Hide the Stack: Toward Usable Linked Data

Aba-Sah Dadzie^{1,⋆}, Matthew Rowe², and Daniela Petrelli³

OAK Group, Dept. of Computer Science, The University of Sheffield, UK Knowledge Media Institute, The Open University, Milton Keynes, UK Art & Design Research Centre, Sheffield Hallam University, UK a.dadzie@dcs.shef.ac.uk, m.c.rowe@open.ac.uk, d.petrelli@shu.ac.uk

Abstract. The explosion in growth of the Web of Linked Data has provided, for the first time, a plethora of information in disparate locations, yet bound together by machine-readable, semantically typed relations. Utilisation of the Web of Data has been, until now, restricted to the members of the community, eating their own dogfood, so to speak. To the regular web user browsing Facebook and watching YouTube, this utility is yet to be realised. The primary factor inhibiting uptake is the usability of the Web of Data, where users are required to have prior knowledge of elements from the Semantic Web technology stack. Our solution to this problem is to hide the stack, allowing end users to browse the Web of Data, explore the information it contains, discover knowledge, and use Linked Data. We propose a template-based visualisation approach where information attributed to a given resource is rendered according to the rdf:type of the instance.

Keywords: Linked Data; Knowledge Visualisation; Information Visualisation; Usable Interfaces; Human-Computer Interaction.

1 Introduction

The Web of Linked Data now connects a wide range of previously disparate and isolated information sources, allowing complex, bespoke queries to be answered that were previously not possible or hard to derive answers for. To tech-savvy users, and in particular, researchers in the Linked Data (LD) community, consumption of LD is easy given their know-how writing SPARQL¹ queries or by applying a follow-your-nose principle to sniff out facts and connections between pieces of information. However, to the mainstream web user – who we define as the frequent user who browses web sites, chats with friends on, e.g., Facebook, but has no real knowledge of the intrinsic functionality of the Web they base their interaction on – there exists a gap between exploiting the Web of Data (WoD) to answer queries and the technological know-how to do so. The regular web user does not (and should not need to) know SPARQL, nor RDF² (Resource Description Framework), what an ontology or Linked Data is, nor any other element which the Semantic Web (SW) encompasses. This is a problem.

^{*} To whom correspondence should be addressed.

¹ SPARQL query language for RDF: http://www.w3.org/TR/rdf-sparql-query

² Resource Description Framework (RDF): http://www.w3.org/RDF

G. Antoniou et al. (Eds.): ESWC 2011, Part I, LNCS 6643, pp. 93-107, 2011.

[©] Springer-Verlag Berlin Heidelberg 2011

To reduce this gap, we propose to make LD usable, allowing end users to embrace the power of the WoD and browse and discover connections between pieces of information and facts as they would on the World Wide Web (WWW – the Readable Web). In bridging the gap we will put a powerful database at the disposal of end users, one which is community maintained and provides answers to unique questions. In essence we propose to *hide the stack* from end users, allowing them to use RDF, SPARQL, ontologies, and all those other elements that make up the *layer cake* [4], without being aware they are doing so. We define this thesis as the *invisible stack* that is expressed by the research question:

How can we make Linked Data usable to real, end users?

From exploring this question we propose a template-based approach to visualising linked data, that, starting from the underlying data structure associated with a given resource, presents information in a legible and coherent form. The rdf:type of a resource provides the primary indicator as to which template(s) to employ, by dereferencing the URI, returning the instance description, and tailoring this information into a legible form, that caters to the user's context, i.e., current task and end goal. Our approach reduces the information load on the user – an endemic problem concerning LD visualisation, given the scale of the WoD – supporting easier interpretation and a coherent view of data.

We have structured this paper as follows: section 2 describes the challenges imposed on the visualisation of LD, based on the current state of the WoD. Section 3 presents related work and how such challenges have been addressed thus far. Section 4 contains our central contribution: our approach to visualising linked data via templates. We define scenarios which motivate and provide references for clarity in explaining our approach. The section concludes by discussing a formative evaluation. Section 6 concludes the paper with the findings and lessons drawn from our work, and plans for future work.

2 Toward Usable Linked Data

Linked Data in its raw form consists of (often very large) sets of RDF statements. RDF was designed to support reading and interpretation of data on the Web by machines. This focus often results in data that is not always easily interpreted by humans, especially outside the SW community. Take, e.g., Fig. 1B, which describes a publication in the Data.dcs³ linked dataset; the URI (Uniform Resource Indicator) http://data.dcs.shef.ac.uk/paper/4169 that references it is fairly cryptic – the only information directly derived from it is that it is a paper belonging to the institution represented by the URI http://data.dcs.shef.ac.uk (henceforth abbreviated as 'data.dcs:'). The complete RDF description, based on its bibtex⁴ citation, is however easily interpreted by humans; this includes its bib:title, publication year (bib:hasYear), the book/collection that contains it (bib:hasBookTitle), and its authors (foaf:makers).

 $^{^3}$ The $Data.dcs^{\alpha}$ LD dataset may be browsed from: http://data.dcs.shef.ac.uk

⁴ See the BibTeX resource pages at: http://www.bibtex.org

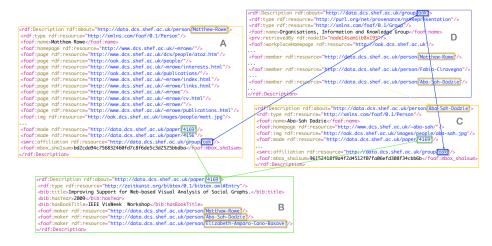


Fig. 1. Extracts from *Data.dcs*, highlighting links between selected resources. A and C describe two foaf:Person resources; D their swrc:affiliation; B a bib:Entry resource (a paper), http://data.dcs.shef.ac.uk/paper/4169, common to both.

Figs. 1A and C describe two foaf:Person resources (with orange borders): data.dcs:person/Matthew-Rowe and data.dcs:person/Aba-Sah-Dadzie, respectively, indirectly linked through co-authorship of the paper in B (bib:Entry, green). D describes a second indirect relation, using a broken link – their common swrc:affiliation, data.dcs:group/oak (blue).

Fig. 1 effectively communicates the inter-relationships because small extracts from Data.dcs have been collected in close proximity and specific regions highlighted and colour coded. This, supplemented with bi-directional arrows between resources, results in simple visual encoding that allows the user to gain very quickly an overview and understanding of the relationships within the data. Overlaying the visual encoding on the text allows the user to delve further into the data to retrieve more detail for regions of interest (ROIs), e.g., browsing from the resource data.dcs:person/Matthew-Rowe to obtain a human-readable description of the object they made: data.dcs:paper/4169.

As data set size increases, however, human ability to identify such relationships and retain them in memory decreases significantly. This poses a challenge for the very large amounts of data generated in today's information-rich society, with datasets containing up to millions of entities. Data.dcs by comparison is tiny, containing only ~ 8000 statements. However even this poses significant cognitive challenges for manual analysis, due to the difficulty obtaining a good mental overview of large amounts of complex, highly interlinked data [8,9,10].

Further difficulties arise when exploring a new environment about which a user has little information, beyond that it contains answers to the questions they wish to ask. While an SW expert would be comfortable with or expect to start browsing LD from a specific URI, the mainstream end user will not have the domain or technical knowledge to do so. In keeping with typical Web usage,

a user such as those in our scenarios (see section 4.1) may have obtained a start address from a flier or via a natural language query in a web search engine. From this point they will start to explore the WoD; whether this is the *Readable* or the *Semantic* Web should be transparent to the mainstream user. End users should be given usable tools to explore the WoD, linking to relevant LD or other data in the wider Web, and access to simple methods for exporting the data to alternative readable formats. For the tech-savvy user it is also useful to allow extraction of the underlying RDF data using formal syntax such as SPARQL.

For such interaction to occur, we have identified key usability challenges which currently exist in using LD:

2.1 Challenges to Linked Data Use

Exploration starting point: where to start; existing LD browsers assume the end user will start browsing from a specific, valid URI. How can a visualisation starting point be presented to users in such a way that it is meaningful?

Combating information overload: presenting end users with all the properties of a given resource, along with the relations through which the resource is linked to other entities, leads to information saturation and a dense information space. How can we present this information in a more legible form?

Returning something useful: RDF is the staple recipe for resource descriptions, returning information using this knowledge representation format inhibits comprehension. How can RDF, and the information contained within instance descriptions, be represented in a more legible, manageable form?

Enabling interaction: end users are familiar with the makeup of the Web and its browsable nature. Is it possible to replicate such familiarity which users experience when browsing the WWW on the WoD?

3 Related Work

The SW community has to date largely focused on the use of text-based representations of linked data, often (explicitly or transparently) through the use of SPARQL endpoints. This is due to two main reasons: (1) the infancy of the LD initiative; (2) the focus on prototypes to serve the needs of the SW community and other specialised domains, to generate and analyse LD and related SW data. The most widely referenced LD browsers include Sig.ma [20], Marbles [3] and URI Burner⁵. Simile⁶ provides access to a set of tools and APIs (Application Programming Interfaces) for presenting and interacting with RDF data.

Haystack [15] was one of the first SW browsers developed, to lower the barrier to consumption of SW resources. It uses stylesheets and views/lenses defined in RDF to aggregate distributed information and customise its presentation for specific users and tasks. Corlosquet et al., [6] describe the extension of Drupal⁷

⁵ URI Burner: http://linkeddata.uriburner.com

⁶ Simile: http://simile.mit.edu

⁷ Drupal Content Management System: http://drupal.org

to create enriched web sites, whose models are defined using standard ontologies, in order to embed RDF into the underlying content. Their approach consequently enables the retrieval also of relevant LD on the fly, via SPARQL endpoints.

Textual presentation of RDF data, while familiar and useful to SW expert users, results in high cognitive load for non-technical and non-domain experts, and even for experts, as data amount and interconnectivity increase. Visualisation is acknowledged to enhance knowledge discovery and analytical ability while lowering cognitive load, by harnessing powerful human perception [10,11,17] to enable intuitive construction of an understanding of the structure of large amounts of complex, interacting data. Visualisation affords a number of everyday metaphors, by encoding data attributes into, e.g., graphs, maps and timelines, providing more understandable user interfaces (UIs) over machine-friendly RDF. The LD community is, not surprisingly, examining visual solutions for both technical and mainstream use. These often visualise the RDF graph structure, e.g., [7,9,13,18]; such graphs are very useful to technical experts, whose requirements include inspection to identify errors, retrieve selected data and relations, and develop specialised applications. A well-known RDF visualisation tool is IsaViz [13], which overlays RDF graphs with stylesheets based on Fresnel lenses [14].

A small number of LD visualisation browsers include *RelFinder* [7], which automates link discovery between user-specified resources. *Tabulator* [5] interprets LD by mapping to standard ontologies such as FOAF⁸ (Friend of a Friend), with output to a nested hierarchy and visualisations including map and calendar views. *RDFScape* [18] is a *Cytoscape*⁹ plug-in to enhance analysis in Bioinformatics, visualising the results of ontology-based inference with node-link graphs.

However, there is a dearth in applications targeted at mainstream, non-technical use. Hirsch et al., [9] help to fill this gap, by visualising the (semantic) knowledge content in $Freebase^{10}$ and $Wikipedia^{11}$ using node-link graphs. They use icons, colour and relative node and edge size to encode data attributes, and draw (labelled) links between clusters of related semantic information. LESS [2] supports the creation of (shareable) web-based templates to aggregate and display LD to mainstream users through the familiar presentation methods of the readable Web. $DBPedia\ Mobile$ [3] exploits a geographical metaphor to link and publish information about resources in the user's vicinity. It further lowers the barrier to interaction with LD by enabling the publication of new information about resources in users' physical location to the LD cloud.

Our review of the state of the art, with respect to the challenges outlined in section 2.1, highlights the work to be done to make LD usable to non-tech savvy end users. In particular we note the need for solutions which allow more flexible, open-ended knowledge discovery. RelFinder and DBPedia Mobile come closest to fulfilling this, by revealing, using different mechanisms, relations between distinct entities. Additionally we note that low user familiarity with linked

⁸ FOAF Ontology Specification: http://xmlns.com/foaf/spec

⁹ Cytoscape: http://www.cytoscape.org

¹⁰ Freebase: http://www.freebase.com

Wikipedia: http://en.wikipedia.org

datasets – neither the intricacies of the data, nor its entire subject scope – hinders the formulation of initial questions. In such cases a viable means of exploration is required, much as Web browsing functions, to allow intuitive information discovery. In turn, a challenge of such discovery mechanisms is information overload.

Visualisation is essential in providing an entry point into the WoD and combating the cognitive load associated with the use of these large, distributed, highly inter-connected data sets. However visualisation has its own limits. In section 4 we explore the synergies between higher level data overviews and detailed, interactive analysis of ROIs. What constitutes a useful detail view however varies depending on the end user and their task; while we focus on the design of UIs that support mainstream end users we are careful not to ignore the expert user. This paper aims also to address this last challenge – graphical overviews, for instance, allow users to obtain a high level understanding of data structure. The mainstream end user can use these to identify a starting point for their exploration, while the expert will quickly recognise valid patterns and clusters in addition to anomalies in the data structure.

4 Template-Based Visualisation of Linked Data

Differences in users, their tasks and environments mean no one solution can claim to exhaustively meet all requirements for using LD. However, template-based approaches hold significant potential for helping to bridge these challenges. Templates are a useful, flexible tool that may be used to define how to format RDF data into a human-readable representation [2,6,14,15,16], and synthesise related but distributed, heterogeneous information [3,9,18,20], improving management of its knowledge content. In this section we discuss our design rationale for a template-based visualisation approach to presenting linked data, a solution that tackles the challenges identified in section 2.1.

To provide sufficient context to illustrate our work, we detail two scenarios:

4.1 Scenarios of Use

- a. A primary school teacher in Sheffield is preparing a technology project for year 6 pupils that examines the future of the Web. To help her pick a topic that will engage her students she wishes to speak to researchers in the local university exploring leading edge Web technology. She goes to the University of Sheffield web site and navigates to the Department of Computer Science. She finds a link to the research areas...
- b. Anne has just completed her 'A' levels and received a grant to study Computer Science (CS) at the University of Sheffield. Her parents know that she wishes to work as a lecturer after she graduates. They want to reassure themselves that she will be exposed to a wide range of topics and be able to interact with academics at the forefront of their fields. Therefore they go the University website and navigate to the Department of Computer Science...

In the remainder of the paper we will complete both stories, illustrating, with *Data.dcs*, how our solution helps these actors retrieve the information they seek.

4.2 Design Rationale and Implementation

Our solution to visualising linked data is to hide the stack; by this we refer to utilising SW technologies inherent in LD, but with these elements transparent to the user. In short, this bypasses the complexity associated with RDF and SPARQL by rendering information returned using such standards in a human-legible form. The proposed solution makes use of custom templates loaded based on the rdf:type of the resource being viewed by the user. Two views are provided: a graph view that draws the relations between resources, displaying information defined using object properties; and a co-ordinated detail view that presents attributes and information defined using datatype properties. Our solution allows the collapsing of detail into compound nodes, revealing detail only for the focus. More advanced options are also available to the tech-savvy user, for advanced querying that links directly to the wider WoD using public SPARQL endpoints.

We first describe our design, and how we translated this into the presentation of information using templates. We then summarise the results of a focus group study designed to collect user opinions about the capability enabled for exploratory knowledge discovery and directed search.

The Templates

The first step in our template design is to identify which resource types in a given data set may be of interest to the end user. At the same time a solution is required that does not assume more than cursory knowledge of data structure or content. Well specified LD should, as far as is possible, re-use and/or extend standard ontologies and vocabularies when describing data content. Core concepts from such ontologies and vocabularies are therefore an ideal first point from which to create reusable templates, adaptable to user context and tasks.

We illustrate this for metadata that describes people and information about people, starting from the organisations they work for. Examples of widely applicable ontologies, from which we select relevant classes and properties or relations to define templates, include: **FOAF** to describe people and their relations with other people and organisations; **SWRC** (Semantic Web for Research Communities¹²), relevant to the use cases we discuss. Further, SWRC is easily generalised to other organisational structures; **BibTeX** to describe publications; **PRV**, the Provenance Vocabulary¹³, to support verification of data content and quality.

These examples both provide a solution and raise another issue — Organization and Person, for instance, are defined in both FOAF and SWRC. For the initial versions of the templates we chose to model concepts using the most widely referenced ontology or vocabulary, to increase reusability. So our templates default to, e.g., FOAF for Organization and Person, but use both member from FOAF and affiliation from SWRC, to model directed relationships between an Organization and a Person, respectively. Documents (of which Publications are a subset) are modelled using BibTFX rather than FOAF or SWRC, with

¹² SWRC Ontology Specification: http://ontoware.org/swrc

¹³ PRV Core Ontology Specification: http://purl.org/net/provenance/ns#

the relation between a Person and a bibtex:Entry modelled using the FOAF property maker. In order to apply the templates also to ontologies that redefine commonly referenced concepts we plan to include selectors that allow cascading to cater for redundancy, in addition to equivalent classes across ontologies, and subclasses defined in a single ontology or by extension in a new ontology.

To support extensibility and reusability we define a template for each key resource first using Fresnel lens SPARQL selectors (see top, Fig.5). We then translate these to a set of presentation formats in a Java prototype, using the design ideas expressed in Fig. 2. Three templates that provide customised views over RDF data are illustrated in this paper, using the *Data.dcs* linked dataset, for the resource types:

Group (Organisation): <http://xmlns.com/foaf/0.1/Group>

Person: http://xmlns.com/foaf/0.1/Person

Publication: http://zeitkunst.org/bibtex/0.1/bibtex.owl#Entry.

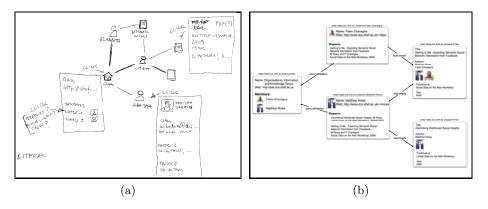


Fig. 2. A design sketch (2a) illustrates the use of icons to encode graph nodes, based on rdf:type. Links show primary relationships, e.g. foaf:Person foaf:made bib:Entry. Interaction with the graph, e.g. onNodeClick, reveals a group detail template (left); a Person (lower, right); a Publication (top, right). 2b illustrates potential implementation as a sub-graph of nodes expanded to show detail, with directed, labelled edges.

Returning to the scenarios, our design aims to support especially the non-expert user (outside the SW and CS domains). This type of user will typically have, at best, a broad idea of where to find the information they seek. They are most likely to exhibit exploratory information seeking behaviour, moving on to directed search once they have a more complete understanding of what knowledge is available. The scenarios have each actor at the start page of the CS department, where they are presented with: traditional, text-based browsing or visualisation-based browsing of the departmental structure. They choose the visualisation over reading the research pages on the web site, because the site explains that the latter displays the relationships between groups and researchers.

From an implementation point of view, that the visualisation is based on Data.dcs is (and should be) transparent to the user. We cannot assume that our target end user will have a specific URI to browse from (the root and initial focus of their visual graph). In this case two options are available: (1) presenting the user with a list of potential starting points, e.g., for Data.dcs, research groups or their heads – this is however only feasible for a restricted data set, which is seldom the case for LD; (2) random selection, restricted to key resource types. We currently use the second option, rooting the (directed) graph with the first subject read that matches one of the pre-specified resource types of interest – in the Data.dcs example this would be a group/affiliation, a person or a publication.

The Graph View

Simply drawing the graph showing the relations within *Data.dcs* results in a representation such as that in Fig. 3a. To reduce the potential for cognitive overload the graph displays only the first two levels from the root at start-up, leaving the user to unfold the graph, up to its leaves. This however quickly results in a dense *web* of nodes, a problem common to node-link graphs [8]. This is seen even with reduced occlusion in Fig. 3a, by abbreviating resource URIs, and with only part of the graph drawn, illustrating expansion from the third to the fourth (out of eleven) levels. Presenting this to users, whether non-expert or expert, only highlights the density and complexity of the information they wish to explore; a more usable solution is necessary.

We resolve this issue first by filtering the dataset, based on key (RDF) resource types, identified through the use of standard templates, similar to the approach in Fresnel [14]: in this case, (research) group (Organisation), Person and paper (Publication). The $\sim\!8000$ statements in Data.dcs generate $\sim\!3000$ distinct graph nodes; the filters reduce the node count to $\sim\!300$. The detail for each instance of these resources is collapsed into what we shall refer to as a compound node. The resulting graph, in Fig. 3b, provides a high level overview, with significant reduction in occlusion, that displays key information in the data.

However, this solution presents yet another issue: the filters hide what is useful, detailed information from the user. As discussed also in [8,17] there is the need to support the examination of detail in selected ROIs. We resolve this by maintaining the visual graph as the centre of the user's exploration activity, and couple it with a view that formats for human consumption the detail for each compound node (see Figs. 4 and 5). This allows the user to maintain the context of surrounding information even while exploring ROIs in the co-ordinated detail window (which we describe in the following sub-section).

Colour coding and node size are the two main visual encoding methods for key properties in the graph. *Organisation* nodes and outlinks have a blue border, and *Person* and *Publication*, orange and green, respectively. The focus node has a red fill, and the borders of its immediate neighbours change to red *onMouseHover*. Keyword query matches are filled in pink. Node size may be weighted by the number of outlinks; this is especially useful when child nodes are folded into a parent, or hidden using a filter. In, e.g., Fig 4, group members with relatively

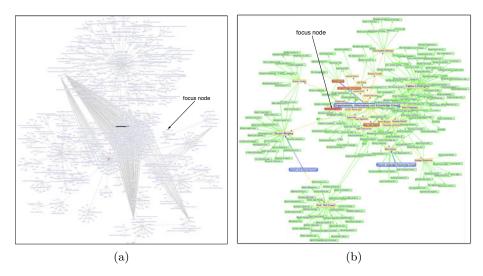


Fig. 3. 3a shows the layout for the first three levels of the complete graph for *Data.dcs*, as it is being expanded to the next level. 3b shows the equivalent graph for the first four levels that collapses detail into compound nodes using a custom filter.

high publication count stand out. The smallest group node has the smallest size (comparing height and node font size; width is determined by label length).

The actors in our scenarios both choose to explore by research group. Anne's parents are simply browsing to get a good feel for the variety and depth of the research carried out. The schoolteacher has a more complex goal, to identify which group is most likely to be able to provide her with information for her technology project. We will concentrate for now on the latter. Moving from Fig. 3b she elects to display only *Organisation* and *Person* nodes (filters – top, right, Fig 4), to remove the clutter of the large number of *Publications*, as these are currently of secondary interest. She then expands the filtered graph to display all levels, shown in the larger pane in Fig 4. The overall structure of the department is easily discerned. Distinct clusters differentiate each group, while links between them due to researchers in multiple groups span the space between clusters.

A quick scan of group names returns: (1) Computational Systems Biology (CompBio), (2) Machine Learning (ML), (3) Natural Language Processing (NLP), (4) Organisations, Information and Knowledge (OAK), (5) Speech and Hearing (SpandH) and (6) Verification and Testing (VT). The teacher eliminates all but one: OAK. Graph visualisation is able to give a high level overview of such large, interlinked data sets. However, the graphs are not as effective for detailed knowledge exploration. We describe next the rationale behind the template design used to provide human-readable detail, which the teacher will use to delve into the group's research, to determine if it is relevant to her project.

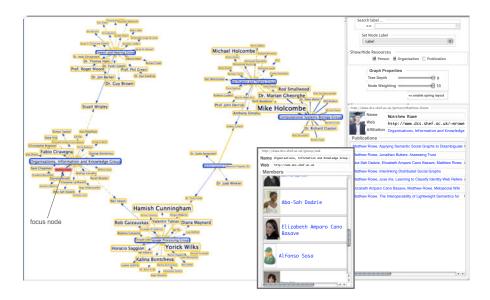


Fig. 4. Filtering out publications highlights links across groups in the departmental structure. The focus node, http://data.dcs.shef.ac.uk/person/Matthew-Rowe, is highlighted in red, and its detail shown in the template window on the right. The group template for this Person is populated and superimposed on the graph.

The Detail View

The teacher browses the information about the OAK Group by clicking on nodes for members (in the graph). This updates the coupled detail template window, allowing her to examine each Person resource in more detail. Alternatively, she could select the OAK Research Group node, to populate and display the Organisation detail template (overlay, Fig. 4), and switch to browse the detail for its members via the thumbnails linking to each Person template.

The member "Matthew Rowe" lists a set of publications that may be relevant (right pane, Fig. 4); the teacher selects one, "Improving Support for Webbased Visual Analysis of Social Graphs", to examine in greater detail – this is the human-readable label for the URI data.dcs:paper/4169. Fig. 5 shows the SPARQL query template (based on the Fresnel template for this resource) used to retrieve the full view for a publication, and the result of applying this template to data.dcs:paper/4169. This is the same node highlighted in Fig. 3; the difference in ability to interpret what the node represents can be clearly seen.

The teacher reduces the depth of the graph to two levels, removes the *Publication* filter, and zooms in to the detail for the three co-authors (bottom, Fig. 5). This allows her to scan (in the graph) the titles of other publications written by the authors, while maintaining the focus on the publication of interest in the detail template window (right). Armed with this information she goes to the web pages of each of the co-authors, via the **Person** detail view. She will also browse the **group**'s web pages, to extract more information about their projects.

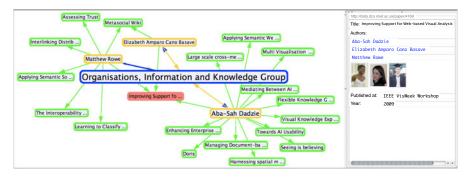


Fig. 5. The detail view for the Publication http://data.dcs.shef.ac.uk/paper/4169 is shown on the right, while the graph in the centre is zoomed in to show other (Publication) links from the Person resources of interest, its co-authors (or makers)

5 Formative Evaluation

To evaluate our approach a focus group study was carried out to: (1) review the requirements for applications for consuming LD; (2) determine if the coordinated views allow both mainstream users and technical and domain experts to obtain a good understanding of data content and structure; (3) identify where the design required revision prior to formal, summative, usability evaluation.

As part of a conference tutorial¹⁴ to study the importance of Human-Computer Interaction (HCI) and user-centred interface design when building end user tools on SW technology, 14 participants from the SW community and related research areas gave feedback on the use of the prototype for information exploration and retrieval tasks. The participants, who would be considered to be tech-savvy, assumed the role of expert reviewers (see [12,19]) inspecting the interface from the point of view of the target end user. At this stage in the design we consider, in addition to the mainstream user, the technical user building SW tools. Such users need support also to identify: (1) incomplete data; (2) errors, e.g., redundancy and incorrect linking during automatic LD generation; (3) key properties of the data and optimal ways for presenting these and their inter-relationships.

The participants were given an overview of the prototype and its design rationale, and challenges to LD exploration and use. They then carried out a practice

 $^{^{14}}$ 'Essential HCI for the Semantic Web' @ ESWC 2010: $\tt http://www.eswc2010.org$

task on a small data set based on a television series, to extract structural information. This was followed by a more complex exploration task, to retrieve information from *Data.dcs* on, among others, collaboration across groups, and researchers with multiple affiliations and relatively high publication count.

In focus groups of up to 4, the participants then reviewed the prototype as part of a participatory design activity. They assessed how well the UI supported user requirements, based on three main criteria: (1) effectiveness, i.e., how successful they were in discovering the information required; (2) efficiency, i.e., time to complete tasks; (3) user satisfaction, guided by a questionnaire.

5.1 Results

Because the evaluation was formative the focus was on collecting qualitative information to validate the requirements that fed into the UI design, and determine how effectively this had been translated to supporting LD consumption. Information was obtained by observing the participants, supplemented by a debrief in which the outcomes of the focus group study were discussed. This phase generated a number of post-it notes that were clustered and analysed.

The participants found, overall, the graphs to be quite expressive and effective in giving a sense of the data distribution. However those not accustomed to graph manipulation had difficulty controlling the display, due first to the large graph size, and also the (spring) layout algorithm that alters the layout to re-centre on focus change. Comments such as: "eventually you got a big picture of the data" and "I liked the direct manipulation but the graph should stay put [when I click]", show both frustration and appreciation. Another key feature recognised was the effectiveness in displaying detail for key resources: "in a neat and concise way".

The prototype was seen to have potential for exploring and debugging LD. The capability for keyword search was much appreciated. This was reinforced by requests for more selective filtering, both naïve and expert (e.g., via SPARQL). Though the latter could be motivated by the participants' technical expertise, it is more likely to indicate the need for tools for both (usable) browsing and search to be integrated into the same interface. Suggestions for improving interaction with the knowledge content included explicit highlighting of search paths in the graph; browsing through previous actions (history); additional options for navigation, e.g., porting between disconnected sub-graphs.

6 Conclusions

We have presented a solution to making LD usable by mainstream end users. This seeks to hide the stack from such users, enabling use of the rich network of information on offer, but without requiring knowledge of elements in the SW technology stack to process the information. For instance, the templates in our approach are, in essence, SPARQL queries, the bound variables within which are tied to regions in the information presentation view. The user need not know that SPARQL is utilised, nor that the resource description is returned as RDF.

We highlighted four key challenges for interacting with LD that we address as follows: the selection of an *exploration starting point* can be achieved via lists of potential starting points, allowing the user to choose the resource to focus on; or by randomly selecting a resource from which to start browsing.

For combating information overload we utilise two types of views, based on templates: a graph and a detail view. The former provides an overview of the network structure surrounding a resource of interest. Our default displays resources up to two steps (levels in the graph) from the focus on the WoD. This provides a clear picture that describes the relations between external resources and possible paths and transitions through the space, while preventing information overload. This is consistent with recent work in [7], where relations are revealed between DBPedia concepts in a graph. Our second view displays the properties of the focus resource, similar to work by [2], by rendering the RDF response from dereferencing the URI into a legible form. By marrying the two views we are able to separate out the logical components of relations and properties, where we use the latter to show datatype properties of the resource, and the former to show the context and role of the resource within the wider WoD.

The use of templates has also addressed the challenge of returning something useful, by providing information in a format that end users are able to understand and interpret. Colour coding resources in the graph view based on rdf:type, for example, enables the user to observe (rather than having to read and interpret) the connections and relations between different types of instances. We address the final challenge of enabling interaction by mimicking the browsable nature of the WWW via clickable regions, which shifts the focus to the selected resource.

We plan to carry out further usability evaluation with a larger sample and range of end users, using specific problem solving tasks to test the utility and usability of our approach. We believe that the specification of such tasks will provide the community with benchmark experiments for validating the effectiveness of methods for visualising LD. Our template-based visualisation approach utilises SPARQL queries to retrieve information from a resource's instance description and present this in a legible manner. We have detailed three static templates which demonstrate this functionality; our future work will enable the construction of templates by end users, in essence creating SPARQL queries in an implicit fashion. Providing end users with more control will also be explored, to allow for the restriction of selected property values when exploring the WoD, and filtering through only content relevant to their current activity. Finally, we plan to incorporate existing work in the field of ontology mapping, to enable templates to be loaded for instances of classes that are defined as being equivalent to others for which templates exist, e.g., loading an existing template for an instance of another class that is defined to be equivalent to foaf: Person.

Acknowledgements. A.-S. Dadzie is funded by the European Commission (EC) 7th Framework Programme (FP) projects *SmartProducts* and *WeKnowIt* (grants 231204, 215453). M. Rowe is funded by *WeGov* (EC FP7 248512). A.-S. Dadzie and D. Petrelli were previously funded by *X-Media* (EC FP6 026978).

References

- 1. Aroyo, L., et al. (eds.): ESWC 2010. LNCS, vol. 6089. Springer, Heidelberg (2010)
- 2. Auer, S., Doehring, R., Dietzold, S.: LESS template-based syndication and presentation of linked data. In: Aroyo, L., et al. (eds.) [1], pp. 211–224
- 3. Becker, C., Bizer, C.: Exploring the geospatial semantic web with DBpedia Mobile. Journal of Web Semantics 7(4), 278–286 (2009)
- 4. Berners-Lee, T.: Semantic web XML 2000 (2000), http://www.w3.org/2000/Talks/1206-xml2k-tbl
- Berners-Lee, T., Hollenbach, J., Lu, K., Presbrey, J., Prud'ommeaux, E., Schraefel, mc.: Tabulator redux: Browsing and writing linked data. In: Linked Data on the Web Workshop at WWW '08 (2008)
- Corlosquet, S., Delbru, R., Clark, T., Polleres, A., Decker, S.: Produce and consume linked data with Drupal! In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 763–778. Springer, Heidelberg (2009)
- 7. Heim, P., Lohmann, S., Stegemann, T.: Interactive relationship discovery via the semantic web. In: Aroyo, L., et al. (eds.) [1], pp. 303–317
- 8. Herman, I., Melançon, G., Marshall, M.: Graph visualization and navigation in information visualization: A survey. IEEE Transactions on Visualization and Computer Graphics 6(1), 24–43 (2000)
- 9. Hirsch, C., Hosking, J., Grundy, J.: Interactive visualization tools for exploring the semantic graph of large knowledge spaces. In: Workshop on Visual Interfaces to the Social and the Semantic Web at IUI 2009 (2009)
- Keller, T., Tergan, S.-O.: Visualizing knowledge and information: An introduction. In: Tergan, S.-O., Keller, T. (eds.) Knowledge and Information Visualization: Searching for Synergies, pp. 1–23 (2005)
- 11. Kerren, A., Stasko, J.T., Fekete, J.-D., North, C. (eds.): Information Visualization: Human-Centered Issues and Perspectives. Springer, Heidelberg (2008)
- 12. North, C.: Toward measuring visualization insight. IEEE Computer Graphics and Applications 26(3), 6–9 (2006)
- 13. Pietriga, E.: Semantic web data visualization with graph style sheets. In: SoftVis '06: Proc. 2006 ACM Symposium on Software Visualization, pp. 177–178 (2006)
- Pietriga, E., Bizer, C., Karger, D.R., Lee, R.: Fresnel: A browser-independent presentation vocabulary for RDF. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 158–171. Springer, Heidelberg (2006)
- 15. Quan, D.A., Karger, R.: How to make a semantic web browser. In: WWW '04: Proc. 13th International Conference on World Wide Web, pp. 255–265 (2004)
- Rutledge, L., van Ossenbruggen, J., Hardman, L.: Making RDF presentable: integrated global and local semantic web browsing. In: WWW '05: Proc. 14th International Conference on World Wide Web, pp. 199–206 (2005)
- 17. Shneiderman, B.: The eyes have it: a task by data type taxonomy for information visualizations. In: Proc. IEEE Symposium on Visual Languages, pp. 336–343 (1996)
- Splendiani, A.: RDFScape: Semantic web meets systems biology. BMC Bioinformatics 9(suppl. 4), S6 (2008)
- 19. Tory, M., Moller, T.: Evaluating visualizations: do expert reviews work? IEEE Computer Graphics and Applications 25(5), 8–11 (2005)
- Tummarello, G., Cyganiak, R., Catasta, M., Danielczyk, S., Delbru, R., Decker, S.: Sig.ma: Live views on the web of data. Journal of Web Semantics 8(4), 355–364 (2010)