Position statement: in favour of (more) intelligence in the semantic UI

Ian Dickinson[1]

Hewlett Packard Laboratories, Bristol, UK
i.j.dickinson@gmail.com

Abstract. In this position statement I advocate for a particular approach in creating interfaces for semantic web user interfaces: designing and building intelligent user interfaces with increased automation of user tasks. My suggestion is that this goal is facilitated by the characteristics of the underlying semantic information model, which will provide raw material for improved models of user profiles and goals.

1 Introduction

This is a position paper based on in-progress work at HP Labs. Since the development of the research prototypes is ongoing, rather than detailed report of the design choices and algorithms, I will discuss some of the reasons why we have picked these particular approaches to the problem of providing an effective user experience to a semantic web information sources.

2 Motivation: what *is* a semantic web user interface?

The premise underlying a workshop on semantic web user interfaces must be that there is something particular about creating user experiences for semantic-web based datasets. That creating a user interface for a semantic application is not simply a matter of applying more-or-less well understood UI techniques from other domains. Where might such differences arise from? I suggest the possibilities include:

- semantic web information sources are loosely structured and extensible, so interfaces which exploit them have to be dynamic and able to cope with content not specifically anticipated by the interface designers. The ability of the semantic web to interlink heterogeneous datasets may mean that an interface has to cope with presenting, and allowing users to interact with, classes of resource types that were not foreseen;
- many kinds of problems for which semantic-web or linked-open-data (LOD) approaches are the right answer are inherently exploratory in nature;
- the semantic meaning attached to the contents of a semantic-web or LOD information store can be exploited both within the task domain, to improve users' queries, and within the user-interaction process itself, to change the way that users formulate and refine their queries.

To be interesting as a semantic web user interface, I argue the characteristics of the underlying semantic information store should be manifest through the interface, either through the form of the data, or in the user's task and interaction model.

2.1 Some semantic web user interaction challenges

Semi-structured data formalisms, such as RDF, are highly extensible. This is indeed one of their key value propositions. Connecting two LOD datasets together can require as little as the acquisition of metadata which creates equivalences or other links between concepts and individuals in the datasets to be linked. For example, MusicBrainz[1] `MusicArtist` and DbPedia [1] `MusicalArtist` correspond and might be denoted `skos:exactMatch` using SKOS metadata[2]. Making such a link opens up a greater range of resources to address a user's query, potentially giving richer, more meaningful or more complete results. However, it also brings challenges. Adding additional datasets may require the means to present unanticipated resource types meaningfully within the UI, consistently with the extant affordances on display. As a simple example, consider a dataset in which resources are given readable labels with `rdfs:label`, while another uses `skos:prefLabel`, etc. Dynamically extending the range of linked data sources requires addressing users' concerns over the provenance and trustability of results. Some approaches to presenting data not anticipated by the designer are available, for example Fresnel[3] and Tabulator[4]. However, it remains in my view an open research question how we select, for example, a particular Fresnel lens given the current user context (goals, screen layout, active resultset etc).

What kinds of tasks do end users want to perform over semantic-web or LOD datasets? Some analyses are available (e.g. by Heath et al [5] and Battle [6]), but in my view this remains an open question in the semantic web research community. In particular, we lack insight into the ways the interlinked, semantically meaningful characteristics of the datasets *are themselves the key enablers* of desirable user behaviours. Put another way, from a user-centred design perspective, how do users' goals and tasks change when the users are interacting with semantic web based information sources as opposed to traditional information sources (whether or not the users themselves are aware of the difference)?

An interesting class of tasks where such differences do show up is in *information seeking* tasks [5]. In our lab, one general investigation theme has been that of *exploratory search* [7]. In exploratory search, the user has a vague idea of what to look for, and develops further insights into the nature of the enquiry during the process of exploration. Applications of exploratory search can be found in many domains. For example, suppose that a researcher is looking for the output from a now-terminated project. No final project report has been archived because the project was terminated early due to organisational changes. However, the archive does contain project documents, including meeting minutes One of the later meetings mentions "a successful discussion with the patent lawyers",

[1] `http://www.musicbrainz.org`

leading the researcher to search the patent filing archive for documents mentioning one or more of the team members. The researcher did not set out with the goal of searching for patents, but adopted that sub-goal after refining the enquiry in the light of interim results.

The usability challenge in exploratory search arises from the tension between two opposing desiderata: that an exploratory search tool should be as open, generic and flexible as possible to allow the user to "follow their nose" during an investigation, vs. the well-established idea that highly usable interfaces are founded on a good understanding of specific user requirements, including users' goals, tasks, background and needs. We would like open-ended, generic exploratory search interfaces providing the fewest restrictions on user behaviour. Conversely, we desire constraints around which to structure the interaction model, since offering too many interaction choices reduces usability [8].

Exploiting the semantic annotations on the data may add further choices that the user has to contend with. Consider the familiar example of semantic search[9]. Users benefit from an interface which provides the means to disambiguate different senses of homonyms, to broaden a search to include semantically broader or similar terms, or to narrow a search to more specific interpretations. Doing so depends on the search application understanding which terms are to be treated semantically, and to match those terms to the underlying ontologies and instance stores. However, having the means to add more capability to semantic search by exploiting ontologies does not by itself address the challenge of how to provide those improved choices to the user while maintaining usability. The UI must provide some affordance for inspecting the translation from the user's input to the controlled vocabulary in the ontology (and correcting the translation if necessary), affordances to help the user understand which tactical choices are available to manipulate the query, and affordances to help the user navigate the linked dataset and remain orientated towards the original query goal or revised sub-goals.

The challenges of providing effective user interaction affordances for semantic web applications are frequently compounded by the size of the underlying linked datasets. Very large sets of instances with very many properties make, for example, faceted browsing[10] difficult, since users may be presented with confusingly large sets of facet choices. Moreover, computing facet counts and other navigation aides while maintaining interactive time response times is challenging, particularly in large datasets and especially so when some of the data may be federated from remote sources.

2.2 Challenges distilled: the increasingly-intelligent semantic UI

The common theme in the brief selection of semantic web user interaction challenges above is the tension between the goal of helping the user work more capably, particularly in open-ended exploratory tasks and the goal of reducing interface complexity to manageable levels. My view is that the key opportunity in semantic web user interfaces *per se* is to be able to use semantic annotations to make the interface itself smarter and more helpful to the user, so that the system is shouldering more of the burden on the user's behalf. In essence, the opportunity is to exploit semantic annotations and structure to gradually increase the degree of automation in the interface itself.

Increasing automation is not a new goal: the research field of *intelligent user interfaces*[2] (IUI) is well-established. However, to date there has been relatively little investigation of the contribution that an underlying semantic store can make to the methods and aims of IUI. For example, using plan recognition to determine user goals may be easier when the user is interacting with semantically-meaningful information objects, rather than uninterpreted text strings or record fields. Building user profiles from the statistics of texts the user is exposed to has been unreliable; doing so from a semantically-annotated dataset may prove more robust.

3 Ongoing research

In this section, I briefly outline some of the ongoing research work in our lab addressed towards some of the goals outlined above.

3.1 Information-management workbench

HP customer CIO's, in common with large organisations everywhere, report that a key challenge for their IT groups and individual staff members is effectively managing increasing volumes of information. Together with partners in HP's software business, we have been investigating what semantic web software might contribute towards the *enterprise information management* (EIM) problem. Our case study involved the problem of *e-discovery*. This has many features common to the archive use case published for this workshop. In e-discovery, attorneys in an organisation have a legal duty to locate, review and deliver all relevant documents germane to a legal dispute, while simultaneously protecting commercial confidentiality as much as possible. Relevant documents may range from managed content in a content-management store, through email messages, to text, MSOffice or other files on individual employees' hard drives. Each individual legal matter is different, so the user's search goals and best strategies will vary markedly from one case to the next.

During our investigation, we created a basic prototype to explore some key hypotheses about our approach to the task. Key characteristics in the prototype were:

- resultsets are first-class objects, which enabled us to characterise the exploration process as a sequence of moves in the space of all resultsets. In particular, this gives a framework for creating a common view of various interface affordances. For example, selecting the value of a facet in a faceted-browsing view moves from a resultset showing one type of resource (e.g. documents) to

[2] `http://www.iuiconf.org`

another resultset showing a smaller set of the same type. The edge between the two states is labelled with the operation (filter) and value (the facet value selected). Alternatively, pivoting from a display of documents to the value of the `dc:creator` property moves to a resultset of person resources.

- free-text fields were dynamically matched, using lookahead search, to terms in the underlying ontologies. This makes it easier for users to enter terms as either free-text or controlled vocabulary elements.
- a simple query planner used the structure of the ontology and the instance data to suggest queries based on a user's partial query. For example, entering the name of an organisation from the controlled-vocabulary (for example, the name of our lab: the Enterprise Informatics Lab, or EIL), and a free-text query string (for example: "rdf"), and a desired result type of `Document`, the query planner would suggest the query "`documents with dc:creator ?a and text-match to 'rdf', where the affiliation of ?a is EIL`". However, changing the result type to `Person` would result in a query of the form: "`persons with affiliation EIL, who are dc:creators of document ?d, where ?d has text-match to 'rdf'`." Our hypothesis is that query suggestion of this form, together with affordances for choosing alternatives when the planner's top suggestion does not match the user's goal, will allow users to construct complex queries more easily than, say, facet browsing with pivoting. However, we do not yet have experimental data to validate that hypothesis.
- the same query suggestion capability is used to provide navigation suggestions to users while browsing through resultset space: given a resultset, the query suggester examines the topology of connections to and from the resources in the resultset to generate a ranked list of moves (i.e. related queries) from the current node in resultset space.
- key user actions, including the history of resultset-space moves, is recorded in a user model. We have two goals for this user model: to use it as an input to the query suggestion capability to improve ranking of relevant queries, and to use it as a means of generating reusable enquiry patterns and strategies. In principle, though we have not done so yet, extracted enquiry strategies could be shared among members of a community or workgroup.

This work is documented in more detail in Reynolds et al[11]. While we do not yet have formal evaluation results, an obvious usability issue that arises in this approach is to generate meaningful labels for the suggested queries. This remains an open issue.

3.2 Mixed-mode visual and non-visual interface

User-interfaces to semantic web systems to date have typically been graphical, whether by facet browsing, variants of graph drawing, or graphical visualisation. Each of these techniques has a useful role in supporting user interaction with semantic web datasets. Relatively less attention has been paid to non-visual interfaces. An ongoing experiment in our lab is developing an interface tool, one of whose primary interaction modalities is a chat window inspired by instant messaging tools. For example, the above search example might be re-rendered as shown in figure 1.

```
prox: Query tool online
me: enterprise informatics
prox: the Enterprise Informatics Lab in HPLabs?
me: yes
me: "rdf"
prox: I can connect the Enterprise Informatics Lab to the keyword
"rdf" by searching for Documents matching "rdf" and document creator
member-of Enterprise Informatics Lab ?
me: yes
prox: 32 matches of type Document
me: show me
prox: I can display a facet view or a tabulator table
me: facet
```

Fig. 1. sample dialogue with chat interface

The example dialogue in Fig 1 is synthetic (the actual code is still in development), but the key hypothesis underlying it is that turn-taking dialogues are a natural fit for the iterative exploration moves in exploratory search. Two of the central technical challenges are once again the generation of sufficiently fluent text output that the system's dialogue turns are not too unnatural, and developing algorithms and rules that can make use of the iteratively constructed context to suggest reasonable dialogue moves by the system.

As shown by the last three lines of the sample dialogue, we do not intend the dialogue to be restricted solely to text chat. In cases where other modes of interaction are more natural, the user should be able to switch smoothly to (say) facet browsing, then return to the chat interface once the dataset has been filtered.

A further issue that arises with this approach is the identity of the other interlocutor. Is `prox` an agent operating on behalf of the user, or of the data-owner, or some other intermediary? How do different results affect the user's perception of trust in the results?

3.3 Ongoing research: coda

The goal of our present research is not to seek the single idealised user interface for the domains we are focusing on. Rather, it is my view that we need to gather more insight into semantic web UI patterns and techniques, determine which are appropriate for different situations, and understand their relative strengths, weaknesses, and open research challenges. The common theme to our investigations is to use the structure in the semantic dataset to drive user interaction affordances in new ways, gradually increasing the amount of automation we can use to assist our users.

4 Conclusion

In my view, the most interesting challenge and opportunity of semantic-web user interface research is to make use of ontologies and other semantic information to drive the user interaction itself. The projects reported above are early investigations into particular approaches to that goal. One of the key hypotheses underlying our approach is that iterative interactions with semantically labelled information sources will provide new tools to capture user models and context, which in turn will enable increased used satisfaction and effectiveness through improved automation. However, while early results are promising, this hypothesis remains as yet unverified.

References

1. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: Dbpedia: A nucleus for a web of open data. Lecture Notes in Computer Science **4825** (2007) 722
2. Miles, A., Perez-Aguera, J.: SKOS: Simple knowledge organisation for the web. Cataloging & Classification Quarterly **43**(3) (2007) 69–83
3. Bizer, C., Lee, R., Pietriga, E.: Fresnel-a browser-independent presentation vocabulary for rdf. In: Proceedings of the Second International Workshop on Interaction Design and the Semantic Web, Galway, Ireland. (2005)
4. Berners-Lee, T., Chen, Y., Chilton, L., Connolly, D., Dhanaraj, R., Hollenbach, J., Lerer, A., Sheets, D.: Tabulator: Exploring and Analyzing linked data on the Semantic Web. In: Proceedings of the 3rd International Semantic Web User Interaction Workshop. Volume 2006. (2006)
5. Heath, T., Dzbor, M., Motta, E.: Supporting user tasks and context: Challenges for semantic web research. In: Proc. ESWC2005 Workshop on End-User Aspects of the Semantic Web (UserSWeb). (2005)
6. Battle, L.: Preliminary inventory of users and tasks for the semantic web. In: 3rd Intl. Semantic Web User Interaction Workshop. (2006)
7. Marchionini, G.: Exploratory search: from finding to understanding. Communications of the ACM **49**(4) (2006) 41–46
8. Schwartz, B.: The paradox of choice: Why more is less. Harper Perennial (2005)
9. Bonino, D., Corno, F., Farinetti, L., Bosca, A.: Ontology driven semantic search. WSEAS Transaction on Information Science and Application **1**(6) (2004) 1597–1605
10. Hearst, M.: Design recommendations for hierarchical faceted search interfaces. In: ACM SIGIR Workshop on Faceted Search. (2006)
11. Reynolds, D., Dickinson, I., Grosvenor, D.: A query refinement model for exploratory semantic search. Technical Report HPL-2009-167, HP Laboratories (2009)