infrastructure for Linked Data providers. In *ESWC 2008: Proc., 5th European Semantic Web Conference*, pages 797–801, 2008.

[28] L. Grammel, M. Tory, and M. Storey. How information visualization novices construct visualizations. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):943–952, 2010.

[29] Wolfgang Halb, Yves Raimond, and Michael Hausenblas. Building Linked Data for both humans and machines. In *LDOW 2008: Proc., Linked Data on the Web Workshop at WWW'2008*, 2008.

[30] Wolfgang Halb, Alexander Stocker, Harald Mayer, Helmut Mülner, and Ilir Ademi. Towards a commercial adoption of linked open data for online content providers. In *I-SEMANTICS '10: Proc., 6th International Conference on Semantic Systems*, pages 16:1–16:8, 2010.

[31] Tuukka Hastrup, Richard Cyganiak, and Uldis Bojars. Browsing Linked Data with Fenfire. In *LDOW 2008: Proc., Linked Data on the Web Workshop at WWW'2008*, 2008.

[32] Marti A. Hearst and Daniela Rosner. Tag clouds: Data analysis tool or social signaller? In *HICSS '08: Proc., 41st Annual Hawaii International Conference on System Sciences*, pages 160–, 2008.

[33] T. Heath. How will we interact with the Web of Data? *IEEE Internet Computing*, 12(5):88–91, 2008.

[34] Tom Heath and Christian Bizer. *LInked DAta: Evolving the Web into a Global Data Space*. Morgan & Claypool, 2011.

[35] Tom Heath and Enrico Motta. Revyu: Linking reviews and ratings into the Web of Data. *Journal of Web Semantics: Special Issue on Science, Services and Agents on the World Wide Web*, 6(4):266–273, 2008.

[36] Jeffrey Heer, Frank van Ham, Sheelagh Carpendale, Chris Weaver, and Petra Isenberg. Creation and collaboration: Engaging new audiences for information visualization. In Kerren et al. [46], pages 92–133.

[37] Jeffrey Heer, Michael Bostock, and Vadim Ogievetsky. A tour through the visualization zoo. *Communications of the ACM*, 53(6):59–67, 2010.

[38] Philipp Heim, Steffen Lohmann, and Timo Stegemann. Interactive relationship discovery via the semantic web. In Aroyo et al. [3], pages 303–317.

[39] A. Hogan, A. Harth, A. Passant, S. Decker, and A. Polleres. Weaving the Pedantic Web. In *LDOW 2010: Proc., LInked DAta on the Web Workshop at WWW'2010*, 2010.

[40] David Huynh, Stefano Mazzocchi, and David R. Karger. Piggy Bank: Experience the semantic web inside your web browser. *Journal of Web Semantics*, 5(1):16–27, 2007.

[41] Alfred Inselberg. The plane with parallel coordinates. *The Visual Computer*, 1:69–91, 1985.

[42] M. Jern, T. Astrom, and S. Johansson. GeoAnalytics tools applied to large geospatial datasets. In *IV '08: Proc., 12th International Conference on Information Visualisation*, pages 362–372, 2008.

[43] David Karger and m.c. schraefel. The pathetic fallacy of RDF. In *SWUI 2006: Proc., 3rd International Semantic Web User Interaction Workshop*, 2006.

[44] Daniel Keim, Gennady Andrienko, Jean-Daniel Fekete, Carsten Görg, Jörn Kohlhammer, and Guy Melançon. Visual analytics: Definition, process, and challenges. In Kerren et al. [46], pages 154–175.

[45] Tanja Keller and Sigmar-Olaf Tergan. Visualizing knowledge and information: An introduction. In Sigmar-Olaf Tergan and Tanja Keller, editors, *Knowledge and Information Visualization*, pages 1–23. Springer, 2005.

[46] Andreas Kerren, John T. Stasko, Jean-Daniel Fekete, and Chris North, editors. *Information Visualization: Human-Centered Issues and Perspectives*. Springer-Verlag, 2008.

[47] Graham Klyne and Jeremy J. Carroll. Resource description framework (RDF): Concepts and abstract syntax. `http://www.w3.org/TR/rdf-concepts`, 2004.

[48] Georgi Kobilarov and Ian Dickinson. Humboldt: Exploring Linked Data. In *LDOW 2008: Proc., Linked Data on the Web Workshop at WWW'2008*, 2008.

[49] Georgi Kobilarov, Tom Scott, Yves Raimond, Silver Oliver, Chris Sizemore, Michael Smethurst, Christian Bizer, and Robert Lee. Media meets Semantic Web - how the BBC uses DBpedia and Linked Data to make connections. In Lora Aroyo et al., editors, *ESWC 2009: Proc., 6th European Semantic Web Conference*, pages 723–737, 2009.

[50] Jörg Koch and Thomas Franz. LENA - browsing RDF data more complex than FOAF. In Christian Bizer and Anupam Joshi, editors, *ISWC2008: Proc., 7th International Semantic Web Conference (Poster and Demonstration Session)*, 2008.

[51] Nuno Lopes, Axel Polleres, Alexandre Passant, Stefan Decker, Stefan Bischof, Diego Berrueta, Antonio Campos, Stéphane Corlosquet, Orri Erling, Jérôme Euzenat, Kingsley Idehen, Jacek Kopecký, Thomas Krennwallner, Davide Palmisano, Janne Saarela, and Michal Zaremba. RDF and XML: Towards a uniïňĄed query layer. In Wood et al. [78].

[52] P. Mutton and J. Golbeck. Visualization of semantic metadata and ontologies. In *Proc. Seventh International Conference on Information Visualization, 2003. IV 2003.*, pages 300–305, 2003.

[53] Tope Omitola, Christos L. Koumenides, Igor O. Popov, Yang Yang, Manuel Salvadores, Martin Szomszor, Tim Berners-Lee, Nicholas Gibbins, Wendy Hall, m. c. schraefel, and Nigel Shadbolt. Put in your postcode, out comes the data: A case study. In Aroyo et al. [3], pages 318–332.

[54] M.T. Pazienza, N. Scarpato, and A. Stellato. Semi-automatic generation of GUIs for RDF browsing. In *IV 2010: Proc., 14th International Conference on Information Visualisation*, pages 267–272, 2010.

[55] Steven Pemberton et al. XHTML 1.0 the extensible hypertext markup language (2nd ed.) – a reformulation of HTML 4 in XML 1.0. `http://www.w3.org/TR/xhtml1`, 2002.

[56] Emmanuel Pietriga. Semantic web data visualization with graph style sheets. In *SoftVis '06: Proc., 2006 ACM symposium on Software visualization*, pages 177–178, 2006.

[57] Emmanuel Pietriga, Christian Bizer, David R. Karger, and Ryan Lee. Fresnel: A browser-independent presentation vocabulary for RDF. In Isabel F. Cruz, Stefan Decker, Dean Allemang, Chris Preist, Daniel Schwabe, Peter Mika, Michael Uschold, and Lora Aroyo, editors, *Proc., 5th International Se-*

mantic Web Conference, pages 158–171, 2006.

[58] Z. Pousman, J.T. Stasko, and M. Mateas. Casual information visualization: Depictions of data in everyday life. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1145–1152, 2007.

[59] A Johannes Pretorius and Jarke J Van Wijk. What does the user want to see? what do the data want to be? *Information Visualization: Special issue on Human-Centered Information Visualization*, 8(3):153–166, Sep 2009.

[60] Eric Prud'hommeaux and Andy Seaborne. SPARQL query language for RDF. `http://www.w3.org/TR/rdf-sparql-query`, 2008.

[61] Yves Raimond, Tom Scott, Patrick Sinclair, Libby Miller, Stephen Betts, and Frances McNamara. Case study: Use of semantic web technologies on the BBC web sites. `http://www.w3.org/2001/sw/sweo/public/UseCases/BBC`, 2010.

[62] Lloyd Rutledge, Jacco van Ossenbruggen, and Lynda Hardman. Making RDF presentable: integrated global and local semantic web browsing. In *WWW '05: Proc., 14th International Conference on World Wide Web*, pages 199–206, 2005.

[63] J. Scholtz. Beyond usability: Evaluation aspects of visual analytic environments. In *IEEE VAST 2006: Proc., IEEE Symposium On Visual Analytics Science And Technology*, pages 145–150, 2006.

[64] m.c. schraefel, Max Wilson, Alistair Russell, and Daniel A. Smith. mspace: improving information access to multimedia domains with multimodal exploratory search. *Communications of the ACM*, 49(4):47–49, 2006.

[65] m.c. schraefel, Jennifer Golbeck, Duane Degler, Abraham Bernstein, and Lloyd Rutledge. Semantic web user interactions: exploring HCI challenges. In *CHI '08: Proc., 26th annual SIGCHI conference on Human factors in computing systems*, pages 3929–3932, 2008.

[66] Andy Seaborne. RDQL – a query language for RDF. `http://www.w3.org/Submission/RDQL`, 2004.

[67] François-Paul Servant. Linking enterprise data. In *LDOW 2008: Proc., Linked Data on the Web Workshop at WWW'2008*, 2008.

[68] Helen Sharp, Yvonne Rogers, and Jenny Preece. *Interaction Design: Beyond Human Computer Interaction*. John Wiley & Sons, 2nd edition, 2007.

[69] B. Shneiderman. The eyes have it: a task by data type taxonomy for information visualizations. In *Proc., IEEE Symposium on Visual Languages, 1996*, pages 336–343, 1996.

[70] Ben Shneiderman, Catherine Plaisant, Maxine Cohen, and Steven Jacobs. *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley Publishing Company, 5th edition, 2009.

[71] J.J. Thomas and K.A. Cook, editors. *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. IEEE CS Press, 2005.

[72] Edward Tufte. *Envisioning information*. Graphics Press, Cheshire, CT, USA, 1990.

[73] Edward R. Tufte. *Visual explanations: images and quantities, evidence and narrative*. Graphics Press, Cheshire, CT, USA, 1997.

[74] Edward R. Tufte. *The visual display of quantitative information*. Graphics Press, Cheshire, CT, USA, 2nd edition, 2001.

[75] Giovanni Tummarello, Richard Cyganiak, Michele Catasta, Szymon Danielczyk, Renaud Delbru, and Stefan Decker. Sig.ma: Live views on the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(4):355–364, 2010.

[76] Frank van Ham, Martin Wattenberg, and Fernanda B. Viégas. Mapping text with phrase nets. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1169–1176, 2009.

[77] Fernanda B. Viegas, Martin Wattenberg, and Jonathan Feinberg. Participatory visualization with wordle. *IEEE Transactions on Visualization and Computer Graphics*, 15:1137–1144, 2009.

[78] David Wood, Stefan Decker, and Ivan Herman, editors. *W3C Workshop – RDF Next Steps*, 2010.

[79] Ji Soo Yi, Youn-Ah Kang, John T. Stasko, and Julie A. Jacko. Understanding and characterizing insights: how do people gain insights using information visualization? In *BELIV '08: Proc., 2008 Workshop – BEyond time and errors: novel evaLuation methods for Information Visualization*, pages 1–6, 2008.