

Exploring Semantic Web Datasets with VisiNav

Andreas Harth and Paul Buitelaar

Digital Enterprise Research Institute
National University of Ireland, Galway

1 Introduction

With organisations and individuals increasingly publishing data online, new applications leveraging integrated web data become possible. In this poster we present VisiNav, a fully-implemented application for searching and navigating RDF datasets collected from the open web. VisiNav has been deployed in several pilots over both web and domain-specific datasets. Here, we present VisiNav from an end-user perspective: what do prospective users have to know in order to use VisiNav for exploring web datasets in ways that go beyond the capabilities of current systems.

In contrast to web search engines that deal with documents, VisiNav assumes an object-oriented data model (essentially the RDF data model). Objects have attributes and links to other objects. Consider a dataset containing objects of type Person. A Person has a name; a Person knows another Person; a Person is based near a Location.

Data on the web takes an open world view: vocabularies are suggestions as to how to structure data, and unlike database systems, instance data might deviate from the specified vocabulary. Moreover, given the minimal amount of coordination among data publishers, there may be competing vocabularies describing the same domain.

2 Query and Visualisation Operations

The leniency in adherence to a schema has implications on the type of user interaction that system can offer. Given the diversity of data and vocabularies, navigation operations over these datasets should be universally applicable. In the following, we first describe a set of query operations that solely operate on the object structure of the data, and second describe a set of visualisations that are dependent on certain schema-level elements.

VisiNav employs four basic operations: keyword search, object focus, path traversal, and facet selection.

- **Keyword search** Users supply a set of keywords, and the systems returns a ranked list of objects that match the specified terms. For example, a search for “tim berners lee” returns all objects matching the keywords.
- **Object focus** Users can bring an object into focus by clicking on a link denoting the object. For instance, by clicking on the Person object “Tim Berners-Lee”, the system returns a page listing attributes of the Person,

and links to other objects, such as the Person instances connected via the “knows” link.

- **Path traversal** User can traverse a path (such as the “knows” link) and arrive at a new result set. From here, users may either navigate to one of the resulting objects, or again traverse a link (e.g., the “based near” link) to arrive at a new result set (the locations of people that Tim Berners-Lee knows)
- **Facet selection** To restrict a result set further, users may select “facets” (attribute/value or link/object pairs). For example, a user has the ability to further restrict the set of locations using the “parentFeature” “Europe” facet, which restricts the displayed locations to the ones located in Europe.

Following the previous steps in sequence yields, with a few clicks only, the locations in Europe where acquaintances of Tim Berners-Lee are based.

The operations used are a combination of RDF browsing primitives [1] and faceted browsing [2]. Many browsing systems use these primitives or subsets thereof, suggesting a consensus in the community that these features are useful in interactive browsing systems. In contrast to linked data browsers which fetch the data to be displayed on demand and thus only allow object focus and path traversal operations, VisiNav integrates data more tightly and allows to query the entire dataset via keyword search and facet selection operations.

Although the navigation operations are schema-independent and operate only on the structure of the object base, visualisations are tied to certain schema-level constructs. For example, visualising a location on a map requires a value for latitude and longitude. Hence, VisiNav suggests visualisations depending on the vocabularies used in describing the result objects. For instance, when the result set contains points described in WGS84 – a vocabulary for representing geographical points – the user may select a map visualisation with the points contained in the result set marked.

3 Conclusion

VisiNav is a system that empowers casual and novice users to construct complex queries using four search and navigation primitives. Users may visualise the query results using external services or export the data to other applications. The prototype is available online¹ and currently contains over 1m objects described in over 15m RDF statements collected from the web.

References

1. N. Athanasis, V. Christophides, and D. Kotzinos. Generating On the Fly Queries for the Semantic Web: The ICS-FORTH Graphical RQL Interface (GRQL). In *Proceedings of the 3rd International Semantic Web Conference*, pages 486–501, 2004.
2. K.-P. Yee, K. Swearingen, K. Li, and M. Hearst. Faceted metadata for image search and browsing. In *Proceedings of the SIGCHI Conference*, pages 401–408, 2003.

¹ <http://visinav.deri.org/>