Toward a General Relation Browser: A GUI for Information Architects
Gary Marchionini and Ben Brunk
University of North Carolina at Chapel Hill
Interaction Design Laboratory
[march, brunkb]@ils.unc.edu

INTRODUCTION
Web sites for corporations, government agencies, and other enterprises often are comprised on tens of thousands of web pages (whether static or dynamically generated). Information architects and webmasters provide site maps or visualizations to help users gain quick understandings of what is and what is not available in the site and how the site is structured. Such overviews aid people in formulating queries and in navigating the site (Greene et al., 2000). Rather than providing only static site maps and visual overviews, the electronic medium can be leveraged to create interactive overviews that support a variety of user interests. Creating interactive overviews requires substantial organizational as well as technical effort in today's WWW environment. HCI research in pre-web days aimed to create User Interface Management Systems (UIMS), analogs to database management systems, that allow the designer to abstract away from the low-level details of electronic storage to focus on entities, relationships, and creating customizable views for end users (e.g., see Dewan, 1990; Kim and Foley, 1990; Myers et al., 1993). As the web matures, new tools for designers and information architects have emerged for created structured documents, however, creating interactive interfaces requires customized programming for each website and alternative view within the website. What is needed are tools and services that manage the graphical user interface (GUI) details and allow information architects to focus on organization and interaction design.

This paper presents the case of ongoing efforts to develop and test generalizable user interfaces that provide interactive overviews for large-scale websites, portals, and other partitions of webspace. The interfaces are called Relation Browsers (RB) because they help people explore the relationships across different attribute sets, thus enabling understanding the scope and extent of the corpus through active exploration of different 'slices' defined by different attribute value juxtapositions. The RB is a generic interface that can be applied to many kinds of webspace partitions. The paper is organized as follows. The relation browser (RB) concept is first examined and the context for the first prototypes discussed. Next, the first versions of the RB and the user studies are summarized. Subsequent iterations of the RB are discussed and examples of implementations for different applications are given culminating with current efforts to apply the design concepts in a digital government setting. Finally, the concept of the RB as the basis for an interface server is presented.

The Relation Browser concept is based on two beliefs: that examining the logical relationships among data attributes is a powerful strategy for understanding a data set; and such comparisons can be made intuitive through direct manipulation control mechanisms. Underlying the first belief are the assumptions that understanding a data set leads to more effective search and usage, and that one important kind of understanding is a realistic overview. Underlying the second belief is the assumption that active engagement in more effective for broader ranges of users than command languages or selections from abstract menus. These beliefs are supported by research and experience.

Logical relationships.
Large portions of cognitive science are devoted to determining what it means to understand a concept or object (e.g., Johnson-Laird, 1983; Quillian, 1968; Rosch, 1975), or in the case here, a partition of the WWW. To simplify discussion, we use 'concept' here to represent objects,

events, processes, websites, and partitions of webspace. Understanding begins with existence; i.e., the concept must be instantiated or recognized in the mind. Concepts do not stand alone— they are componetized, aggregated, and related to other concepts. Regardless of the theory (e.g., mental models, prototypes) relationships among components or other distinct concepts are crucial to definition and context. For example basic relationships such as 'part of' and 'isa' help people understand relationships among component objects. More subtle relationships are represented in language and tools such as thesauri (e.g., synonyms, broader/narrower terms) and ontologies aim to explicate these relationships (see Green and Bean (1995) for a taxonomy of relationships in text).

To understand a webspace partition, people must first understand what is in the partition: What is the nature of the information? What is its form and extent? How is it organized? What are the components (attribute sets[1])? How are they named? How are the attribute sets organized (categorized)? What are the attributes associated with these sets? How are they named? Are they categorical, ordinal, or interval? How many instances (items) are associated with each attribute? These questions require that an interface must represent what is available (and what is not) and the overall structure of the information to users—this is a primary function of information architecture. Once the basic attribute sets and their attributes are understood, more complex relationships can be examined. For example, beyond knowing that 'price' and 'location' are two attribute sets available in a real estate portal, people may want to examine various relationships between price and location. The easiest comparison is pair wise between specific attributes; for example, the properties with a specific price in a specific location (or the locations with properties at a specific price). The comparison could be based on binary existence (e.g. Do any properties exist for $100,000 on Elm Street?) or the number of houses that meet both these conditions (logical AND), or whether there are houses that meet either condition (logical OR), or exactly one of these conditions (exclusive OR). Arbitrarily complex logical relationships are precisely expressible in formal languages such as the predicate calculus or SQL. Carefully expressing a relationship yields a subpartition of the overall webspace partition and the aggregation of all such expressions would constitute a complete understanding of the overall partition. Of course, other than experts and frequent users, most people have great difficulty with formal languages and they do not need a complete understanding of a webspace partition.

The goal in the RB is to provide an overview of the webspace partition by providing short lists of attributes for a small number of attribute sets and simple interaction mechanisms (e.g., mouse movements) to explore relationships among attributes. Key design decisions are related to selecting attribute sets and attribute values (dependent on what data is available and how best to name these values for the user population). Key implementation challenges are related to acquiring the appropriate data (slicing the data by an attribute may make good sense from a user perspective but this may entail creating customized metadata for the interface). The benefit to users is making several, user-controllable slices available that reveal existence and extent, and allowing simple pair-wise explorations to be made.

Interactive control mechanisms.
Interactive interfaces that support searching and browsing in information spaces must provide appropriate representations for the information objects in the space and user control mechanisms tuned to those representations (Marchionini, 1995). Shneiderman has made the case for direct

---

[1] Database parlance uses entity set and attributes within entity sets to distinguish a primary concept of interest from the characteristics (attributes) composing the entity. Entities are then represented as tables and attributes as fields in the table. We adopt a more precise parlance, referring to the concept as an attribute set (to show it is composed of attributes).

manipulation interfaces (1983) that give realistic representations of objects (rather than abstractions) controllable with physical actions (rather than abstract commands) that are "rapid, incremental, and reversible" and closely couple display with action. Hutchins, Hollan and Norman (1986) and others have reinforced this concept and it has become very influential in HCI theory and practice.   Direct manipulation is instantiated in information spaces by the many examples of dynamic queries that Shneiderman (1993) and his colleagues have developed (see Hearst, 1999 for a review of interactive interfaces for information spaces).

Control mechanisms are defined by input device (e.g., keyboard, mouse, microphone) and the actions these devices afford to people using them, output device (e.g., display, speakers) and the actions these devices afford to people using them, tasks supported by the information system, and mappings between people's actions and tasks. For a device like a single mouse with a single button, typical actions include: moving the mouse or holding it stationary, holding down the button, and pressing and releasing the button (clicking). When timing is considered, these primitive actions can be extended (e.g., two clicks within an interval becomes a 'double click,'). Additionally, these primitive actions can be combined to create more complex actions (e.g., hold mouse button down and move the mouse is a 'drag'). The resulting set of mouse actions can then be mapped to particular tasks. For example, dragging over text can highlight it for some larger purpose (changing font, cutting, etc.). An important part of designing interactive interfaces is determining the mappings between the representations of interest to users and the specific actions that allow them to control those representations. Some of these mappings are so pervasive that they become conventions or standards. For example, clicking on a hyperlink in most WWW browsers selects the link and generates an HTTP request to the server. Clearly, designing an interface in which a click on a hyperlink did something else would be confusing to users. In the RB work, we have adopted the mouseover (hovering) as an important user control mechanism. The action maps onto a process that reveals the relationship between the attribute hovered over and all attribute values in a second attribute value set. Continuing the real estate example, hovering over a location value (e.g., a zip code or street) would reveal how many properties are available in each of the price ranges categories.

THE RELATION BROWSER
The RB is strongly influenced by the dynamic query perspective and aims to couple simple representations and user control mechanisms in order to provide alternative views (slices) of the corpus, allow users to 'look ahead' to quickly determine whether to examine objects further (usually requiring an HTTP request), and minimize mouse clicks and subsequent page loads. Related approaches to these challenges include the Attribute Explorer (Tweedie et al., 1994; 1996) that visualizes with pictographs and lines the relationships among lists of attribute sets. Wittenburg and his colleagues (Wittenburg et al., 2001; Lanning et al., 2000) have used visual bars (bargrams, sliding rods) to dynamically display attribute values in image databases. In the case of the RB, the first efforts to juxtaposition lists of attribute values and highlight relationships across those attribute values as users hovered the mouse over a value were developed for interface prototypes for the Library of Congress National Digital Library (Marchionini et al., 19; Plaisant et al., 19). The first RB prototype resulted from work with the Bureau of Labor Statistics to develop an alternative site map for the FedStats website.

RB Version 1.
The RB was designed as an alternative site map for the Fedstats portal to U.S. government statistics (www.fedstats.gov). The portal provides access to about 200 websites and a million webpages and files from more than 70 agencies. The first goal was create the architecture to support information exploration. Three input streams provided guidance: user needs assessments, actual data available in federal websites and underlying databases, and the existing structure of

the various federal statistical websites and the Fedstats portal. Because Fedstats crosses all federal agencies, it was important to identify attributes that applied to as many types of statistics as possible. The challenge was to find attribute sets that were common to as many websites as possible but still provided some level of discrimination power for site map users.

From our previous work with statistical agencies, we knew that novice users of statistical websites are task and topic oriented rather than agency or data oriented. On the other hand, expert users can focus on specific data attributes such as source (survey) and variable technical name. For the RB, we decided that specific attribute sets like agency or survey, although they may apply across all websites, required too much knowledge about federal statistics to be helpful to novice or casual users. The first step was to investigate "topic" as the primary attribute set for the RB. The Fedstats and White House websites were studied since they presumably reflect the data available as well as designer expectations about user preferences for topical entry points. Taking these two organizational schemes as a starting point with the intention of coming up with a topical scheme that would be familiar and useful to a broad range of citizens, a variety of classifications were explored.

Based on discussions with the BLS and Fedstats teams in the fall of 1998 and examination of other websites that organize public information, consensus was reached that space (geographic region) and time (when the data were collected) were important attributes to include to characterize data. Because we felt it was important to help citizens quickly understand what types of data were and were not available, we aimed to include data type and data volume as attributes even though there was not consensus on their value or how they could be defined. It is important to note that the decisions about attribute sets and attributes are very much a political process where many stakeholders must agree. For example, considerable discussion led to the selection of 'substate' as the attribute for regional data aggregated at the county or equavalent level because not all U.S. states have counties. Using these first organizations, a prototype was demonstrated and discussed with the agency teams and final agreement was made on topic as the most crucial attribute set (using the 14 Fedstats program areas as topics), data type, region, and date as three other attribute sets, and data volume (number of websites) as the common property for all attributes. A template was developed to assist in collecting data and 196 websites were indexed (top page and one link deep for each) by hand.

In designing the RB, several control mechanisms were considered, including: hypertext links, expandable (cascading) hierarchical lists, tabbed views, zoomable space, magic lens', mouse-over pop-ups (tooltips), coordinated lists (supermenus), and animations. Based on our experiences with previous interface designs for an educational digital library (Baltimore Learning Community) and the Library of Congress American Memory digital library, we adopted coordinated lists with mouseovers as the primary relationship exploration mechanism with tabbed views and hypertext links as secondary mechanisms for changing pair wise exploration and selecting a specific website respectively. Paper and electronic prototypes were mocked up and discussed with the agency teams before a first Java prototype was developed. There is a RB website http://squash.ils.unc.edu/rave that provides a history of the different versions and more technical details on how the versions evolved. Marchionini (1999) provides details on the development rationale and process.

The prototype was developed in early 1999 and was user tested in late April and early May and that year. Figure 1 illustrates this prototype. The topic list was implemented as specified in a paper mockup with parenthetical values for the number of websites available for each topic. The data type list was also based on the paper mockup with the addition of a scale that allowed estimates for the number of websites containing that type of data. One significant difference
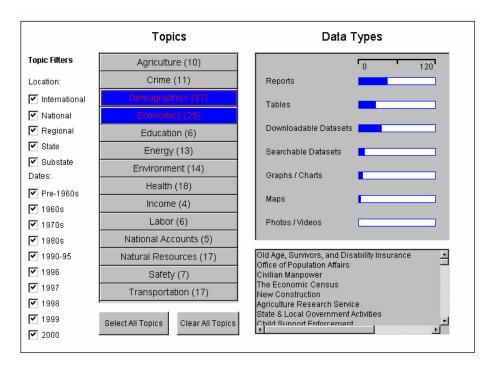
Figure 1. Early RB Prototype Interface (used in first usability study)

between the Java prototype and the paper mockup was the filters. Check boxes for location and date were used with all values set as the default (all values checked). For date, an ordered category scale was used to give preference to data from the 1990's. The websites associated with a given relationship are displayed in the window at the bottom right. As users mouse over (hover on) the topics, the bars on the data type list change dynamically to reflect the number of websites associated with that topic and data type. Clicking on a topic freezes the bars and listed websites and allows continual exploration through mouseovers. In the figure, the user has clicked on demographics and economics and the resulting number of data types available are shown in the data type list and the websites in the website window.

Usability Study.   To assess RB version 1 in 1999, a usability test was conducted with nine subjects on the version of the interface shown in figure 1. The main goal was to determine how easy the relation browser was to use and how well it provided an overview of the Fedstats portal and associated sites. Because exploration is difficult to assess empirically, a set of search tasks for the existing site map and the RB were devised to give test subjects some exploration guidance. In hindsight, this decision clearly biased users toward using the RB as a search tool rather than as an overview explorer. Testing took place in the Bureau of Labor Statistics Usability Laboratory in Washington, DC. The lab facility has two video cameras, one mounted in the ceiling to capture keyboard and mouse activity, and one mounted in the wall to capture users' facial expressions. There is a microphone to capture audio, and a scan converter to record the computer screen. All four signals are combined through a video mixer to produce a three window videotape of each session.

To focus on the site map and relation browser features, subjects with WWW experience were recruited. Six subjects were selected from a subject pool developed through a newspaper ad. Each subject from this group was paid $25 to participate in the study. The other three subjects

were employees of the Bureau of Labor Statistics who were unpaid volunteers. Six subjects were male, three were female, and all reported using the WWW at least once a week.

The results of the initial usability study are detailed in Marchionini, et al., 2000. Subjects had mixed success in using the Fedstats site map and RB to complete the six search tasks. One subject successfully completed all six tasks, and two subjects successfully completed no tasks. Subjects had equal success in the tasks using the Fedstats site map and the RB. Domain knowledge was clearly a factor in finding information. One subject who was a medical student was able to find the health care cost value quickly, and noted in the debriefing that navigation was skillful "because I was knowledgeable about terms." Another subject who worked with federal statistics regularly did not know that the CPI is produced by the Labor Department, but was able to use highly specialized knowledge about educational data to find education costs (searching for data on a specific city, knowing that the result would be compared to the national average).

RB Version 2
Based on the user testing, a number of changes were made to the prototype. Although subjects said they liked the filters, few were able to really take advantage of them in doing their search tasks--they liked the idea of filters for searching but the RB is meant to give an overview of the site and become a launching point for search as a secondary goal. In the revision, the filters were treated as attribute sets just as the data type concept. A tab mechanism that allowed users to relate topics to either data types, regions, or time was added to better use screen real estate. In addition, a few other minor revisions were made, resulting in the interface shown in Figure 2.

Field Test. After the revision of the prototype, a link to the tool was added to the Fedstats portal. Because the RB is a Java applet, only web browser versions that support Java could access the tool and it took from a few seconds to many seconds to download the applet and accompanying data depending on network speed. In order to get public feedback for the tool, an online questionnaire was created and Office of Management and Budget (OMB) clearance to use the questionnaire was sought. The questionnaire aimed to determine whether the RB provided users with an overview of the available federal statistics, gather general impressions about the usefulness of the tool, and elicit information about the settings in which users were accessing the site. The questionnaire contained 10 Likert-scaled questions, one checklist question, and two open-ended questions. Detailed results are discussed in Marchionini (2000).

Transaction logs from the web server hosting the RB tool provided additional details for our analysis. A total of 34956 requests were logged from November 12, 1999 to May 3, 2000. Because the logs consist of the applet request and three associated file requests but no records of what users actually examined while they used the applet or what sites they eventually clicked on, no attempts to do session analyses were made. One indicator of usage is that 7697 unique IP numbers were included in the requests. Because Internet Service Providers (ISPs) dynamically assign IP addresses (and individuals may use different machines), we cannot conclude that there were 7697 unique users of the RB over this time, but our approximation is that the RB was used by perhaps 7000 different users.

A total of 74 responses to the questionnaire were received during the field trial. Using the 7697 unique IP addresses as a conservative base, this represents less than a 1% response rate. These respondents voluntarily chose to complete the questionnaire and had already chosen to use an experimental/optional tool from the Fedstats site, so they cannot be considered a statistically representative sample. For the purposes of judging the potential usefulness and usability of a new interface technique, they served this research well by giving general impressions and specific reactions to the design and possible deployment.
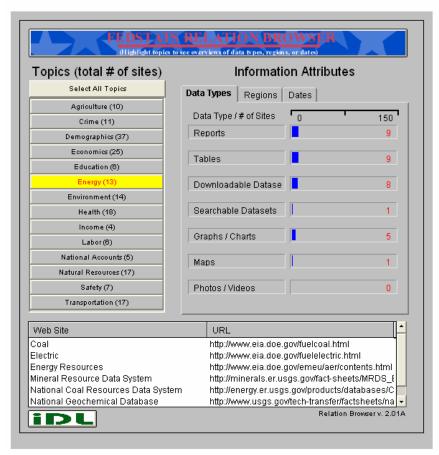
Figure 2. Relation Browser Version 2.

The field test online questionnaire yielded valuable feedback on the efficacy of the Relation Browser. Responses ranged from "This thing totally stinks," to various suggestions for improving the interface, to "I wouldn't change it at all, but please make it a permanent fixture [of the Fedstats site]." Overall, the responses and commentary were positive and lent support for the RB as an alternative tool for the Fedstats site. The most common complaint was with the amount of time the applet takes to load on low-bandwidth connections. The design trade-off involving the pre-caching of data to support the fast mouseover functionality is at the root of the problem. We continue to seek an effective compromise between front-loading system overhead that results in relatively long download times and the highly desirable response of the interface during runtime that requires local data.

The experience of mounting and testing the site within the development group and in the Fedstats group was itself valuable as we learned how to define goals, explain our mission, and reflect on the possibilities for the tool and what would be necessary to deploy it as a permanent addition to the Fedstats interface repertoire. For example, to implement the prototype, the developers hand indexed 200 websites in the Fedstats portal according to the RB architectural scheme. An indexing template was developed to help the webmasters for those sites to index their sites fully and email the results to a database that would serve as the central repository for the RB. Actually

assigning webmasters in more than 70 different federal agencies to this task requires managerial coordination that has yet to transpire.

RB Version 3
While the field test was underway over a six month period, the RB was revised based upon comments and suggestions as they came in. The main changes in the third version were the addition of a mouseover help button to explain how to use the RB, some minor changes in font sizes, and a change in the results (preview) pane to show which attributes are associated with each website in the results. Additionally, the results pane was made dynamic; when users hovered over the results list the presence or absence of the various attribute values was displayed. See Figure 3 for the RB Version 3 display. In this example, the user selected the 'energy' topic attribute and then hovered the mouse over the results display, showing, for example that the 'natural gas' website has reports, tables, and downladable datasets. This version was used to conduct another laboratory user study.
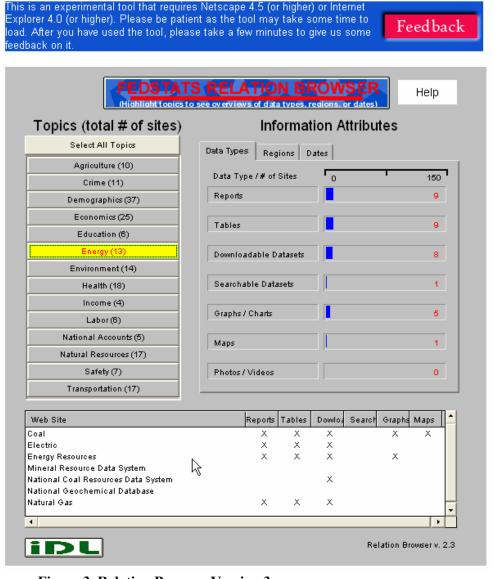

Figure 3. Relation Browser Version 3

Usability Study. Nine more subjects were recruited to participate in the second phase of the usability study. This time, five female and four male subjects were recruited. All were experienced professionals: four were librarians, three were news analysts/journalists, and two were research analysts. These subjects all had substantial WWW experience and were paid a $25 stipend for participation. Six of the subjects reported regularly using BLS and other statistical sites while the other three reported occasional usage of BLS and other statistical sites. Only one subject reported using Fedstats, three had looked at Fedstats but did not use it, and five had never looked at or used Fedstats before the testing.

The same testing protocol as in the previous year's study was used with minor changes to reflect the revised tool (e.g., since there were no filters, no questions were asked about filters; since a help function was added, reactions to the help were sought, and since attribute values for specific websites were added in the results windows, reactions to this addition were sought). The same six tasks were used and subjects did three with the Fedstats sitemap and three with the RB. Sessions lasted 40-60 minutes and the same testing protocols were followed.

Detailed results of the second usability test can be found in Marchionini (2000). Participant reactions to the Relation Browser ran the gambit from delight to confusion but were overall quite positive about the possibilities of using it in their work. Several noted that it would be helpful in one or more ways. They saw the possibilities for gaining overviews of federal statistical sites from several perspectives. First, some noted that the tool set up a process for understanding what is available. One noted that "it funnels you to information." Another said: "Get good jumping off point to broad overview of categories that are available. I can see what kind of websites are available." Second, one participant said it revealed relationships among the data: "To enable the searcher to understand the available websites on different subjects and to connect—to see the correlations between different subjects and relevant government agencies. To give the searcher understanding of the regions covered, types of data and time periods covered." Third, one participant noted that more information was provided: "This allows you to drill down into topics and go in to see data type, dates, etc. Gives you a lot more information, more robust." Fourth, half the participants said something about the forms of information (attributes) as beneficial features of the RB. "This is very good for getting an overview…it tells you what is included. As a researcher, I'd like this because it tells me how the data are available, plus this gives the URLs…this is a much more direct way." Another said: "The fact that a lot of this information is coming from a lot of different websites…its pulling from a lot of different sites…the way this is organized is really phenomenal…it's breaking it down by the way people really look for this information. This really helps me zero in on topics…I have a much bigger overview of what is available, I'm not just blindly going down this road." Another said: "Well, I like how it gives data date and region." These participants saw several ways that the RB can provide an overview of data.

Although these comments illustrate excellent promise for the RB, half the participants made some comment about needing some time to get used to using it. One participant suggested that reporters without Internet experience would have difficulty, noting: "Oriented more to the expert." Three other participants noted: "I like it…but I'm not sure it would be used a lot by the novice." "[A] little more explanation, help." "Useful but hard for me…tricky….If I used it a lot I'd find it very useful….If I was required to find statistical data one or two times a day I'm sure I'd use it."

There were several specific suggestions for improvement. One participant complained about the terminology: "Things that aren't clear like 'information attributes' has more to do with personality like nice hair." Two subjects questioned the value of the graphical bars showing how many web sites had specific data types (redundant since the numeric values were there). One

noted that the fonts were hard to read. Two respondents were bothered by the amount of screen activity. One said: "There is so much happening…when roll the mouse over, there is so much going on, roll here, something happens, go there, something else happens. Just too much going on."

In sum, participants liked the RB but eight of the nine subjects noted that the site map was more familiar and easier to use. Subjects preferred different tools for different tasks: the RB was preferred as an *overview tool* and the site map was preferred for *finding specific information*. They noted that the RB gave more information, most commonly citing the data type, region, and date attributes and the URLs as examples of information given by the RB but not the site map. These results support the use of the RB as an overview tool that could be used as an adjunct to traditional tools such as the site map, A-Z list, agency list and keyword search tools. The additional information provided by the RB gives people new views of federal statistics but these new views come with associated costs of processing and understanding. When understood as an alternative to existing tools, rather than a replacement, the RB was a welcome addition for many of the subjects.

GENERALIZING THE RB
While the BLS work on the Fedstats portal progressed, two types of generalization were undertaken. First, the underlying data architecture was changed from a simple flat file to a database scheme. The Java applet was redesigned to utilize a database backend in anticipation of further efforts to generalize the application. The new applet utilized the more powerful, lightweight, and flexible Jave SWING libraries. The revised applet allowed the information architect to change titles, colors, and labels as parameters in the applet tag that called the applet. This version of the RB was called the Generalized Relation Browser (GRB) because it aims to be usable to many other types of data than federal statistics. Second, students in HCI seminars over a three year period applied the RB to a variety of corpuses.

Fall 2000 course using RB version 4 (Generalized Relation Browser). In the summer of 2000, the Java code was rewritten to use Swing classes and JDBC to integrate a MySQL database. Students in an HCI course were introduced to the early versions of the RB and its applications at BLS. They were asked to come up with their own data sets and apply the RB to them. This was essentially an information architecture assignment in which students choose a collection of data, a small number of attribute sets and attributes for those sets, and mapped these data onto a flat file sequenced according to the the RB 3-table MySQL scheme. Once the data was properly formatted, the file was passed to the developer who then mapped it to the RB scheme and created the applet tags to link it to the RB. Thus, students focused on information architecture rather than MySQL, Java, JavaScript, and HTML. Example applications that semester include:
- Online international newspapers by country with browsable relationships by topic, coverage, and data type http://squash.ils.unc.edu/grb/spinks.html
- Stock imagery companies by category (rights, types) with browsable relationships by data type and language http://squash.ils.unc.edu/grb/smith.html
- Night spots in the Research Triangle by city with browsable relationships by types of drinks, entertainment, and music http://squash.ils.unc.edu/grb/zac.html
- Art libraries by region with browsable relationships by organization type and membership type http://squash.ils.unc.edu/grb/hardin.html
- An urban planning library website links by library location with browsable relationships by topic location, institutional affiliation and areas/issues http://squash.ils.unc.edu/grb/gus.html

- A journalism slides collection by topic with browsable relationships by dates, topics, and events http://squash.ils.unc.edu/grb/denn.html
- A historical advertising collection by topic with browsable relationships by year, special features (e.g., cartoon, sports), target audience and publication name http://squash.ils.unc.edu/grb/dengz.html.

These examples demonstrated the range of problems amenable to the RB and revealed many of the challenges of organizing information from a general database scheme for visual exploration. For example, the RB's dynamic effectiveness diminishes dramatically when users have to scroll. Screen real estate thus limits the number of attribute values feasible for an attribute set. Attribute sets such as 'author' for a collection are problematic as there are many more than 10 or so names that will fit into the RB primary attribute set or secondary attribute set panes. Students had considerable difficulty mapping their data onto the required file structure that could be imported into the MySQL database and this illustrated the need for finding ways to help information architects more easily map their organizations to the RB undlerlying data structure.

Fall 2001 course using RB version 5 (RAVE).
During the 2001-02 academic year, Brunk worked with the UNC-CH Digital Library team to apply the RB to the campus digital library. The strong limitations of number of attribute values led to a major redesign that separated the results (preview) pane from the attribute value panes. Results are displayed in a resizable, separate dialog box that overlays the exploration panes. A second major change was to add a third level of attribute sets that provided ways to scale the interface to larger collections by giving another dimension to partition on. The third set of attribute sets was made available as a pull down list. A new backend database scheme was also created so that the database (consisting of 9 tables) and the RB could be served from different physical servers. This new version was named the Relation and Attribute ViEwer (RAVE). See figures 4a and 4b for the new version. In the Fall 2001 offering of the HCI seminar, students were given the same assignment as in 2000 but were asked to present their data using Oracle or Access. They also had some control over the colors used and the pane titles and descriptions. Examples from this class include (all of these examples are available from http://squash.ils.unc.edu/inls357/ :
- E-books by format with browsable relationships by genre, delivery, and publisher
- Bridal stores by region (only one region was implemented) with browsable relationships by merchandise, city, and user rating
- Art museums by city with browsable relationships by culture, visitor information, and other specifics
- Accessibility regulations and guides by type (guideline, legislation, policy) with browsable relationships by format
- Electronic library resources by indexes and databases (a second attribute set, e-journals was laid out in the interface but not populated in the database) with browsable relationships by full-text and off campus access
- A radiology test construction database by diagnosis (cancer or benign) with browsable relationships by lesion type and difficulty of diagnosis

As in the previous semester, students demonstrated a wide range of applications. The greatest challenges were mapping their data onto the RAVE database scheme, which had nine tables. Some students created Access databases with the exact scheme and then exported to Oracle, some used SQL commands directly, and most used an application tool (Toad) to enter data via a GUI rather than using SQL commands. Some students focused on actual applications with large data sets (e.g., the library resources and accessibility regulations and guidelines).

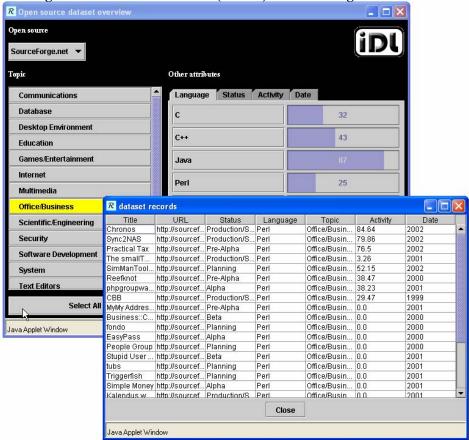**Figure 4a. Relation Browser (RAVE) after clicking on Office/Business**

Figure 4b. Relation Browser (Rave) with results after clicking on Perl

In addition to these examples, the RAVE interface was applied to a large patent database for carbon dioxide as part of the Science and Technology Center for Environmentally Responsible Solvents and Processes at UNC-CH. In this application, 4000 patents are organized by applications (pharmaceuticals, textiles, microelectronics, polymers, etc.) with browsable relationships by date and assignee types. These attribute value sets, the attribute values in each set and overall design were discussed over time with chemists and patent attorneys to identify the best ways to slice and explore relationships in the CO_2 patents. The results pane leads to the full text of the patents in the main Oracle database, which are also full-text searchable.

Fall 2002 course using RB version 6 (RAVE 5.01)
Over the summer of 2002, Brunk and Meichun Li re-designed the underlying database scheme (consisting of 12 tables) and created an Access macro that leads information architects through the database scheme creation process. The data can then be easily entered by hand through the tool. For large existing data sets, custom scripts must be written to map the data to the RB scheme. In the Fall 2002 version of the HCI seminar, five new applications were created by students that aimed to test the latest version with the goals of determining how to streamline and automate piping data from legacy applications to RB and how it scales up to larger data sets. The five examples are:

- Almost 10,000 UNC Library films and videos collection by genre (19 genres) with browsable relationships by decade and format (e.g., NTSC, DVD, etc.). This data was exported from the library database and mapped to the RB scheme with Perl scripts
  http://squash.ils.unc.edu/rave/sheat.html
- Almost 10,000 programs from the SourceForge repository by topic (14 topics such as communication, database, education) with browsable relationships by language (C++, Java, etc), status (planning, pre-alpha, alpha, beta, stable, mature), activity (number of downloads), and date (of contribution). This data was obtained by a Java program that crawled the repository and piped the data to the RB scheme.
  http://squash.ils.unc.edu/rave/junliang.html
- A subset of The Energy Information Administration website by EIA category (cost, consumption, production, US states, Census divisions, and international) by fuel (petroleum, natural gas, coal, nuclear, etc.) with browsable relationships by geography and sector. This example focused on different ways to slice and dice the massive amounts of data in the EIA site. This particular scheme is one of several under investigation for an ongoing research project (Government Statistics), thus only a few data elements are actually accessible (a few web pages at EIA).
  http://squash.ils.unc.edu/rave/suc.html
- More than 100 knots by type (bends, bindings, hitches, etc.) with browsable relationships by name (a set of 5 alphabetical ranges A-F, G-L, etc.), use (5 categories), difficulty (3 categories), number of ropes (1-4), and number of steps (1-14 in groups of two). This implementation was based on a knot book and demonstrates a way to handle many attribute values in an attribute set and several different attribute sets.
  http://squash.ils.unc.edu/rave/ahughes.html
- North Carolina tertiary institutions by colleges/universities and community colleges (community colleges were not populated in the example) by size (4 categories of student population) with browsable relationships by location (4 regions in the state), tuition (5 categories), affiliation (7 denominations), and degrees (5 categories). The particular challenge in this example was gathering the data for more than 50 actual institutions as this particular combination of data is not available in any single aggregate form.
  http://squash.ils.unc.edu/rave/jld.html

These examples again show a range of applications but more importantly demonstrate the scalability of the RB. This latest experience also shows the possibility of automating much of the mapping between existing data sets and the RB. In the case of existing digital libraries or large databases, it is feasible to follow the following steps:

- plan a set of attribute sets and relationships of interest
- create the RB scheme for this architecture
- crawl (or query) the repository to get the data for the attribute values
- pipe the resulting data into the RB scheme
- embed the RB database in a web page using an applet tag

Clearly, determining ways to slice the data to support useful exploration is the most conceptual step and defines a large part of what information architects do.

In the case of creating a new data set, the data collection itself becomes the significant task. The experiences with these classes over three years demonstrates several useful points. First, it demonstrates the viability of the RB in terms of range of applications, scalability, and usefulness by information architects who can focus on data collection and organization rather than interface programming. Second, it demonstrates a kind of participatory design where people with real data needs (many of the applications were related to students' jobs or personal interests) participate in the ongoing development and testing of a software suite. Third, it demonstrates a pedagogical perspective of actively engaging in a real UI development project. Although there were many problems students encountered in working with unpolished software and a changing data structure, every student learned about the many compromises made between an idealized prototype that demonstrates an interaction concept and actually implementing a working application.

STATUS AND FUTURE WORK

The RB code for RAVE is downloadable from the RB website. We are currently applying the concept to government statistics portals and websites and initiating a major redesign. Our studies and experiences with non-expert users of statistical web resources (e.g., Hert et al) demonstrate that simplicity is a key element of UIs as default. Our current project to define a statistical knowledge network that integrates across agencies and across different levels of government will require substantial support for exploration and explanation/help. Our approach is to develop multiple levels of UI and several kinds of online help that assist people in transitioning across these UIs. The RB will be developed in a basic and pro versions, taking advantage of a common underlying data structure. We are working with our partners at the Human Computer Interaction Laboratory at the University of Maryland on these efforts.

A second line of work is to continue to find ways to streamline the way that different information architects with different data and needs can easily adopt the RB. Our concept is to create an interface server that allows data providers/webmasters to easily point their data at the RB and provide their users with RB exploration capabilities. The easiest solution for us is for data providers to download a version of RB appropriate to their needs and implement it themselves. Another solution is to create a service where providers send their data to be mapped to a RB-type UI and an integrated package is provided. Our work aims in more general directions where UIs are requested on the fly. This could be through data providers redirecting end user requests to a RB server or a more ambitious scheme where providers publish their schemes (through XML or Open Archives Initiative protocols) allowing many different interface servers render their data. Thus would eventually allow end users to choose to render data from any website (e.g., commercial sites, digital libraries) using the RB and other interactive UIs according to their needs and preferences.

Conclusion. This paper describes the evolution of a general class of user interface that has been developed, tested, and revised over several years and in several different settings. The RB approach to helping people use coordinated lists to explore the relationships among attributes in an information space has been successful on many fronts. It helps people to get an overview of what is and is not in a website or other partition of webspace and how the concepts are organized. This in turn helps searching and understanding what is found. The simplicity of pair wise visual comparisons controlled by simple mouse movements without having to click and wait for a new page to load is perhaps its greatest appeal—in essence it is a simple interactive technique for directly exploring and manipulating information. The GB has been shown to be applicable to a wide range of data sets. Finally, these experiences lead to the possibility of using GB and other general-purpose UIs as the basis for a new type of WWW-service we call interface servers. Unlike environments like Visual Basic where programming is simplified but not eliminated, the RB environment trades off some generalizability for ease of implementing a data exploration UI. We believe that tools like the RB will help information architects and other designers and data providers focus on their data and ways to make it comprehendible to people and leave the UI details to HCI specialists and developers.

References

Brunk, B. (1999). Overview and Preview Tools for Navigating the World-Wide Web, *SILS Technical Report TR-1999-03*. July 31, 1999. Available http://ils.unc.edu/ils/research/reports/TR-1999-03.pdf (accessed Feb. 27, 2003)

Brunk, B. & Marchionini, G. (2001). Toward an Agile Views WWW Sitemap Kit: The Generalized Relation Browser, *SILS Technical Report TR-2001-06*. Available http://ils.unc.edu/ils/research/reports/relation_browser.pdf (accessed Feb. 27, 2003)

Brusilovsky, P. & Milosavljevic, M. (Eds.) (1998). Introduction to Special Issue on Adaptivity and User Modeling in Hypermedia Systems. *The New Review of Hypermedia and Multimedia*. 4, 1-260.

Bederson, B. & Hollan, J. 1994. Pad++: A zooming graphical interface for exploring alternative interface physics. In *Proceedings. of ACM UIST '94*. 17-26.

Dewan, P. (1990). A tour of the SUITE interface software. *ACM SIGGRAPH Conference* (Snowbird, Utah). 57-65.

Dodge, M. & Kitchin R. (2000). *Mapping Cyberspace*. New York, NY: Routledge.

Geisler, G. (2000). Enriched Links: A Framework For Improving Web Navigation Using Pop-Up Views. *SILS Technical Report TR-2000-02*. February, 2000. Available http://ils.unc.edu/ils/research/reports/TR-2000-02.pdf (Accessed Feb. 27, 2003)

Green, R. & Bean, C. (1995), Topical relevance relationships. 2. An exploratory study and preliminary typology. *Journal of the American Society for Information Science*; 46 (9) Oct 95, p.654-62

Greene, S., Marchionini, G., Plaisant, C., & Shneiderman, B. (2000). Previews and overviews in digital libraries: Designing surrogates to support visual information seeking. *Journal of the American Society for Information Science*., 51(4), 380-393.

Hearst, M. (1999). User interfaces and visualization. In R. Baeza-Yates and B. Ribeiro-Neto (Eds.) *Modern information retrieval*. NY: ACM Press.

Hert, C. Y Marchionini, G. (1998). Information Seeking Behavior on Statistical Web Sites: Theoretical and Design Implications. *Proceedings of the 61st Annual Meeting of The American Society for Information Science*. (Pittsburgh, PA, Oct. 25-29, 1998). 303-314.

Hutchins, E., Hollan, J., & Norman, D. (1986). Direct Manipulation Interfaces, In Donald Norman and Stephen Draper (Eds), *User Centered System Design*, pp. 87-124.

Kim, W. & Foley, J. (1990). DON: User interface presentation design assistant. *ACM SIGGRAPH Conference* (Snowbird, Utah). 10-20.

Lamping, J. & Rao, R. (1996). Visualizing large trees using the hyperbolic browser. *Proceedings of ACM CHI '96 Conference Companion* (Vancouver, April 13-18, 1996). New York: ACM Press. 388-9.

Lanning, T., Wittenburg, K., Heinriches, M., Fyock, C., & Li, G. (2000). Multidimensional information visualization through sliding rods. *Proceedings of AVI 2000*. (Palermo, Italy). 173-180.

Johnson-Laird, P. (1983). *Mental Models*. Harvard University Press, 1983

Marchionini, G. (2000). Overviews to Previews to Answers: Integrated Interfaces for Federal Statistics. *Final Report to Bureau of Labor Statistics, June 30, 2000*. Available http://www.ils.unc.edu/~march. (Accessed Feb. 27, 2003)

Marchionini, G., Brunk, B., Komlodi, A., Conrad, F., & Bosley, J. (2000a). Look before you click: A relation browser for Federal statistics websites. *Proceedings of the ASIS 2000 Annual meeting*. (Chicago, IL, November 15-18, 2000).

Marchionini, G., Geisler, G. & Brunk B. (2000b). Agileviews: A Human-Centered Framework for Interfaces to Information Spaces. *Proceedings of the ASIS 2000 Annual Meeting*. (Chicago, IL, November 15-18, 2000).

Marchionini, G., Nolet, V., Williams, H., Ding, W., Beale, J., Rose, A., Gordon, A., Enomoto, E., & Harbinson, L. (1997). Content+Connectivity=Community: Digital resources for a learning community. *Proceedings of ACM DL '97*. (Pittsburgh, PA, July 23-26, 1997), p 212-220.

Marchionini, G., Plaisant, C., & Komlodi, A. (1998). Interfaces and tools for the Library of Congress National Digital Library Program. *Information Processing & Management*. 34(5), 535-555.

Myers, B., McDaniel, R., & Kosbie, D. (1993). Marquise: Creating complete user interfaces by demonstration. *Proceedings of ACM InterCHI '93* (Amsterdam, The Netherlands, April 24-29, 1993). 293-300.

NASA. (1998). *NASA Home Page Sitemap*. Available http://www.nasa.gov/siteindex.html. (Accessed, Sept. 1998).

Nation, D. Plaisant, C., Marchionini, G., & Komlodi, A. (1997). Visualizing websites using a hierarchical table of contents browser: WebTOC. In *Proceedings of Designing for the Web: Practices and Reflections* (3rd Conference on Human factors and the Web, Denver, June 12, 1997). http://www.uswest.com/web-conference/proceedings/nation.html (Accessed September, 1999). [preprint available at ftp://ftp.cs.umd.edu/pub/hcil/Demos/WebTOC/Paper/WebTOC.html (Accessed Feb. 27, 2003)]

Nielsen, J. (1990). *Hypertext & Hypermedia*. San Diego, CA: Academic Press.

Plaisant, C., Marchionini, G., Bruns, T., Komlodi, A., & Campbell, L. (1997). Bringing treasures to the surface: Iterative design for the Library of Congress National Digital Library Program. *ACM CHI '97 Conference*. (Atlanta, March 22-27, 1997), p. 518-525.

Quillian, R. (1968) Semantic memory. In Minsky, (Ed.), *Semantic Information Processing*, Cambridge, MA: MIT Press, 216—270.

Robertson, G., Czerwinski, M., Larson, K., Robbins, D., Thiel, D., Y Van Dantzich, M. (1998). Data Mountain: Using Spatial Memory for Document Management. *Proceedings of the 11th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, Nov. 1-4, 1998). 153-162.

Rosch, E. 1975. *Cognitive representations of semantic categories*. *Journal of Experimental Psychology: General* 104: 192-253

Tweedie, L., Spence, B., Williams, D., & Bhogal, R. (1994). The attribute explorer. *CHI94 Companion*, (Boston, MA, April 24-28, 1994). 435-6.

Wittenburg, K., Lanning, T., Heinriches, M., & Stanton, M. (2001). Parallel Bargrams for consumer-based information exploration and choice. *Proceedings of ACM UIST 01* (Orlando, FL, November 11-14, 2001). 51-60.