

# Faceted Browsing, Dynamic Interfaces, and Exploratory Search: Experiences and Challenges

Robert Capra, Gary Marchionini  
School of Information and Library Science  
University of North Carolina at Chapel Hill  
100 Manning Hall  
rcapra3@unc.edu, march@ils.unc.edu

## Introduction

The Relation Browser (RB) is a graphical interface for exploring information spaces, developed by the Interaction Design Lab at the University of North Carolina at Chapel Hill for use in research on how to support users' needs to understand and explore information. In this abstract, we describe the Relation Browser, results of recent studies, and the design goals for the next-generation RB in current development. At the workshop, we will demonstrate the current RB and a prototype of our next-generation RB.

## Current Relation Browser (RB++)

The Relation Browser is designed as a tool for understanding relationships between items in a collection and for exploring an information space (i.e., a set of documents). It has been through a number of major design revisions [2,3]. The current version is called the RB++ (Figure 1). Facets are central to the RB and are displayed at the top of the interface. Results of queries are shown at the bottom of the screen in tabular format. Blue bars and numbers to the left of each facet category indicate how many documents match that category. Previews of queries can be issued by simply mousing over facet categories. For example, mousing over the topic "inflation" will dynamically update the blue bars and number to reflect only documents that are in the inflation topic. By clicking on a facet category and then pressing the "Search" button, results are retrieved and displayed in the lower part of the screen. Results are tightly coupled with the facet categories and with search boxes displayed above each result field. For example, typing "occupations" into the text box above the title field will not only narrow the results shown at the bottom of the screen, but will also update the blue bars and numbers shown at the top of the screen. The current RB allows searching the metadata fields in the result sets, but does not support full-text keyword searches. Because of this, the current RB encourages a "facets first" strategy of exploration.

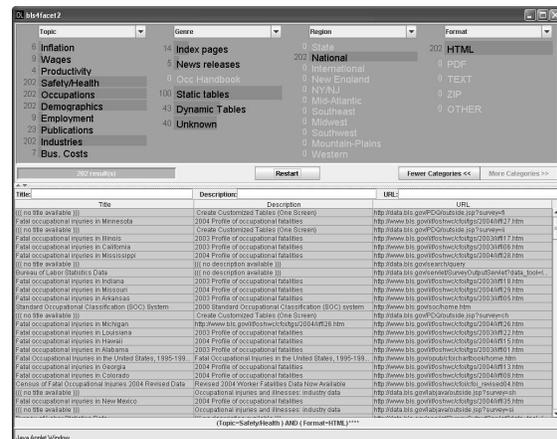


Figure 1. Relation Browser displaying BLS data

The RB is designed as a generic interface that can accept and display data for many different types of collections. RB instances have been developed for a variety of data sets including U.S. federal statistics (Bureau of Labor Statistics, Energy Information Administration, NSF Science and Engineering Indicators), classical music, the Open Video collection, a university movie database, the CIA World Factbook, and a database of roller coasters. A new version of the RB, called RB07, is currently in development.

## Structure and Interaction Study

In the summer of 2006, we conducted two studies to compare three different interface styles (handcrafted web site, simple facet interface, and the Relation Browser) for three different task types (simple lookup, complex lookup, and exploratory search) for the U.S. Bureau of Labor Statistics (BLS) web site data. This data set was fairly large (over 67,000 documents) and semi-structured, providing a good test set for examining facet use for data that does not have a fully defined set of metadata on which to organize.

The BLS web site uses a polyhierarchical structure with two levels of topics displayed on the home page. The design of the BLS site was handcrafted based on a series of needs assessments and user studies. For the simple facet (SF) and Relation Browser (RB) interfaces, we created a facet set using a variety of semi-automated techniques [1]. The results of the studies surprised us: we found no significant differences among the three interfaces for measures of task completion time, accuracy, confidence, or mental effort. The semi-automated facet interfaces (SF & RB) performed just as well as the handcrafted BLS site and no significant two-way interactions between task type and interface were found. These results indicate that facet sets generated using semi-automated methods can provide useful interfaces to large, semi-structured data sets.

Perhaps even more interesting than the quantitative results are the qualitative data and observations we made during the study. One of the common observations was that our participants (recruited from the UNC community, aged 18-35) often attempted to use a “keyword search first” strategy, even in the interfaces that did not directly support this. The BLS web site provided a keyword search feature one click away from the home page, but the SF and RB did not (they both only allowed search on the metadata). Related to this, a number of users expressed feelings of “not being in control” for the RB interface. The current RB strongly emphasizes facets and encourages users to adopt a “facets first” strategy that may be at odds with users’ preference for using keyword search first. Despite this conflict, many users appreciated and noted the benefits that facets provide. Thus, we concluded that interfaces should support agile, user-controllable searching and browsing. This has been a design goal for the next-generation of the Relation Browser, described in more detail in the next section.

## **Next-Generation Relation Browser (RB07)**

One of the primary goals of the RB is to provide a tool for exploring data spaces – for gaining a better understanding of the documents and how they are related to each other. Adding full-text search support while maintaining agile exploration is the challenge for our next-generation RB, called RB07.

RB07 is currently in development and we will show a demonstration of the prototype at the workshop. The new design includes flexible facet views so that the user can control how the facets and document counts are presented. The initial two facet views are the traditional view as in the current RB++, and a representation of the facets as a dynamic tag cloud. The new design also includes a choice of ways to view the results of a search. A “grid view” that is similar to the current RB++ results table provides a way to see a concise summary of essential metadata about matching records. A “list view” presents results in a format that is typical of search engines with a document title, matching text snippets, and a URL (if available). The new views for facets and results continue to be tightly coupled using an extensible architecture that can support additional “plug-ins” for displaying facets and results.

Whereas the current RB++ is a pop-up applet that displays in a new window, the new RB07 is designed to be an embedded component of a web page, providing tighter integration with an existing web site if desired. For the implementation of the new RB07, we considered Java and JavaScript as two well-supported client-side languages. JavaScript has easy-to-use access to its surrounding web page, which would have benefits for RB-website integration. However, in the RB, the dynamic updating of the display based on each mouse movement over a facet category triggers a series of computations that are a function of the number of facets, categories, and documents. After extensive testing with JavaScript, we found that it was not fast enough to support the dynamic updating aspects of the RB on current hardware for document collections larger than 5000 to 10000 documents. Thus, we are developing the new RB07 as a Java applet.

Support for full-text keyword searching is being implemented using the Apache Lucene search engine as packaged in the Apache SOLR search server. Although SOLR provides some support for facets, we are not currently leveraging that support, but instead implement facets through the RB itself (this is in large part due to the need to do fast dynamic updates for the mouseovers). Documents are linked between SOLR and the RB07 through a unique document identifier. One of the interface design issues that arose during our development is how to provide clear distinctions between using keywords to search within a result set versus starting a new keyword search. In our current design, new searches are started using a keyword textbox at the top of the display and refinement searches are issued through a textbox closer to the result

set. We hope that the new RB07 will help address users expectations of how to explore document spaces, while still providing powerful interface components for seeing relationships in the collection and refining result sets.

## Future Questions

While facets are widely regarded as being useful to information seeking, especially in large data sets with complete metadata that can be used as facets (such as shopping domains), a number of research questions still remain: How are facets used during the information seeking process? When, how, and why are facets helpful (i.e. facets first versus facets to refine)? How is facet use affected by task type? What role do facets play in exploring and gaining an understanding of the information space? Do facets help users refine documents they have seen before, acting as waypoints?

Understanding these issues will help us create better tools for information discovery and exploratory searching and we expect that user studies and other empirical investigations will lead to answers to these questions and guide future system designs.

## References

- [1] Capra, R., Marchionini, G., Oh, J. S., Stutzman, F., and Zhang, Y. (2007). Effects of structure and interaction style on distinct search tasks. In Proceedings of the 2007 Conference on Digital Libraries (Vancouver, BC, Canada, June 18 - 23, 2007). JCDL '07.
- [2] Marchionini, G. & Brunk, B. (2003). Toward a General Relation Browser: A GUI for Information Architects. Journal of Digital Information, 4(1), <http://jodi.ecs.soton.ac.uk/Articles/v04/i01/Marchionini/>
- [3] Zhang, J., and Marchionini, G. (2004). Coupling Browse and Search in Highly Interactive User Interfaces: A Study of the Relation Browser++. Proceedings of the 4th ACM/IEEE Joint Conference on Digital Libraries (Tucson, AZ: June 7-11, 2004), 384.