A Demonstrator for Parallel Faceted Browsing

Sven Buschbeck[1], Anthony Jameson[1], Raphaël Troncy[2],
Houda Khrouf[2], Osma Suominen[3], and Adrian Spirescu[1]

[1] DFKI, Saarbrücken, Germany
anthony.jameson@dfki.de
[2] EURECOM, Sophia Antipolis, France
raphael.troncy@eurecom.fr
[3] Aalto University, Helsinki, Finland
osma.suominen@aalto.fi

Abstract. The data exploration paradigm of faceted browsing has been applied successfully in many systems that provide access to either semantic data or more conventional databases. A general limitation is that the user can examine the results for only one query (i.e., one combination of facet value constraints) at a time. But there are many everyday tasks for which it is desirable to be able to compare the results for two or more queries or to refine several queries incrementally in parallel. In this paper, we introduce *parallel faceted browsing* which enables the user to construct a tree of queries and associated results whose structure captures the commonalities among the queries. The entities represented in this demonstrator are events associated with Helsinki's role as the 2012 World Design Capital, as well as related events from other data sources that took place in Finland during the same period. Semantic Web technologies are used in the back-end for representing and reconciling those heterogeneous data sources. Several types of task have been identified in which parallel faceted browsing offers added value. Careful attention has been paid to interaction design with the goal of making this inherently more complex paradigm as intuitively understandable as normal faceted browsing. Readers are encouraged to try the EVENTMAP demonstrator itself at `http://eventmap-ui.appspot.com/`.

1 Parallel Faceted Browsing

Innovative Computational Model. Faceted browsing (or faceted search [1]) is a widely used paradigm for the exploration of a large repository of entities, each of which has a number of attributes (or facets), each of which can take different values. In the simplest case, a user specifies values of one or more facets in order to narrow the focus down to some subset of the entities in the repository, which are then displayed. The user can then proceed to specify values of additional facets to narrow the focus further, until he or she has identified a sufficiently small set of interesting items. This procedure can also involve keyword search.

Although faceted browsing has been applied with great creativity and success in the semantic web field (e.g., the work on MSPACE[4]), all of the realizations that we are

[4] `http://research.mspace.fm/mspace`

aware of share a general limitation: At any given time, the user can see the results corresponding to only one query (i.e., one set of constraints on values of facets). With the new paradigm of *parallel faceted browsing* (PFB) introduced here, the user can create in parallel a number of queries and see their results displayed simultaneously in a way that visualizes their relationships to each other. Some other interfaces for the exploration of semantic data such as the PARALLAX[5] have used terms that sound similar to "parallel faceted browsing" but these approaches are in fact quite different.

Tasks, Domains and Applications. There are several types of task with which PFB offers added value relative to normal faceted browsing. We list them here using examples from the domain of events associated with Helsinki's role as the 2012 World Design Capital (WDC).

1. Creating a display of a number of entities organized into selected subsets: A person may be interested in several particular types of WDC events occurring in different places during different periods of time. With PFB, she can straightforwardly create a structured display of the events in all of the subclasses that correspond to her interests.

2. Conducting several interrelated searches in parallel: A WDC visitor who plans to attend events on three successive days can best search for events on these three days in parallel rather than in succession, since the results should be not only individually appropriate but also compatible with each other (e.g., in terms of location and topic coverage).

3. Searching and exploring without clearly defined requirements and preferences: Even if a WDC visitor is looking for just one event to attend, she may not have adequate knowledge or predetermined preferences with regard to the various facets. With PFB, she can do justice to this uncertainty by in effect constructing a number of queries in parallel, pursuing at each point in time the lines of exploration that seem most promising in view of the sample of results that they are returning. This strategy is roughly comparable to the computational search strategy of beam search.

Normal faceted browsing has already been found useful in a wide variety of application domains, by no means restricted to those involving semantic data, including e-commerce sites and digital libraries [1,2]. Since tasks of the three types just listed can occur in most or all of these domains, the potential applicability of PFB is very broad.

Human Factors. Because of this broad applicability, PFB needs to be realized in the user interface in a way that makes it easily comprehensible to just about any computer user. Iterative user testing so far has suggested that this goal can be achieved through the use of an interface metaphor [3] of "card sorting": The user starts with a large stack of cards, each of which describes one entity in the repository. Specifying a facet value constraint is visualized as the operation of selecting a subset of a stack of cards to yield a smaller stack, whose cards the user can inspect whenever he or she likes.

2 Data scraping, Representation and Enrichment

The dataset used in this demonstrator corresponds to events that take place in 2012 in Finland. More precisely, we build a wrapper around the three RSS feeds of the World

[5] http://www.freebase.com/labs/parallax/

Design Capital in Helsinki web site[6] to convert the description of events into RDF. We also add more events that took place in the same locations from the EventMedia dataset which is obtained from three large public event directories (Last.fm, Eventful and Upcoming) [4]. We used the LODE and Media Ontology respectively for expressing linked data descriptions of events and photos. The LODE ontology[7] is a minimal model that encapsulates the most useful properties for describing events. The goal of this ontology is to enable interoperable modelling of the "factual" aspects of events, where these can be characterized in terms of the four Ws: What happened, Where did it happen, When did it happen, and Who was involved. This data is fed into a SPARQL endpoint within the specific `http://data.linkedevents.org/wdchelsinki` named graph. As of today, the dataset contains 1140 events and more than 102,130 triples.

We also perform several enrichments on the data. First, we use Harava[8] to add extra information for categorizing the events. Hence, several `dc:subject` properties are added when events can be described with terms from the KOKO ontology. Second, we perform several interlinking processes using SILK attempting to discover connection between the agents and the venues and locations with Linked Data. We compare agents' names using Jaro, an efficient metric to match short strings and we keep a high threshold of about 0.96, so maintaining a high precision. Similarly, we enrich venues' descriptions using the Geonames dataset. The similarity function combines the geographic distance between two venues using their GPS coordinates with Jaro applied on labels. Overall, we generate more than 22,513 `owl:sameAs` links.

3 Architecture and Key Components

The demonstrator is composed of (a) a client that runs in the user's browser; (b) a server-side application that runs in Google's APPENGINE, which is a cloud-based service; and (c) a data repository provided by a single-server VIRTUOSO instance.

The implementation of the client and server is supported by the GOOGLE WEB TOOLKIT (GWT)[9] framework. It provides cross-browser-compatible widgets for creating rich user interfaces (UIs), handles client server-communication via remote procedure calls, and provides several multipurpose APIs. We use some of the APIs[10] to create a persistent, multilevel caching. In-memory caches are integrated into each running client instance and each server instance. The third layer of caching is a shared and persistent cache. Another API is used to parallelize data acquisition from the data repository.

One of the system's internal data structures is essentially a tree that implicitly defines a set of interrelated complex queries, each of which is a conjunction of simpler

[6] `http://wdchelsinki2012.fi/en/rss` for the English RSS feed

[7] `http://linkedevents.org/ontology/`

[8] `http://www.seco.tkk.fi/tools/harava/`

[9] `https://developers.google.com/web-toolkit`

[10] The following APIs have been used: the NOSQL datastore for persisting cached data, the MEMCACHE JAVA API for in-memory caching and the URLFETCH API to execute multiple SPARQL queries in parallel.

queries associated with individual nodes of the tree. The system accesses the data repository via SPARQL queries[11] of two major kinds: (1) Populate the facets represented in the system with values that can be shown in the selection menus; the most frequently used values are displayed, while others can be typed in by the user. (2) Gather detailed data about events that satisfy the constraints specified by the user.

The queries are generated semiautomatically by a combination of templates and the particular facet type. Currently, the system supports two facet types: (a) value list, for example the artists that perform during an event; and (b) number ranges, for example the beginning and end date of an event.

4 Conclusion and Future Work

The EVENTMAP demonstrator appears to be a suitable candidate for the IESD 2012 Challenge in that it incorporates an efficient implementation of an innovative computational and conceptual model that offers advantages in a broad range of practically important tasks and application domains and that is being made accessible to all computer users through careful attention to interaction design. In future work, the PFB approach can be generalized to other types of queries than those which specify combinations of facet values, for example: variants of a given keyword search query that involve different expansions or variations; different evaluation functions for identifying sets of objects whose overall evaluation exceeds a certain threshold.

Acknowledgments

The research leading to this paper and demonstrator was partially supported by the European Union's 7th Framework Programme via the projects GLOCAL - Event-based Retrieval of Networked Media (GA 248984), LinkedTV (GA 287911), ALIAS - Adaptable Ambient Living Assistant (AAL-2009-2-049) and the Finnish Funding Acency for Technology and Innovation (Tekes) via the projects SUBI and FinnONTO. The development and evaluation of the EVENTMAP demonstrator is being funded by the activity EVENTMAP (Event-centric Multimedia content Access Platform, TFMC 12116) of the EIT ITC Labs. The authors would like to thank the students of the 2012 course on User Interface Design at Saarland University who designed and user-tested mockups of PFB in a number of different application domains.

References

1. Tunkelang, D.: Faceted Search. Morgan & Claypool, New York (2009)
2. Hearst, M.A.: Search User Interfaces. Cambridge University Press, Cambridge, UK (2009)
3. Neale, D.C., Carrol, J.M.: The Role of Metaphors in User Interface Design. In Helander, M., Landauer, T.K., Prabhu, P.V., eds.: Handbook of Human-Computer Interaction (2nd ed.). North-Holland, Amsterdam (1997) 441–462
4. Khrouf, H., Troncy, R.: EventMedia: a LOD Dataset of Events Illustrated with Media. Semantic Web journal, Special Issue on Linked Dataset descriptions (2012)

[11] The SPARQL endpoint is available at `http://eventmedia.eurecom.fr/sparql`