Who Votes For What?
A Visual Query Language for Opinion Data

Geoffrey M. Draper, *Member, IEEE*, and Richard F. Riesenfeld

Abstract— Surveys and opinion polls are extremely popular in the media, especially in the months preceding a general election. However, the available tools for analyzing poll results often require specialized training. Hence, data analysis remains out of reach for many casual computer users. Moreover, the visualizations used to communicate the results of surveys are typically limited to traditional statistical graphics like bar graphs and pie charts, both of which are fundamentally noninteractive. We present a simple interactive visualization that allows users to construct queries on large tabular data sets, and view the results in real time. The results of two separate user studies suggest that our interface lowers the learning curve for naive users, while still providing enough analytical power to discover interesting correlations in the data.

Index Terms—Visual query languages, radial visualization, data analysis, human-computer interaction.

1 Introduction

Opinion polls play an important role in quantitative political and marketing research efforts. Polls provide a practical mechanism for politicians and executives to gauge public interest in their platforms and products. While poll results are always a popular news item in the media, they are of special interest this year in the United States, due to the November 2008 presidential election.

Professional polling firms regularly release reports to the public that summarize the results of their recent surveys. These reports (see, for example, [13] and [30]) typically consist of a written analysis, perhaps a table of numbers, and an occasional graphic like a bar graph or pie chart.

Statistical graphics like these are convenient for end-users who simply want to view a pre-canned analysis, however, they offer rather little interactivity. Moreover, depending on the editorial predisposition of the polling agency, a report may emphasize one aspect of the data more than others. If users had access to the original data along with a usable analysis tool, they would be free to expore the data on their own and discover relations among a variety of variables. Yet for most people, this kind of interactive analysis is out of reach for at least two reasons. First, the data sets used by polling agencies are often proprietary [7]. Second, current data analysis tools generally have a prohibitively steep learning curve for casual users. Our research addresses the second of these two concerns. Indeed, finding ways to simplify data analysis, bringing it to "the masses" [33], is a subject of ongoing research in the human-computer interaction and information visualization communities.

This paper introduces a novel interactive visualization for querying and analyzing tabular demographic data. Although the visualization we propose may be applied to a variety of data types, to focus the present discussion we will restrict our examples to those relating to opinion polls. Many good visualizations exist for multivariate data [20, 43], but poll data creates a unique set of challenges. Among these considerations are:

- Analysts who work with poll data must often function under tight deadlines. On the night of an election, for example, polling organizations usually try to present their results as soon as the polls close so they can be first to predict the outcome of the election. This demands that an interface allow for the rapid and facile execution of many hypothesis formulation and evaluation cycles to identify both expected and unexpected trends in the data.
- Demographic data sets focus on the many, not the few. In some application domains, the principal objective is to "drill down" into a massive data set in search of a handful of salient entities. However, in an opinion poll, the goal is to uncover meaningful trends for broad segments of society. In this context, information on individual entities is uninteresting at best, and misleading at worst. Visualizations are needed that provide uncluttered summaries of large data sets, typically from thousands to millions of entities, with comparable clarity.
- The results of opinion polls are of interest to a wide range of people, not just a handful of specialists. Hence, a visualization for opinion poll data must support not only rapid querying of the data, but also an effective presentation of the query results that requires minimal explanation even for naive users.

The requirements listed above guided the design of our visualization. Our design goals were to create an integrated query interface that supports rapid exploration and "information foraging" [27] to focus on global trends in the data. Above all, we aimed for simplicity of use. We sought a design that would be easy to learn for naive users, while still providing sufficient power for many of the tasks involved in real data analysis.

For a number of reasons, our visualization employs a radial design in which the components of the user interface are arranged in a ring shape. First, this increases the accessibility of widgets by placing them equidistant from the center of the canvas [9]. Second, ring-based user interfaces are trivially delineated; an icon is either inside the ring, on the ring, or outside the ring. This reduces the number of "states" that a user has to remember.

Our interface is based on the direct manipulation metaphor, in which queries are constructed by drag and drop. Rather than navigate an external interface, queries are constructed directly within the visualization itself. The user can adjust the level of detail dynamically, viewing attributes in isolation or in comparison with others. Transitions from one query to the next are smoothly animated to preserve the user's sense of context [16, 44].

The main contributions of this paper are:

- a highly interactive canvas for querying multivariate data,
- an integrated radial visualization for displaying query results,
- results from two preliminary user studies suggesting that our method is an easy-to-learn metaphor for multivariate data analysis.

• *Geoffrey M. Draper and Richard F. Riesenfeld are with the University of Utah School of Computing, E-mail: {draperg , rfr}@cs.utah.edu.*

Manuscript received 31 March 2008; accepted 1 August 2008; posted online 19 October 2008; mailed on 13 October 2008.
For information on obtaining reprints of this article, please send e-mailto:tvcg@computer.org.

We organize the remainder of this paper by, first, providing some background information on tabular data in general and opinion polls in particular. Then we review some of the relevant related research in visual query interfaces and radial visualization. We describe our visualization design in detail and present a few details about our prototype implementation. After walking through a typical usage scenario, we then discuss the results of two preliminary evaluations of the prototype by both expert and novice users. Finally, our conclusions and suggestions for future work are presented.

2 Definitions

To facilitate the present discussion, we recall a few key terms [34] from the field of database design.

- A tabular data set consists of several distinct *entities*.
- Each entity possesses a number of *attributes*.
- Each attribute, in turn, may assume any one of several *values*.
- The set of possible values for each attribute is called its *domain*.

Each entity has the same set of attributes, but the values assumed by each attribute may vary widely from entity to entity.

In an opinion poll, each *entity* corresponds to a person who responded to the poll. The *attributes* are the questions on the poll, and the *domain* is the set of allowable answers (*values*) to the poll questions. In many data sets, like an opinion poll, the values for each attribute are *mutually exclusive*; a respondent may select at most one answer for any given question. Moreover, while a tabular data set may consist of any number of entities, a scientifically-conducted opinion poll usually includes hundreds or thousands of respondents.

3 Related work

This research combines aspects of two recurring themes in information visualization, namely, visual query languages and radial user interfaces.

3.1 Database query visualization

Making databases easier to use has been a subject of research for several decades. One of the first such efforts that is still in widespread use is Query By Example [45]. Cammarano et al. [3] make the observation that most user interfaces for databases take either one of two approaches. Either the user interface aids in formulating the query, or in visualizing the results.

Sinha and Karger [35] propose a system for aiding in navigation of semistructured data sets by suggesting navigation hints to the user. Goldman and Widom [14] propose a method for exploiting similarities among pages in the same website to perform more effective queries. Polyviou et al. [29] describe an interface for performing database queries based on the ubiquitous filesystem browser interface. The VisTrails system [32] makes use of provenance data to maintain a history of past queries for creating visualizations.

In the second category, there are many systems which offer a direct-manipulation interface for browsing the results of a query. Furnas and Rauch [11] as well as Stonebraker [39] present canvas-based visualizations that support zooming and panning. Visage [31] by Roth et al. is a highly interactive direct manipulation system that uses a variety of graphing techniques to communicate results to the user. ManyEyes [42] and Swivel [40] are websites for collaborative visualization. While not necessarily interactive, parallel coordinates [20] and scatterplots [43] also support the simultaneous visualization of many variables.

In contrast, the Polaris system [38] incorporates both a novel query interface mechanism and an integrated visualization. Based on the well-known metaphor of a pivot table in a spreadsheet, Polaris allows users to view correlations in the data with respect to any particular attribute in the data set.

Our work differs from existing systems primarily in our strict focus on ease of use and interactivity. To our knowledge, this is also the first visual query language which uses the radial layout metaphor for both querying and visualization. We review related work in radial user interfaces in the next section.

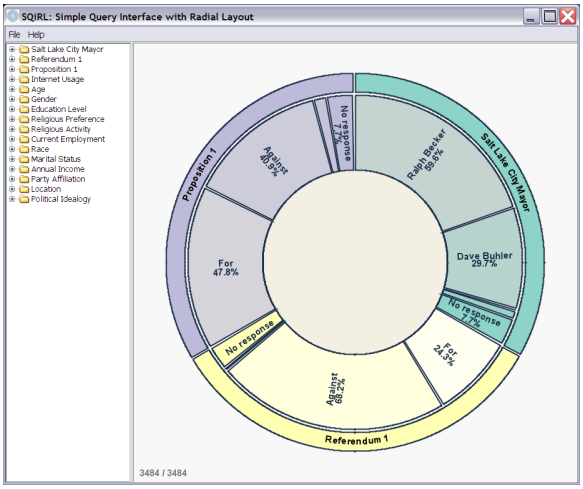

Fig. 1. Startup screenshot of our prototype system. By default, attributes relating to voter opinion appear on the ring, with no attributes or values restricting the size of the sample population.

3.2 Radial user interfaces

An increasingly popular metaphor in contemporary information visualization, radial charting techniques nonetheless have a tradition spanning back hundreds of years. Pre-digital examples of radial information layouts include William Playfair's 1801 invention of the pie chart [28, 36], and Florence Nightingale's rose diagrams [25] for communicating sanitary conditions in British military hospitals during the Crimean War. In the mid-twentieth century, Northway used radial diagrams to track the social behaviors of gradeschool children [26].

Much of the recent work in radial user interfaces traces its lineage to research in graph layout algorithms for computer graphics [17]. These algorithms, in turn, have inspired techniques for visualizing multivariate data. Many such designs involve positioning data points as nodes on the spokes on a wheel [15, 18]. In these visualizations, the center point of the canvas holds some semantic meaning, and the distance of each node from the center shows a relationship relative to it. A recent example of this is the DataRose [6].

In contrast, a second variety of radial visualization (called *radial space filling* or RSF [37]), the data points are typically arranged in compact concentric rings [23], and rendered so as to form a circle. Each ring represents a different attribute of the data. Examples include polar treemaps [21], fan charts [5], and Radial Traffic Analyzer [22].

Another general category of radial visualization arranges the data points around the circumference of a ring, while reserving the interior of the ring for other data. Correlations among data points are often rendered as lines between nodes on the circumference and nodes in the interior. Examples include Daisy [4], NetMap [12] and VisAlert [24].

Our method is most closely related to the latter category. Our approach differs from previous work in that we reserve the interior of the ring for constructing user-specified queries rather than for rendering line segments between related entities. Instead, correlations in the data are displayed in a series of curved bar charts on the ring's circumference.

4 Visualization and interaction

4.1 User interface

The visualization canvas occupies the main portion of the user interface (Fig. 1). To the left of the canvas is a side panel containing a two-level tree structure of attributes and possible values. These can be dragged into and out of the canvas as needed.

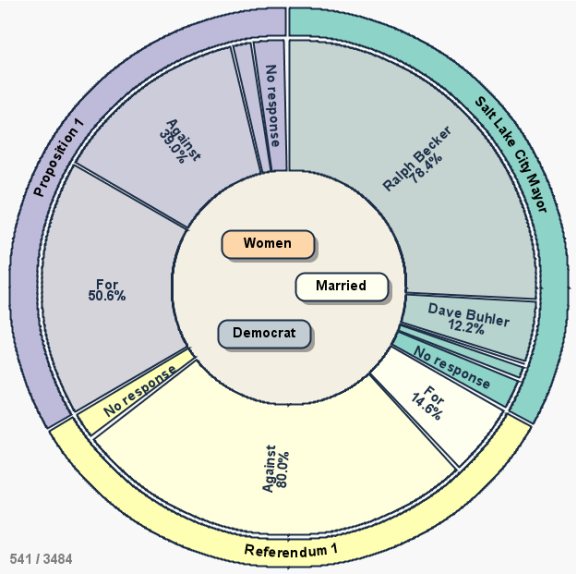

Fig. 2. The current population is specified by dragging values for attributes into the ring's interior. The percentages in the sectors indicate the breakdown of the population by attribute. The size of the population specified by the current query, relative to the total population, is shown in the lower left corner of the canvas.

The canvas itself is a workspace upon which icons representing attributes and values can be manipulated. The most prominent feature of the canvas is the large ring in the center. We divide the canvas into three distinct regions: the ring itself, the area inside the ring, and the area outside the ring.

- **Ring.** The ring is divided up into multiple equiangular sectors, each representing one of the attributes in the data set. Each sector is partitioned into two concentric layers. The outer layer of the sector displays the name of the attribute. The inner layer contains a sequence of subsectors that create a (curved) stacked bar chart [43]. Each subsector is scaled in direct proportion to the number of entities exhibiting that attribute value, and displays the corresponding name and percentage. By default, the percentages are computed relative to the entire population, unless the population is restricted by placing values into the interior of the ring.
- **Interior.** By dragging specific values for an attribute into the ring's interior, the user can selectively refine the population on which the displayed percentages are based. For example, if the value *Democrat* from the attribute *Party Affiliation* is placed inside the ring, then the results displayed on the ring will reflect only those entities whose party affiliation is listed as Democratic.
- **Exterior.** The area outside the ring simply serves as a working storage area, or cache, for attributes and values that are not currently part of any query, but which recently were or soon may be. Attributes and values that are not likely to be needed soon can optionally be dragged to the side panel to clear up space on the canvas.

The ring itself shares some similarities to a multi-series donut chart [10]. With this visualization, however, the interior of the ring is more than a decoration; it serves as a query workspace. If desired, the user can change the size of the ring's interior and exterior radii by selecting and dragging the ring's borders.

In an effort to reduce "visual clutter" in the display, text labels are used only when it makes sense. When space permits, each subsector on the ring displays the name and percentage of the attribute value corresponding to it. When only a small fraction of the population manifests a particular value for a given attribute, the corresponding subsector may be too small to accommodate displaying both the name and percentage labels. In that case, only the name is presented. If the subsector is too small even for that, then no text is rendered in that subsector. In any event, "mousing over" any component of the canvas shows a tooltip revealing details about the underlying attribute. Alternatively, the canvas can be zoomed and panned to show more detail. Thus, textual information about small areas of the visualization is available on demand, but does not otherwise crowd the display.

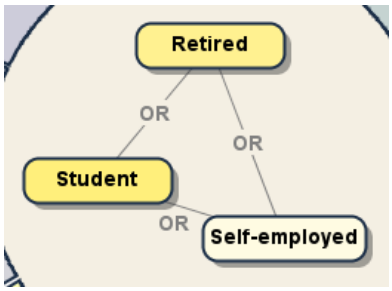

Fig. 3. In general, values in the interior of the ring are ANDed together to form the query string. Multiple values from the same attribute, on the other hand, are ORed instead. In the visualization, this is indicated by rendering a thin labeled edge between the appropriate icons.

Within the exterior visualization area, attributes are represented as drop-down combo boxes. When expanded, an attribute box reveals a menu of possible values, which the user can then select and drag inside or outside the ring.

For each value inside the ring, the sample population is refined by ANDing the values together. For example, if the values *Democrat*, *Women*, and *Married* (from the attributes *Party Affiliation*, *Gender*, and *Marital Status*, respectively) were placed in the ring's interior, then the percentages displayed on the ring would reflect the subset of the total population that fits the description of married women who are Democrats (Fig. 2). There is no theoretical limit to the number of values that can be placed within the ring's interior. However, in practice, adding too many constraints may limit the sample population to so few entities that the results are no longer statistically significant. Accordingly, our prototype system displays the size of the current query population in the lower left corner of the visualization canvas.

The user can also drag multiple values for the same attribute into the interior of the ring. However, as noted in Section 2, the possible values for each attribute are mutually exclusive. For example, a person's party affiliation may be either Democratic or Republican, but not both simultaneously. Hence, performing a query to find all entities that are both Democratic AND Republican would yield a population of zero. Therefore, queries involving multiple values from the same attribute use an implied OR operator instead of AND. In the visualization, this is shown by rendering a thin line between the icons as a visual reminder that they belong to the same attribute (Fig. 3).

Each time a value is dragged into or out of the ring's interior, the sectors on the ring are updated to reflect the values for the population specified in the current query. The transition from the previous to the current query's results is smoothly animated to help the user maintain a sense of context [44]. The queries take place in real time, enabling interactive data exploration and rapid testing of hypotheses.

4.2 Multifaceted data exploration

If taken in isolation, the method described above would make this interface merely a nice widget for analyzing the demographics of people who vote in certain ways. However, our technique goes one step further. The user can see percentage information about *any* attribute in the data set, not just those related to respondents' opinions. Consistent with the other aspects of our user interface, this is done by direct manipulation. The user can drag any attribute from the tree in the side panel, or from the area outside the ring, and drop it onto the ring itself

(Fig. 4). The visualization immediately updates to show the breakdown of the survey population with respect to that attribute. Likewise, any attribute on the ring's circumference may be dragged to the exterior, therefore excluding it from consideration. This ability to view the data from the perspective of any attribute permits the user to find relationships or spot trends that the compilers of the data may not have envisioned. In some ways, this behavior is reminiscent of a pivot table in a spreadsheet [10], albeit with an arguably simpler interface.

In effect, what we have described is a direct manipulation interface for specifying two complementary kinds of queries.

1. By dragging an attribute value into (or out of) the ring's interior, the user restricts the query to a subset of the total population that matches certain characteristics.

2. By dragging an attribute onto (or off of) the ring's circumference, the user specifies the attributes for which quantitative information is desired, regarding the given population.

Stated another way, the values inside the ring could be considered independent variables in that they determine the results of the dependent variables on the ring's circumference.

4.3 SQL and geometry generation

Each time the user adds or removes an entity from the ring, the system issues a batch of SQL queries to the underlying database. The results of these queries determine the size of the sectors.

4.3.1 SQL generation

The SQL statements are of the form:

```
SELECT COUNT(*) FROM T WHERE Q1 AND Q2
```

where *T* is the name of the database table, and *Q1* and *Q2* are lists of conditionals of the form *attribute=value*.

Q1 is derived from the values in the ring's interior. The conditionals in *Q1* are ANDed or ORed together as appropriate (see Section 4.1). For example, if the values *Married*, *Student*, and *Retired* were inside the ring, *Q1* would be:

```
maritalStatus=married AND (employment=student
   OR employment=retired)
```

Q2 is determined by the attribute/value pairs on the ring itself. Whereas *Q1* is constant for each query in a given batch, *Q2* is unique for each attribute/value pair. For example, if the attribute *Party Affiliation* were on the circumference, *Q2* would assume a different value for each possible value of that attribute:

$\mathbf{Q2_1}$ **:=** `partyAffiliation=republican`

$\mathbf{Q2_2}$ **:=** `partyAffiliation=independent`

$\mathbf{Q2_3}$ **:=** `partyAffiliation=democrat`

Thus, the full SQL syntax for a typical query, using a table name of "voters" and the values for *Q1* and $Q2_1$ as outlined above, is expressed as:

```
SELECT COUNT(*) FROM voters WHERE
   maritalStatus=married AND
   (employment=student OR employment=retired)
   AND partyAffiliation=republican
```

Similar queries are issued for each attribute/value pair.

Fig. 4. (a) Any attribute can be added to the ring by dragging its icon onto the circumference. (b) The sectors on the ring resize to accommodate the newly-added attribute. (c) Any attribute on the ring can be dragged outside the ring to remove it from the current query. (d) As before, the sectors resize to fill the available space. (Not shown: the resizing of the sectors is smoothly animated so the user maintains context during the transition.)

4.3.2 Geometry

Each sector of the ring corresponds to a single attribute in the data set, and is allocated an equal proportion of the ring's circumference. Thus, if A is the number of attributes currently on the ring, then each attribute is allocated $\frac{2\pi}{A}$ radians. As noted previously, the inner layer of each sector is a stacked bar chart showing the percentage of the survey population which matches both that value for the attribute and those inside the ring. We now describe the algorithm for tabulating these queries and sizing each subsector of the stacked bar chart. First, let:

$$S = \{\text{all values currently inside the ring}\} \tag{1}$$

We define a function $E(S)$ that yields the number of entities in the population whose attributes match the values in S.

The domain of each attribute is variable-sized, dependent on the schema of the data set. Let $V(a_i)$ be the number of values in attribute a_i's domain. Thus, whenever a value is added to or removed from either the ring or its interior, the system issues N separate queries to the underlying database, one query for each value of each attribute currently on the ring. N is computed thus:

$$N = \sum_{i=1}^{A} V(a_i) \tag{2}$$

Furthermore, each subsector of an attribute's stacked bar chart is functionally identified by the attribute and value corresponding to it. We write this as a pair (a_i, v_n) where a_i is an attribute and v_n is a value within that attribute. Hence, each of the N queries is uniquely identified by the pair $(S, (a_i, v_n))$.

Let Q be a function that returns the number of entities in the data set that match the tuple $(S, (a_i, v_n))$. This number is then divided by $E(S)$, to give a fractional value $P(S, (a_i, v_n))$ between 0 and 1.

$$P(S, (a_i, v_n)) = \frac{Q(S, (a_i, v_n))}{E(S)} \tag{3}$$

Let θ_{v_n} represent the radians allocated to a value v_n within attribute a_i's stacked bar chart.

$$\theta_{v_n} = \left(\frac{2\pi}{A}\right) P(S, (a_i, v_n)) \tag{4}$$

This computation takes place for each of the N queries, each time the user drags a value onto, inside, or outside the ring. The geometry for each bar chart is reconstructed based on the radial values calculated above. The animated transitions between queries are implemented in a straightforward manner, by linearly interpolating the radial magnitude of each sector for each key frame [1].

The SQL generation, query execution, and geometric computations take place automatically, behind the scenes. All the user sees is that the sectors "magically" resize according to the terms he or she dictates by dragging icons around the canvas.

4.4 Observations on scalability

Our visual design was inspired by the growing popularity of radial user interfaces in information visualization and visual analytics. Our visualization differs from previous radial layouts in at least two respects, both of which increase the scalability of our technique.

First, it is a common practice in some radial visualizations to render a separate icon on the ring's circumference for each *individual* entity in the data set (see for example [4] and [12]). The work by Livnat et al. [24] does alleviate the problem somewhat by allowing similar nodes to be clustered together, but does not fundamentally alter the "one icon per entity" paradigm. This approach does not scale to very large data sets; ultimately the icons for individual entities become too small to be useful. A key difference in our work is that we put attributes, not entities, on the ring's circumference. This approach is particularly well-suited to opinion polls, which typically have no more than a few dozen questions (attributes), but can and do have thousands

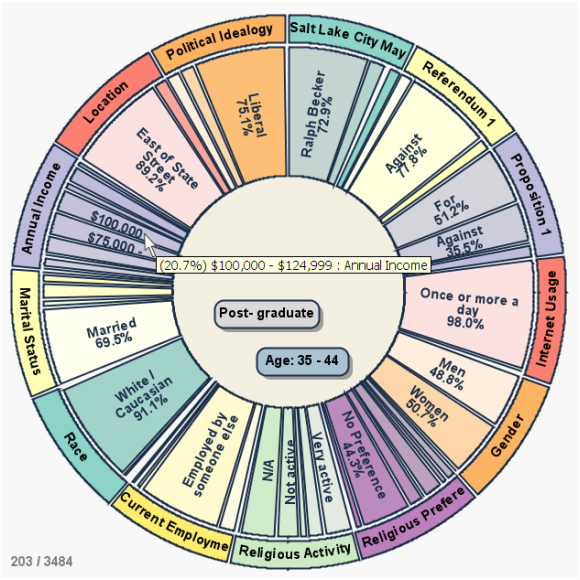

Fig. 5. In this pathological example, all 16 attributes from the November 2007 election data set have been placed on the canvas. 14 appear on the ring's circumference, and two appear in the ring's interior. Text is suppressed from sectors when they are too thin, however, labels are still available on demand; notice the tooltip in the middle of the image. The canvas's resolution could have been increased to give the sectors more room to grow, but in this case, we intentionally limited the area to show that the visualization is still useful even when many attributes are examined simultaneously. By design, the icons on the ring depict a relatively small number of attributes, while the number of entities in the data set can grow arbitrarily large.

of individual respondents (entities). In fact, a larger data set simply results in a richer experience for the user, because the data is representative of a larger population and therefore less subject to skew occasioned by inconsistencies in the data. We have tested our system with a data set of one million entities with only minimal slowdown (2 to 3 seconds on a standard PC) in system responsiveness.

Although this approach is not appropriate for data sets with hundreds or thousands of attribute types, opinion polls rarely exhibit this. Even if the user chooses to put every attribute on the ring (Fig. 5), this does not significantly compromise the utility of the visualization, as the most frequently occuring values for each attribute still occupy a dominant proportion of its sector.

Second, previous radial visualizations often render lines between nodes on the circumference of the ring to convey relationships within the data [4, 12, 24]. While this can be an effective technique for visual correlation when employed in limited quantities, as the number of lines increases, the visualization becomes virtually a "cloud" of intersecting lines, making it difficult for the user to gain insight regarding specific nodes. Holten [19] addresses this issue by bundling related edges together, but this too has its limitations. In contrast, our visualization renders lines between icons only in limited situations (see Section 4.1). Instead, there is an implied "many to many" correlation between the terms of the query shown in the ring's interior and the results of the query shown on the ring itself. Thus, associations within the data are implicit, and avoid a lot of crisscrossing lines which obscure the display.

5 Implementation Notes

5.1 Software prototype

As a practical demonstration of the concepts outlined in this paper, we have implemented a software prototype of the visualization. We call our prototype SQiRL, an acronym for *Simple Query Interface with a Radial Layout*. The client software is implemented in Java 1.6, with

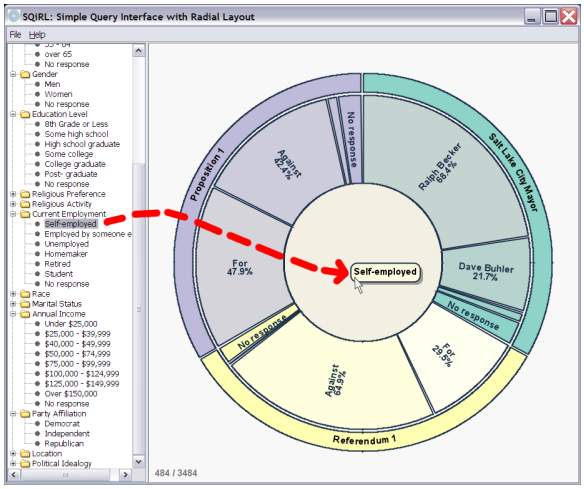

Fig. 6. Dragging a value into the ring's interior restricts the sample population to those who match that value for a given attribute.

a MySQL database backend. The default color scheme for the ring's sectors was chosen somewhat arbitrarily, but it can be customized via a "Preferences" dialog.

5.2 Data set

The data set used in our examples and screenshots is from an exit poll taken during the November 2007 mayoral election in Salt Lake City, Utah. While any number of survey data sets would have sufficed, we chose this one because its content was relevant to our base of volunteer testers, who reside primarily in the greater Salt Lake City area.

The survey was conducted by the Utah Colleges Exit Poll, a nonpartisan university-sponsored research program. The data set consists of 3,484 unique responses to the survey. The data set was given to us in SPSS format [8], but we converted it to a relational database for the purposes of our implementation.

6 Walkthrough

We describe a scenario of how the visualization could typically be used. The data set in this example is the November 2007 election data set described in Section 5.2. For a more detailed usage scenario, the reader is referred to the supplementary video on the conference DVD.

When the user is first presented with the visualization canvas, there are no values in the ring's interior. Thus, the results shown on the ring reflect the total population surveyed. By default, the ring shows the results for the three issues on the ballot.

Next, the user might pose the question, "Do people who run their own business tend to vote in a certain way?" To answer this question, the user restricts the sample population to entrepreneurs only. She does this by expanding the attribute labeled *Current Employment* in the side panel, and dragging the value labeled "Self-employed" into the ring's interior (Fig. 6). (Alternatively, she could have dragged the entire *Current Employment* attribute onto the canvas, and then selected and dragged the desired value from the resultant combo box into the ring's interior. These two approaches are functionally equivalent.)

As soon as the user drops the icon inside the ring, the system initiates a new batch of queries on the underlying data. The ring's sectors immediately begin to rescale via a smooth animation. When the animation completes, the stacked bar charts on the ring reflect the attributes of those survey respondents who identified themselves as "self employed."

Next, the user wants to further refine the query to "all voters who are self employed *and* who use the Internet more than once a day." This is done by dragging the value labeled *More than once a day* from the attribute *Internet Usage* into the ring's interior. As before, the percentages on the ring are updated to reflect the new query terms.

The user now wishes to see how well-educated the surveyed population is. Recall that any attribute in the data set can be placed on the ring. Thus, she drags the attribute *Education Level* from the side panel onto the ring. The attributes already on the ring gradually decrease in size to accommodate the newcomer, and the percentages for each value on the new attribute are shown.

With this gesture, the user has begun analyzing general characteristics of the survey population, unrelated to election results. She next removes all ballot-related issues from the ring and replaces them with other attributes of purely demographic interest, such as *Annual Income* or *Religious Preference*. The user can now perform queries on the population not necessarily related to voter behavior. This a strength of our technique: *each attribute of the data is manipulated in a completely analogous manner*.

By examining the data from the perspective of any paticular attribute, users can uncover trends that the compilers of the data set may not have envisioned. (Do people who live on one side of town tend to have higher incomes? Are married or single people more likely to earn advanced degrees?) This information is readily available in the data set, but is unlikely to show up in a report written to communicate a specific type of information, such as voter preferences, to a general audience. Such insights are only available when users are granted access to the original data set and a tool that makes arbitrary queries simple and fast.

7 Evaluation

7.1 User studies

To gain understanding and experience towards validating our technique, we conducted two preliminary user studies. The first involved two expert users, and the second involved 52 casual users. The purpose of these initial studies was to verify that this technique is easy to learn for both expert and novice users, not to perform a quantitative comparison against existing methods. Nevertheless, motivated by the positive feedback from these preliminary studies, we are in the process of preparing a formal comparative user trial in which we will compare users' performances with this tool versus existing visualization techniques.

7.1.1 Expert review

We conducted a qualitative expert review of our prototype implementation involving two political science professors from a neighboring university. Neither of the two had prior knowledge of our system, although both were expert users of commercial data analysis software. The evaluation consisted of an informal 30-minute demonstration and discussion. After less than 5 minutes of instruction on the interface, they felt comfortable using it to construct their own queries. One of the participants described the interface as "engaging," explaining how its interactivity engenders exploration. On the other hand, our expert reviewers were split on the radial style of the visualization. One thought it might be confusing to novices, while the other appreciated how it kept all the information in one place.

Overall, their response to the prototype was enthusiastic. One participant suggested that the interface could augment the election result visualization used by CNN [2]. They also recommended our visualization for use as a teaching aid in political science classes. Both participants appreciated the use of animation to create a smooth transition from one query result to the next.

7.1.2 Novice review

For our second user trial, we recruited 52 participants (20 women, 32 men) from the Salt Lake City area. Their ages ranged from 17 to 55, with a median age of 26. Of the 52 participants, relatively few (7) were computer science students. The test was run on a standard PC with a 1400x1050 resolution display. Each participant completed the user study individually, without collaboration with others.

The structure of this study was as follows. First, each participant watched a 3-minute video which gave a brief demonstration of the software. Each participant was then allotted 2 minutes of "free play" time in which he or she could practice using the interface. Next, we gave

Table 1. Tasks, average completion times, and success rates

#	Task	Completion Time	Success Rate
1	What percentage of all voters voted against Proposition 1?	17.8 sec	90.4%
2	What percentage of women voted for Ralph Becker?	22.7 sec	94.2%
3	What percentage of women whose current employment is "retired" voted for Ralph Becker?	26.6 sec	100%
4	What percentage of retired women attained an education level of "post-graduate?"	70.0 sec	90.0%
5	What percentage of retired people attained an education level of "some college?"	34.4 sec	88.5%
6	Of the total population surveyed, what percentage were men?	40.4 sec	94.2%
7	Of those voters whose location is east of State Street, what percentage earned a post-graduate degree?	47.7 sec	88.5%
8	Of those voters whose location is west of State Street, what percentage earned a post-graduate degree?	32.2 sec	92.3%
9	Of those voters whose location is west of State Street, what percentage claim a party affiliation of Democrat?	42.8 sec	92.3%

each participant a sequence of 9 analysis tasks, based on the November 2007 election data set. The first 3 tasks were straightforward queries regarding the results of the election. The final 6 tasks were more general, nudging the user to uncover broad demographic characteristics of the survey population. At the conclusion of the 9 tasks, each participant was invited to submit written comments. Table 1 shows the tasks together with the average completion times and success rates.

A large majority of our participants (88%) said that they enjoyed using the interface. Slightly over two thirds (71%) completed all 9 analysis tasks with no errors, however there was a wide variation in the amount of time it took participants to complete the tasks. Of those who finished the tasks successfully, the fastest took 2 minutes and 43 seconds, and the slowest took 12 minutes and 47 seconds. So while most participants reportedly found the visualization fun and useful, clearly some "got it" faster than others.

The most revealing observation was that some users were confused about the difference between the gesture required for specifying the characteristics of the sample population versus the gesture for exposing numerical information about that population. Consider, for example, task #9: "Of those voters whose location is west of State Street, what percentage claim a party affiliation of Democrat?" The required gesture was to drag the icon labeled "West of State Street" into the ring's interior (thus limiting the sample population to only those who live west of State Street), and then drag the attribute "Party Affiliation" to the ring's circumference (thus revealing the relative representation of each political party in the sample population). Some participants first tried the opposite approach, placing the "Democrat" icon in the ring's interior, and the "Location" attribute on the ring. This is a perfectly valid query, but it poses a different question, namely, "What percentage of all Democrats live west of State Street?" This will be something to watch for in future user studies and in subsequent evolutions of the visualization design.

Informal qualitative feedback was predominantly positive. For example, many participants voluntarily lingered after they had completed the tasks, in order to explore the data set on their own. One participant wrote, "I'm not good with new programs, but it was easy to catch on." Another participant commented, in comparison with current data analysis tools, "This ... is way easier to use than a spreadsheet or pivot table." Especially encouraging were comments such as, "I watch a lot of political news programs and I have never seen any polling data that was so comprehensive," and "I need this for my work!" Negative feedback related mainly to confusion over whether to put icons inside the ring or on the ring, as previously noted.

7.2 Suitability for purpose

Our interactive technique and visualization provide a fast and effective way to analyze opinion poll data. As with any technique, it is important to use the right tool for the right job. Our query metaphor may not be appropriate for *every* analysis problem, but it *is* well suited for the kinds of analysis performed regularly by polling agencies in their public reports, and similar applications.

To justify this statement, we have read a number of press releases and whitepapers from a variety of public opinion organizations. To focus the present discussion, however, we mention three typical reports: a press release prepared by the Utah Colleges Exit Poll regarding the November 2007 Salt Lake City election [41], a June 2008 report from Gallup, Inc. regarding the demographics of likely voters in the U.S. presidential election [13], and a June 2008 report from Rasmussen Reports, Inc. regarding public perceptions of U.S. presidential candidates [30].

Each of these reports makes several statistical assertions similar to "X *percent of all voters in group* Y *voted for candidate* Z." Given the proper data sets, our visualization could easily confirm the results reported in these reports, as well as spot other trends not necessarily reported. We found no statements in the vast majority of press releases we examined that could not be readily performed with our system.

Thus, we do not suggest that our technique does anything that *cannot* be done with existing tools; rather, it makes certain classes of frequently recurring queries both *easy* and *fast* to perform, with minimal training — a conclusion supported by our preliminary user studies. Further testing is necessary to establish statistically significant conclusions with respect to competing visualizations.

8 Conclusions and Future Work

This paper presents a visual query language for analyzing poll-based surveys. The method consists of a direct manipulation interface in which icons representing entire attributes and individual values can be introduced into and withdrawn from queries in a straightforward manner. The highly interactive visualization allows users to experiment with the data and test multiple hypotheses in rapid succession. Because our technique uses attributes, not distinct entities, as the basic unit of visualization, it can be used to view very large data sets. We have tested the prototype with data sets containing up to one million entities while still maintaining interactive speeds. Preliminary user trials suggest that the interface is simple enough for both novice and expert users to learn quickly. We are currently planning additional quantitative user studies to compare user performance using our technique versus existing tools.

In the process of creating the visualization metaphor described in this paper, we faced a number of design and interface challenges. We overcame many of these, but there are still a number of remaining issues. First, we have not yet designed an elegant way to visualize the special case of when the attribute for a value inside the ring also appears on the circumference. We currently handle this by making all the subsectors equiangular for that attribute, but this may not be intuitive. Second, and more importantly, a clean interface for comparing the results of two queries is needed. We have explored a variety of design options, but are still searching for one that balances the competing virtues of conserving screen real estate and maintaining a non-cluttered display. Additional user testing will be required to determine which of our several hypotheses will be most meaningful to users.

Although initially created for the purpose of analyzing opinion poll data, we believe that this visualization metaphor might be generalizable to many kinds of tabular data where the set of attributes is not excessively large. For example, the visualization could be used in a healthcare setting to analyze demographic characteristics of individuals who have contracted a particular disease. In the domain of homeland security and law enforcement, the visualization could be used to discover commonalities among known terrorists or other criminals.

More experimentation is needed to determine how adaptable the interface is to data drawn from applications other than opinion polls.

Acknowledgements

The authors wish to thank Andries van Dam and Dan Olsen for their useful feedback. Our thanks to Quin Monson of the Utah Colleges Exit Poll for permission to use the November 2007 election data set in the development and testing of our prototype. We are grateful to the anonymous reviewers, whose valuable suggestions greatly improved this paper.

References

[1] R. L. Burden, J. D. Faires, and A. C. Reynolds. *Numerical Analysis*, pages 81–130. Prindle, Weber & Schmidt, 1978.

[2] Cable News Network LP. Election Center 2008. http://www.cnn.com/ELECTION/2008/. Accessed 24 March 2008.

[3] M. Cammarano, X. L. Dong, B. Chan, J. Klingner, J. Talbot, A. Halevy, and P. Hanrahan. Visualization of heterogeneous data. In *InfoVis 2007*, pages 1200–1207, 2007.

[4] Daisy Analysis Ltd. Daisy 2003. http://www.daisy.co.uk/. Accessed 14 November 2007.

[5] G. M. Draper and R. F. Riesenfeld. Interactive fan charts: A space-saving technique for genealogical graph exploration. In *8th Annual Workshop on Technology for Family History and Genealogical Research*, Provo, Utah, USA, 2008. Brigham Young University.

[6] N. Elmqvist, J. Stasko, and P. Tsigas. DataMeadow: A Visual Canvas for Analysis of Large-Scale Multivariate Data. In *IEEE Symposium on Visual Analytics Science and Technology (VAST 2007)*, volume 2, pages 187–194, Oct. 2007.

[7] Email correspondance between the author and the associate director of Quinnipiac University Poll. Dated 17 March 2008.

[8] A. Field. *Discovering Statistics Using SPSS*. Sage Publications Ltd, 2nd edition, 2005.

[9] P. M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. In *Journal of Experimental Psychology*, pages 381–391, 1954.

[10] C. Frye. *Microsoft Office Excel 2007 Step by Step*. Microsoft Press, 2007.

[11] G. W. Furnas and S. J. Rauch. Considerations for information environments and the NaviQue workspace. In *INEX Workshop*, pages 79–88, 2004.

[12] J. Galloway and S. J. Simoff. Network data mining: methods and techniques for discovering deep linkage between attributes. In *APCCM '06: Proceedings of the 3rd Asia-Pacific conference on Conceptual modelling*, pages 21–32, Darlinghurst, Australia, Australia, 2006. Australian Computer Society, Inc.

[13] Gallup, Inc. About One in Four Voters Are "Swing Voters". http://www.gallup.com/poll/108466/About-One-Four-Voters-Swing-Voters.aspx. Accessed 30 June 2008.

[14] R. Goldman and J.Widom. Interactive query and search in semistructured databases. In *International Workshop on the Web and Databases, WebDB'98*, pages 52–62, 1998.

[15] S. Havre, E. Hetzler, K. Perrine, E. Jurrus, and N. Miller. Interactive visualization of multiple query results. In *InfoVis 2001*, pages 105–112, 2001.

[16] J. Heer and G. G. Robertson. Animated transitions in statistical data graphics. In *InfoVis 2007*, pages 1240–1247, 2007.

[17] I. Herman, G. Melançon, and M. S. Marshall. Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):24–43, Jan. 2000.

[18] B. Hetzler, P. Whitney, L. Martucci, and J. Thomas. Multi-faceted insight through interoperable visual information analysis paradigms. In *InfoVis 1998*, pages 137–144, 161, 1998.

[19] D. Holten. Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. In *InfoVis 2006*, pages 741–748, 2006.

[20] A. Inselberg and B. Dimsdale. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *IEEE Visualization '90*, pages 361–378, 1990.

[21] B. S. Johnson. *Treemaps: Visualizing Hierarchical and Categorical Data*. PhD thesis, University of Maryland, 1993.

[22] D. A. Keim, F. Mansmann, J. Schneidewind, and T. Schreck. Monitoring network traffic with radial traffic analyzer. In *2006 IEEE Symposium On Visual Analytics Science And Technology (VAST)*, pages 123–128, 2006.

[23] D. A. Keim, J. Schneidewind, and M. Sips. CircleView: a new approach for visualizing time-related multidimensional data sets. In *AVI '04: Proceedings of the working conference on Advanced visual interfaces*, pages 179–182, New York, NY, USA, 2004. ACM Press.

[24] Y. Livnat, J. Agutter, S. Moon, and S. Foresti. Visual correlation for situational awareness. In *InfoVis 2005*, pages 95–102, 2005.

[25] F. Nightingale. *Notes on matters affecting the health, efficiency, and hospital administration of the British army : founded chiefly on the experience of the late war*. London: Harrison and Sons, 1858.

[26] M. L. Northway. *A Primer of Sociometry*. University of Toronto Press, 2nd edition, 1967.

[27] P. Pirolli and S. Card. Information foraging in information access environments. In *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 51–58, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[28] W. Playfair. *The statistical breviary; shewing, on a principle entirely new, the resources of every state and kingdom in Europe; illustrated with stained copper plate charts, representing the physical powers of each distinct nation with ease and perspicuity. To which is added, a similar exhibition of the ruling powers of Hindoostan*. London: for J. Wallis, 1801.

[29] S. Polyviou, G. Samaras, and P. Evripidou. A relationally complete visual query language for heterogeneous data sources and pervasive querying. In *ICDE 2005. Proceedings of 21st International Conference on Data Engineering*, pages 471–482, 2005.

[30] Rasmussen Reports, Inc. Public Perceptions of Obama and McCain Shifting Rapidly. http://www.rasmussenreports.com/public_content/politics/election_20082/2008_presidential_election/public_perceptions_of_obama_and_mccain_shifting_rapidly. Accessed 30 June 2008.

[31] S. F. Roth, P. Lucas, J. A. Senn, C. C. Gomberg, M. B. Burks, P. J. Stroffolino, J. A. Kolojejchick, and C. Dunmire. Visage: A user interface environment for exploring information. In *InfoVis 1996*, pages 3–16, 1996.

[32] C. E. Scheidegger, H. T. Vo, D. Koop, J. Freire, and C. T. Silva. Querying and creating visualizations by analogy. *IEEE Transactions on Visualization and Computer Graphics*, 13(3):1560–1567, 2007.

[33] B. Shneiderman. Session 2: "InfoVis for the Masses". *IEEE Transactions on Visualization and Computer Graphics*, 13(6):iii, Nov. 2007.

[34] A. Silberschatz, H. F. Korth, and S. Sudarshan. *Database System Concepts*. McGraw Hill, 4th edition, 2002.

[35] V. Sinha and D. R. Karger. Magnet: Supporting navigation in semistructured data environments. In *SIGMOD*, pages 97–106, 2005.

[36] I. Spence. No humble pie: The origins and usage of a statistical chart. *Journal of Educational and Behavioral Statistics*, 30(4):353–368, 2005.

[37] J. Stasko and E. Zhang. Focus+context display and navigation techniques for enhancing radial, space-filling hierarchy visualizations. In *InfoVis 2000*, pages 57–65, 2000.

[38] C. Stolte, D. Tang, , and P. Hanrahan. Polaris: A System for Query, Analysis, and Visualization of Multidimensional Relational Databases. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):52–65, 2002.

[39] M. Stonebraker. Visionary: A next generation visualization system for data bases. In *SIGMOD*, page 635, 2003.

[40] Swivel. . http://www.swivel.com/. Accessed 30 June 2008.

[41] Utah Colleges Exit Poll. SLC Voters Divided by Religion, Geography. http://exitpoll.byu.edu/Main/documents/Utah Colleges Exit Poll Election Night Release.pdf. Accessed 28 March 2008.

[42] M. Wattenberg, J. Kriss, and M. McKeon. ManyEyes: a Site for Visualization at Internet Scale. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1121–1128, 2007.

[43] L. Wilkinson. *The Grammar of Graphics (Statistics and Computing)*. Springer, 1st edition, 1999.

[44] K.-P. Yee, D. Fisher, R. Dhamija, and M. Hearst. Animated exploration of dynamic graphs with radial layout. In *InfoVis 2001*, pages 43–50, 2001.

[45] M. M. Zloof. Query-by-example: a data base language. *IBM Systems Journal*, 16(4):324–343, 1977.