Visual Guidance in the Exploration of Large Databases

Jiang Du Ian Spence Michael J. McGuffin

University of Toronto and École de Technologie Supérieure (ÉTS)

Abstract

Software tools for visualizing very large multidimensional databases have become increasingly important to discover interesting relationships among variables. While current tools implement operations such as drilling down, rolling up, and slicing data tables to help users notice interesting features of the data, the onus is on the user to choose the dimensions for drill down, or other operations. Expert knowledge is required to do this effectively and, since many users are novices, incorrect choices often lead to dead-ends, backtracking, confusion, and frustration. We suggest a novel approach to the selection of dimensions that relies on the interactive presentation of small multiples of thumbnail visualizations, before performing drill down or roll up operations. These previews of distributions, relationships, and associations, before variable selection, compel visual comparisons of change and difference, thus highlighting the options that are most likely to lead to productive paths.

1 Introduction

It is now possible to accumulate data at rates and in volumes that were unthinkable only a few years ago. Governments, institutions, e-commerce vendors, and others, collect terabytes of data in a matter of a hours or even minutes. In the sciences, it is also unexceptional to acquire very large amounts of data in short periods of time; research in proteomics and genomics, astronomy, particle physics, weather and climate, results in the creation of large databases. These huge volumes of data have brought both opportunities and challenges for the data analyst. The discovery of interesting patterns, associations, and structure in very large databases is an increasingly important challenge.

Statistical approaches for the discovery of relationships are generally predicated on the notion that much of the information in the typical dataset is redundant or irrelevant or merely noise. Consequently, techniques that are designed to reduce the dimensionality of the variable space have traditionally been used to analyze multivariate data sets. These include principal component analysis, factor analysis, multidimensional scaling, and a wide range of algorithms for clustering. Sometimes these dimensionality reduction techniques are combined with regression modeling, machine learning algorithms, or pattern analysis. Unfortunately, however, these procedures are often impractical with large data sets; the sheer size of many current real-world databases makes implementing traditional statistical approaches extremely difficult. Consequently, more computationally tractable techniques have been developed.

The construction of a data warehouse is a common preliminary step to subsequent on-line analytical processing (OLAP). The data warehouse is maintained separately from the operational database, often in the form of a multidimensional data cube, and is intended for modeling and analysis, not for transaction processing. OLAP operations such as slicing, dicing, pivoting, rolling up, and drilling down are intended to provide views of the data that make it easier to discover interesting patterns and relationships. Although there have been some notable attempts to automate the discovery of significant associations [eg. 1,2,3,4], typically a human analyst must select the particular OLAP operations to be implemented and subsequently make sense of the tables and numbers that are produced. This is not always a simple process, particularly for novice users, and appropriate visualizations can assist the analyst to discover interesting relationships that might otherwise go unnoticed.

Tableau is a popular commercial tool for the visual exploration of large databases [5,6,7]. The software makes it easy to drill down, roll up, and slice data sets, switching among tabular views using drag-and-drop operations. The *Tableau* interface facilitates a question-and-answer workflow, during which the analyst may notice interesting features or relationships (e.g.: "Sales of product X fall more in December than expected. Why?"), which may suggest further operations such as drilling down to find likely explanations ("Low sales are due to certain stores not offering rebates to compete with other stores.") *Tableau* can also assist the analyst by automatically choosing default graphical presentations (bar chart, line graph, etc.). However, ultimately, the onus is on the user to choose the dimensions for drill-down or other operations. Inappropriate choices can lead to dead-ends, backtracking, confusion, and frustration, especially if the data are sparse in some regions. Thus *Tableau*, and other similar data-slicing tools, offer little guidance regarding the selection of the dimensions that define the tabular presentation. Visualizations are presented after the drag and drop selection of dimensions; there are no preview visualizations. Depending on the context, considerable expertise may be required to make optimal use of tools like *Tableau*.

Another general problem with current tools like *Tableau* is that many users are novices with only a rudimentary understanding of the structure of the database and the different possible ways to view it. For example, middle-level managers may be relatively unfamiliar with database management tools and may have difficulty in interpreting summaries that are presented in tabular form. These users may be hesitant to learn how to use a database querying system, even those with a seemingly simple drag-and-drop interface like the one in *Tableau*. However, such users are likely to be key decision makers and it is important to give them visualization tools to support their intuitions and help them answer important questions. A superior interface is required to achieve this goal.

We propose a hierarchical database visualization interface with preview thumbnails that will help guide users to navigate and to choose appropriate dimensions for drilling down and rolling up. The interface should be intuitive and easy-to-use; and, importantly, it should implement more than just drag-and-drop selection of variables. Furthermore, it should not require sophisticated expertise in database technicalities. The interface should provide automatic previews in ways that make interesting dimensions “discoverable”. Animated transitions should make clear what is happening during drilling down or rolling up. In other words, the interface should provide natural visual guidance.

Our proposed interface design was tested using a prototype that has been implemented in a JavaScript front-end. The figures in this paper show screenshots from the prototype. Note that we paid little attention to aesthetic considerations, using simple outlines and shading only.

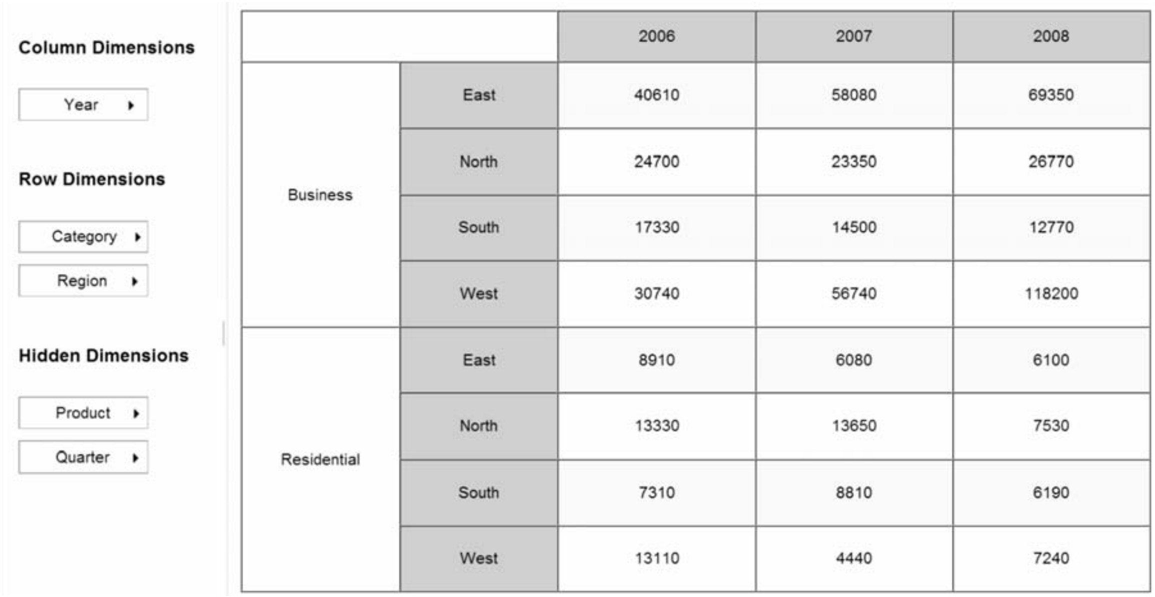
2 The Interface

We illustrate the interface using a small artificial example: a simulated e-commerce database with five variables or dimensions. A real-world database would have many more dimensions.

Figure 1 shows one of the many possible tabular views of the database. The table shown corresponds to a particular view of the data in a unique row-subrow-column arrangement. Other row-subrow-column-subcolumn arrangements are possible and different choices of the visible (row and column) and hidden dimensions would yield different views. The frequencies in the cells represent the numbers of sales as a function of the type of community, geographical region, and year. The visible dimensions shown are Year (2006, 2007, 2008), Category (Business, Residential), and Region (North, South, East, West) and there are two additional hidden dimensions, Product and Quarter. The visible (row and column) and hidden dimensions are always listed to the left of the table.

2.1 Drilling Down

If the cursor is hovered over one of the hidden dimensions at left (as shown in Figure 2), thumbnails of the distributions of values along that dimension appear instantly within each cell. The frequency in each cell indicates the number of sales over the four quarters, and the thumbnail bar charts show the sales broken down by Quarters. These small multiples [8,9] help the user to decide, quickly, whether it might be useful to drill down on that dimension. The thumbnails allow viewers



Column Dimensions

Year ▸

Row Dimensions

Category ▸

Region ▸

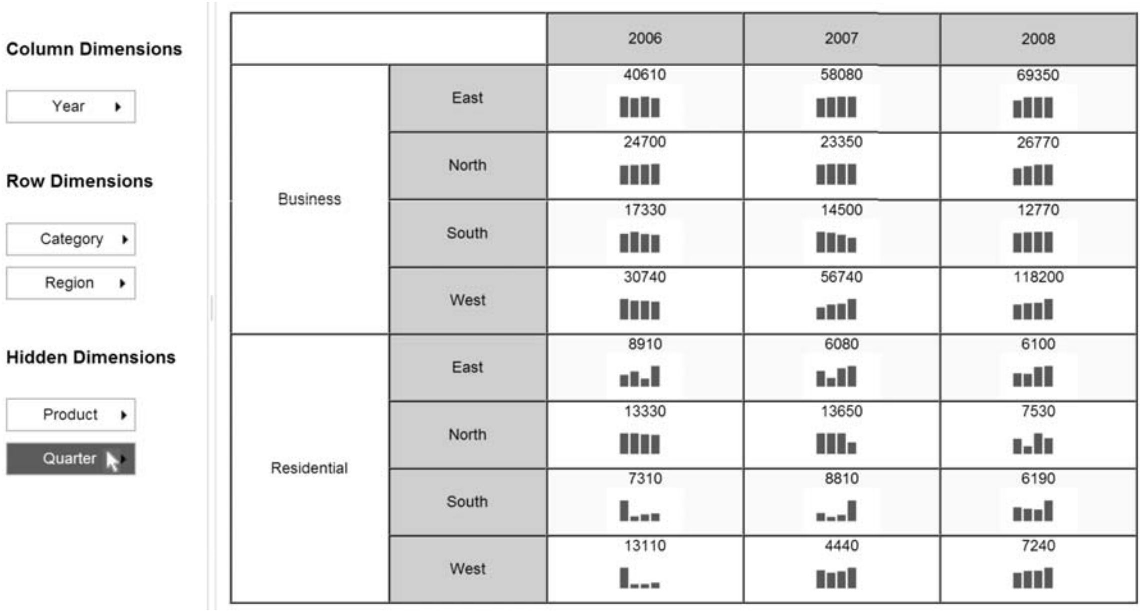
Hidden Dimensions

Product ▸

Quarter ▸

		2006	2007	2008
Business	East	40610	58080	69350
	North	24700	23350	26770
	South	17330	14500	12770
	West	30740	56740	118200
Residential	East	8910	6080	6100
	North	13330	13650	7530
	South	7310	8810	6190
	West	13110	4440	7240

Figure 1: Screenshot of a particular tabular view of a database with five dimensions. Three dimensions are visible, Year, Category, and Region, while the two remaining dimensions, Product and Quarter, are hidden).



Column Dimensions

Year ▸

Row Dimensions

Category ▸

Region ▸

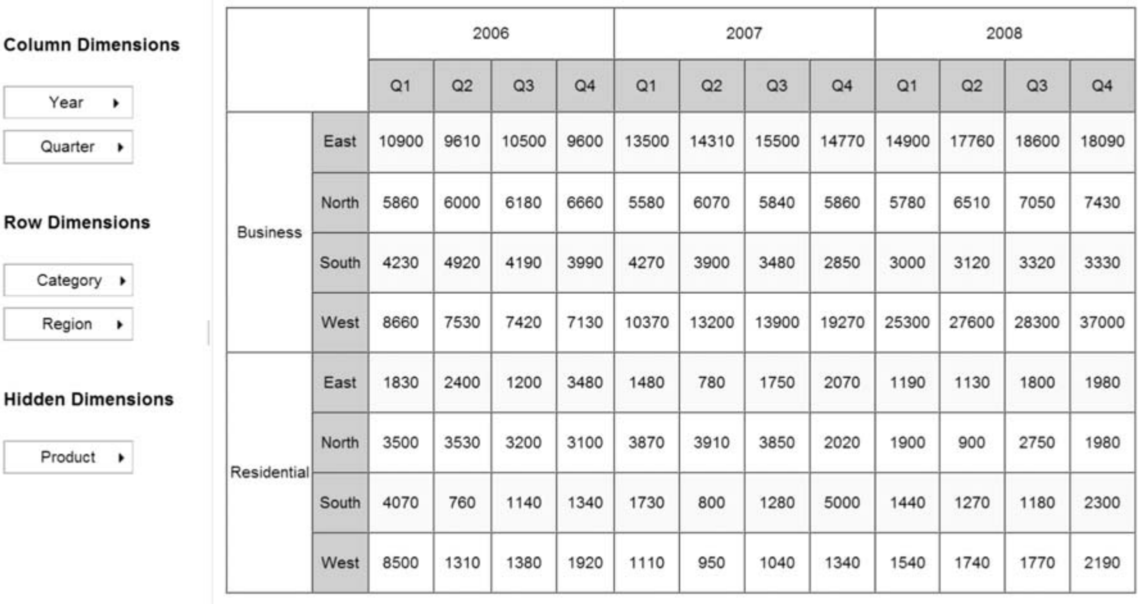
Hidden Dimensions

Product ▸

Quarter ▸

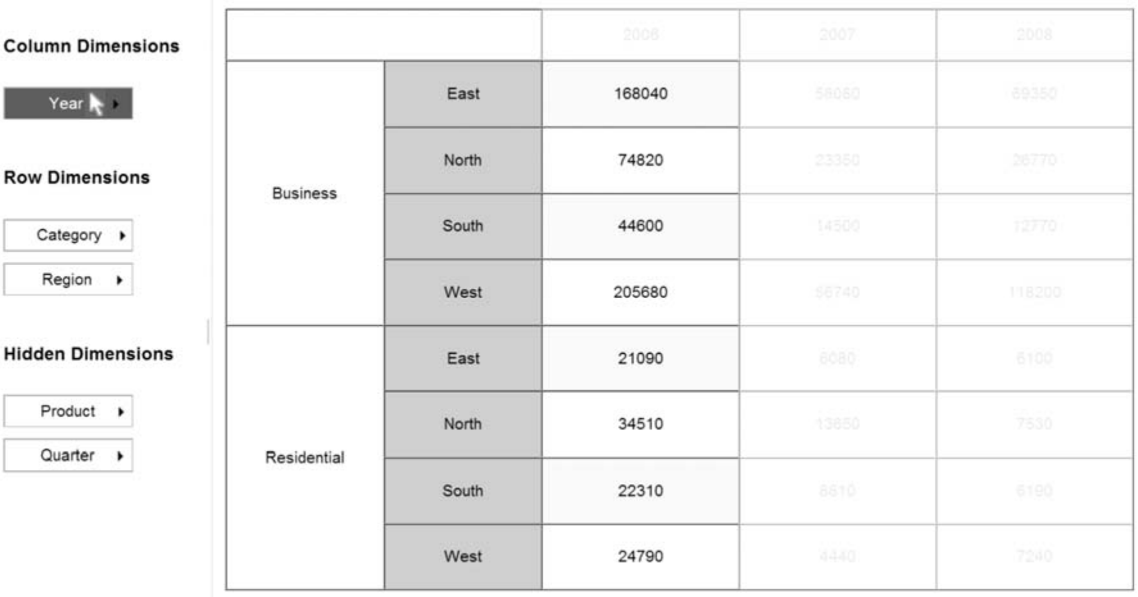
		2006	2007	2008
Business	East	40610	58080	69350
	North	24700	23350	26770
	South	17330	14500	12770
	West	30740	56740	118200
Residential	East	8910	6080	6100
	North	13330	13650	7530
	South	7310	8810	6190
	West	13110	4440	7240

Figure 2: Screenshot of the cursor hovering over the hidden dimension, Quarter. The number in each cell shows the total sales for the four quarters and the thumbnail bar charts show the sales for individual quarters



		2006				2007				2008			
		Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
Business	East	10900	9610	10500	9600	13500	14310	15500	14770	14900	17760	18600	18090
	North	5860	6000	6180	6660	5580	6070	5840	5860	5780	6510	7050	7430
	South	4230	4920	4190	3990	4270	3900	3480	2850	3000	3120	3320	3330
	West	8660	7530	7420	7130	10370	13200	13900	19270	25300	27600	28300	37000
Residential	East	1830	2400	1200	3480	1480	780	1750	2070	1190	1130	1800	1980
	North	3500	3530	3200	3100	3870	3910	3850	2020	1900	900	2750	1980
	South	4070	760	1140	1340	1730	800	1280	5000	1440	1270	1180	2300
	West	8500	1310	1380	1920	1110	950	1040	1340	1540	1740	1770	2190

Figure 3: Screenshot of the result of drilling down in columns along the Quarter dimension, which now becomes a column dimension (in the table and in the menu at left). The transition from the tables in Figures 1 and 2 to this table is smoothly animated.



		2006	2007	2008
Business	East	168040	58080	69350
	North	74820	23350	26770
	South	44600	14500	12770
	West	205680	56740	118200
Residential	East	21090	6080	6100
	North	34510	13650	7530
	South	22310	8810	6190
	West	24790	4440	7240

Figure 4: The same initial view as in Figure 1. However, when the cursor hovers over a row or column dimension, the data to be rolled up along that dimension fade out and the new totals fade in. Here the individual years fade out and the data shown are the total sales numbers rolled up over the three years.

to make comparisons at a glance. Since the graphical format is consistent across cells, changes in shape reflect the changes in the data that are easier to detect in a graphical rather than in a numerical format. The use of small multiples provides an immediate visual comparison of changes and differences among objects. Users quickly identify those cells that are different. Or, they may notice trends in how the shapes change over one or more of the visible dimensions.

If the user clicks on a hidden dimension, a menu pops up (not shown here) beside the dimension, allowing the user to select "Drill down in columns" or "Drill down in rows". If the former choice were selected, the result would be the table shown in Figure 3, which shows the result of drilling-down in columns along the Quarter dimension. Quarter now becomes a column dimension in the table and also in the menu at left. In our prototype, an animated transition (lasting approximately 0.5 to 1.0 second) smoothly fades in the new column divisions and spreads out the columns to help the user understand the meaning of the drill-down operation. Although not yet implemented, a potential improvement would be to animate the bars in the thumbnails which would split up horizontally into each of the columns to reinforce their meaning.

2.2 Rolling Up

This is the opposite of drilling down. If the cursor hovers over a column or row dimension at left in Figure 1, the result is shown in Figure 4. This provides an immediate preview of the result of rolling up along that dimension. The sums of sales, rolled up across the years, are shown in the usual dark font, whereas the numbers of sales in the individual years fade to a light gray.

If the user decides to click on Year, a small menu pops up, as shown in Figure 5. If rolling up is selected, the column lines dividing the years fade out and the remaining column(s) collapse into the final view shown in Figure 6. Note that Year then becomes a hidden dimension and the table in Figure 6 resembles the preview that was obtained by hovering the cursor over Year in Figure 4.

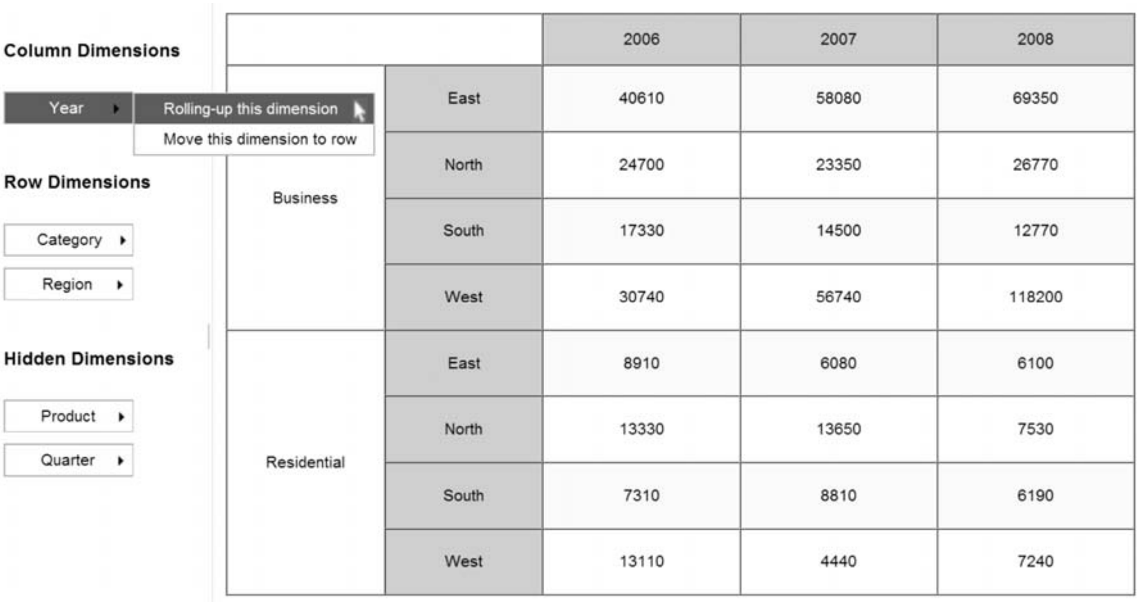
2.3 Visual Guidance

The user may drill down and roll up along any dimensions multiple times. This approach has some similarity to how *Tableau* [5] and other statistical software for analyzing contingency tables, display the table that shows the current view of the database. However, our interface has at least three advantages relative to the approach used in *Tableau* and other tools.

First, the operations required of the user are even simpler than dragging and dropping and, more critically, are "discoverable". With *Tableau*, the user needs to have some prior intuition or knowledge of which dimensions might yield productive selections. With our interface, since the dimensions (both visible and hidden) listed to the left of the table are always available, users will naturally pass the cursor over them to see what happens. The resulting immediate visual feedback in the table gives a strong hint of what clicking will do. Clicking down on a dimension pops up a menu containing explicit choices ("Drill down in columns", "Drill down in rows", or "Roll up"), thus providing further guidance to the user, who can experiment with these operations, and receive immediate visual feedback. In contrast, with *Tableau*, users must hover over a dimension, click down, and then drag the dimension to an appropriate target location in the user interface before receiving any visual feedback.

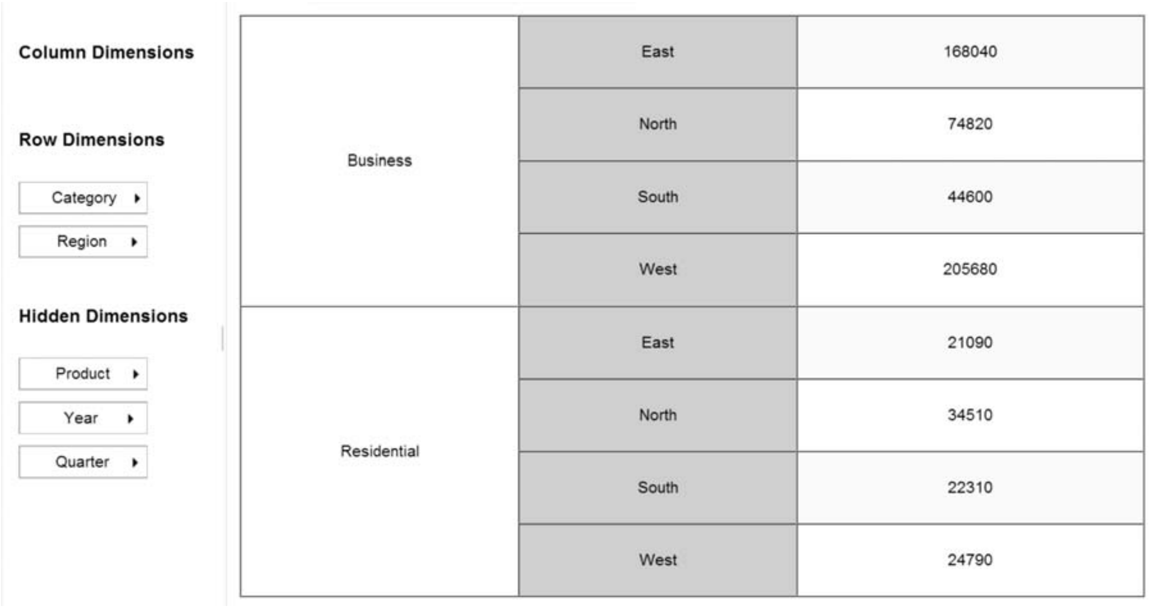
A second advantage of our approach is that transitions between table views are performed using smooth animations, thus helping the user to understand the meaning of drilling down and rolling up. The use of smoothly animated transitions has become more common in visualization systems [10,11,12], but, so far, has not been applied to tabular views of a database. Note that the animated transitions could be performed not just by showing columns or rows fade out+collapse or fade in+expand, but could simultaneously show dimensions listed to the left of the table smoothly rearranging themselves. For example, the Year dimension would move down to the list of hidden dimensions during the rolling up operation illustrated in Figures 4 to 6.

A third advantage is our automatic preview of small multiples. These thumbnail charts show what the data look like along a given dimension before actually drilling down. While *Tableau* can also display small multiples, it does so only after the user has made the selection of the dimensions that define the new tabular view. No guidance is provided in making this selection. Our approach can be thought of as providing the user with a kind of "information scent" [13,14] to guide



		2006	2007	2008
Business	East	40610	58080	69350
	North	24700	23350	26770
	South	17330	14500	12770
	West	30740	56740	118200
Residential	East	8910	6080	6100
	North	13330	13650	7530
	South	7310	8810	6190
	West	13110	4440	7240

Figure 5: Screenshot of the popup obtained by clicking on the visible dimension, Year. Two choices are presented. In this case, the user plans to roll up the dimension Year and will select the first option



Business	East	168040
	North	74820
	South	44600
	West	205680
Residential	East	21090
	North	34510
	South	22310
	West	24790

Figure 6: Screenshot of the result of the roll up operation on Year, which becomes a hidden dimension.

the drill down operations. Without the visual guidance provided by small multiples of thumbnails, the user could easily spend much time drilling down blind alleys only to have to back up and try a different direction. This kind of frustrating retracing of steps is not conducive to effective exploratory data analysis.

3 Visualizations

Although we have only implemented pivot tables with embedded thumbnail bar charts in our current prototype, our design could be extended to support any of the visualizations found in *Tableau*, or other similar software for the statistical analysis of multidimensional data tables. However, unlike in *Tableau*, where the visualizations are the consequence of the selection of particular dimensions, our visualizations are previews designed to provide visual clues to the discovery of interesting relationships before the selection of dimensions.

3.1 One Hidden Dimension

In the drill-down operation illustrated in Figures 2 and 3, the thumbnail previews are shown as small bar charts below the number in each cell. This is just one possible way to display the preview visualizations. Another possibility is to draw the thumbnail semi-transparently on top of the number (so that more space can be devoted to the thumbnail). Also, obviously, other graphics may be used in place of bar charts. Examples include line graphs, pie charts, radar/star charts, horizontal bars, trees, sparklines, and so on. Figure 7 shows some examples. Other possible candidates that take up little space are listed in [15,16].

Note also that the interface could be enhanced by highlighting dimensions or slices within the data that have been visited before or that match a search criterion (for example, all sales above $50,000).

3.2 Two Hidden Dimensions

When the cursor is hovered over a single hidden dimension, thumbnail previews of the distribution of that dimension are generated as described above. It may also be of interest to look at the association of that particular hidden dimension with another hidden dimension. Although we have not completed the implementation of this feature, if the user holds down the CTRL key and then hovers the cursor over any one of the other hidden dimensions, a graphic summarizing the association between the two selected hidden dimensions will appear in each cell. This could be a scatterplot, a contour/density plot, a heatmap, mosaic, or a glyph summarizing the degree of association as low medium or high (see Figure 8).

In Figure 8 (bottom), the shapes of the glyphs represent the magnitude of the correlation between the two selected dimensions. These glyphs are similar to Corrgrams [17] in shape and in function, however Corrgrams are used within a scatter plot matrix showing all possible pairs of dimensions whereas, in our application, the glyphs in Figure 8 (bottom) show correlations between the *same* two dimensions for many different slices of the data (each slice corresponding to a cell in the table).

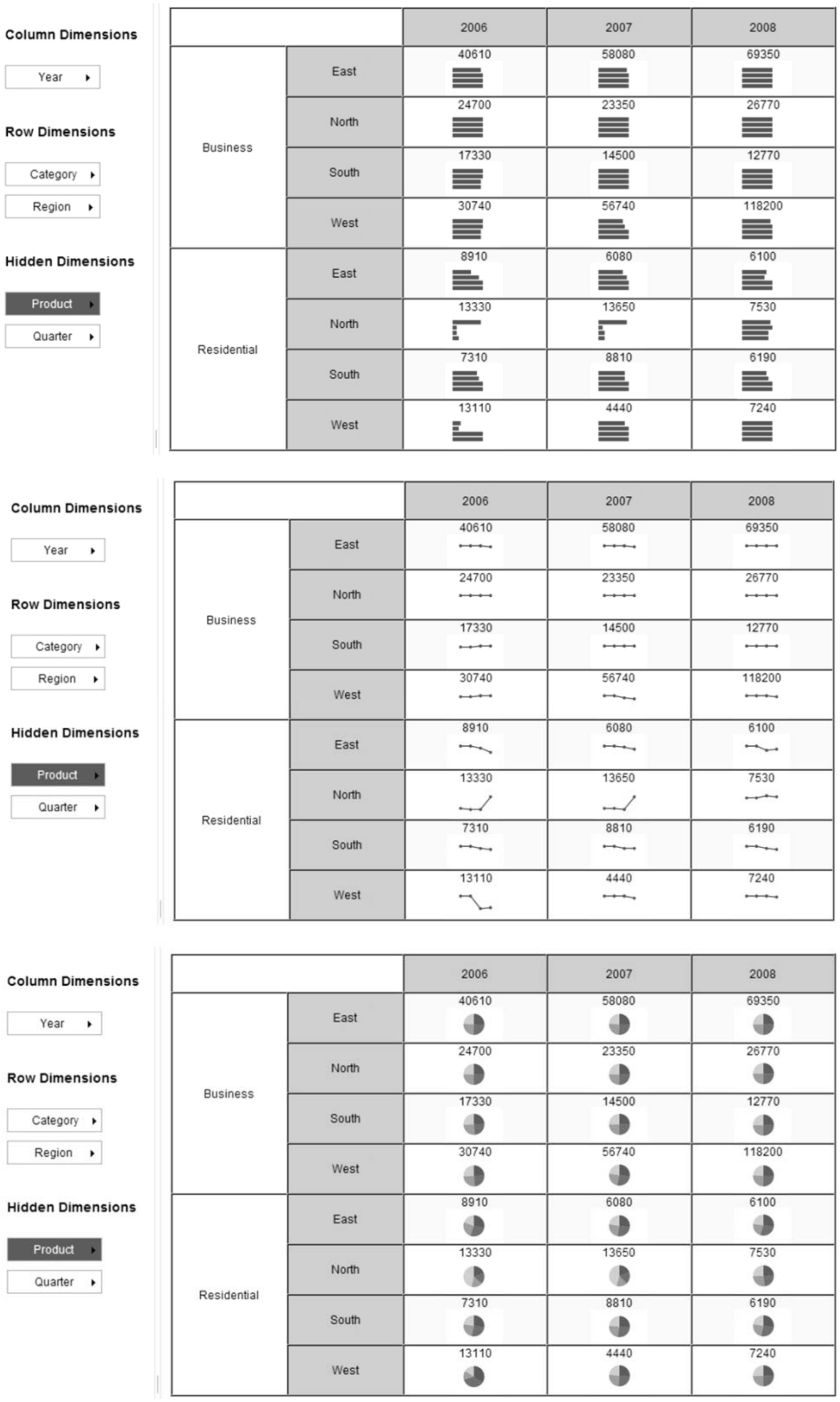
3.3 Visible Dimensions

It may often be useful for the analyst to request a visual representation of the association among the visible dimensions. This could be done by magnitude coding of the cell frequencies using color, density, or by displaying icons or pictograms (see Figure 9 for an example using grayscale).

3.4 Unusual Cells

If the analyst notices an unusual cell, it should be possible to obtain more information by hovering the cursor over the cell, thus bringing a variety of information related to the corresponding cell to the foreground [9]. In addition to thumbnail representations, the information could also include statistical summaries of relationships involving the hidden dimension and the visible dimensions.

Another interesting possibility would be to develop methods of automatically restructuring the tabular view by conditioning on the cell in question. The hidden dimension could be moved to become the sole (and visible) column dimension while all other dimensions (visible and hidden) become hidden. Then the user would be able to make a rapid visual assessment of the relationships between the (visible) column dimension and the remaining (hidden) dimensions by hovering the cursor over each hidden dimension in turn. One or two of the most interesting hidden dimensions could then be moved to the rows for a more nuanced picture. This sequence could be assisted by the provision of appropriate popup menus.

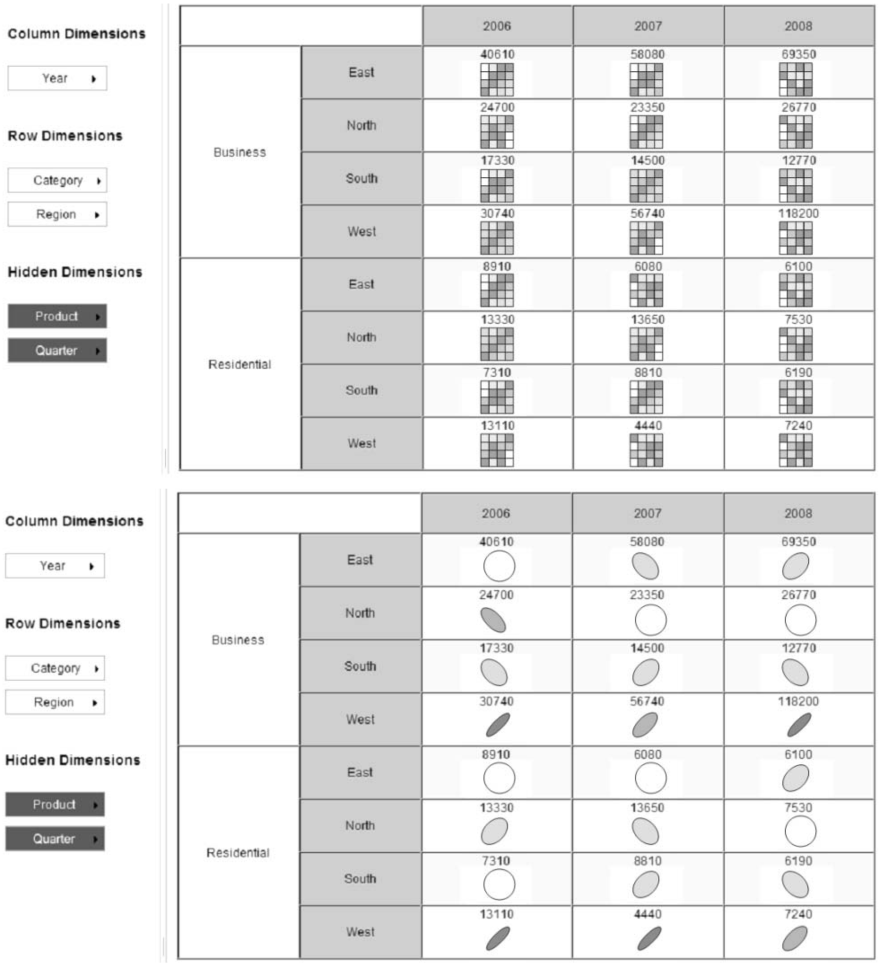


		2006	2007	2008
Business	East	40610	58080	69350
	North	24700	23350	26770
	South	17330	14500	12770
	West	30740	56740	118200
Residential	East	8910	6080	6100
	North	13330	13650	7530
	South	7310	8810	6190
	West	13110	4440	7240

		2006	2007	2008
Business	East	40610	58080	69350
	North	24700	23350	26770
	South	17330	14500	12770
	West	30740	56740	118200
Residential	East	8910	6080	6100
	North	13330	13650	7530
	South	7310	8810	6190
	West	13110	4440	7240

		2006	2007	2008
Business	East	40610	58080	69350
	North	24700	23350	26770
	South	17330	14500	12770
	West	30740	56740	118200
Residential	East	8910	6080	6100
	North	13330	13650	7530
	South	7310	8810	6190
	West	13110	4440	7240

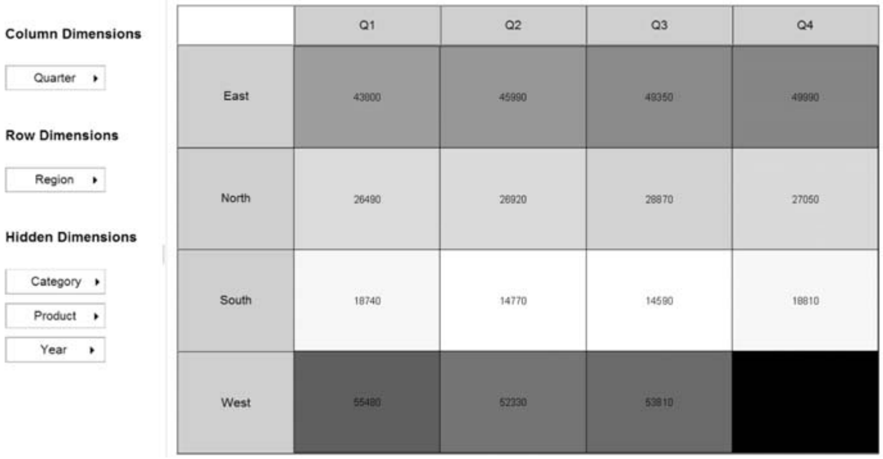
Figure 7: Other possible thumbnail visualizations (not yet implemented in prototype).



		2006	2007	2008
Business	East	40610	58080	69350
	North	24700	23350	26770
	South	17330	14500	12770
	West	30740	56740	118200
Residential	East	8910	6080	6100
	North	13330	13650	7530
	South	7310	8810	6190
	West	13110	4440	7240

		2006	2007	2008
Business	East	40610	58080	69350
	North	24700	23350	26770
	South	17330	14500	12770
	West	30740	56740	118200
Residential	East	8910	6080	6100
	North	13330	13650	7530
	South	7310	8810	6190
	West	13110	4440	7240

Figure 8: Examples of graphics for showing associations between two hidden dimensions in individual cells (not yet implemented in prototype.)



	Q1	Q2	Q3	Q4
East	43800	45990	49350	49990
North	26490	26920	28870	27050
South	18740	14770	14590	18810
West	55480	52330	53810	

Figure 9: Grayscale background proportional to number of sales (not yet implemented in prototype.)

4 Implementation

An important goal of our project was to design an interface that would be device independent. While desktops and workstations will undoubtedly continue to be the platforms of choice for querying and visualizing large databases, other devices like laptops, netbooks, tablets, and smarphones will increasingly provide a window to the database, particularly for casual users. Thus our interface is browser-based and platform agnostic.

The interface is implemented using a client/server structure. Animations and previews are computed on the client side in JavaScript. The client sends SQL requests to the server. The Java Servlet on the server accepts the request, executes the SQL, repacks the ResultSet into JSON and sends the results back to the client. The client parses the results and displays them in accordingly. Note that our approach supports all relational databases that implement a SQL interface (e.g., DB2, Oracle, MS SQL Server). Minor differences in SQL dialects are easily accommodated.

5 Future Development

Our goal is to make tabular views of a database easier to understand and navigate, particularly for novice users. This is accomplished by enriching the user interaction using a variety of visualizations and methods of restructuring the tables to reveal relationships. The intention is to allow the data analyst to see ahead before generating new tabular views, thus helping to avoid blind alleys.

The current implementation is intended as a proof-of-concept. There are many areas where we plan to make improvements and enhancements. These include the aesthetics of the interface; the quality and utility of the animations; the use of brushing, linking, and highlighting; the addition of statistical functions and heuristics, including the pre-computation of statistics that are required to produce some visualizations; and, also, tools to encourage collaborative analysis.

Increasingly, smartphones, tablets, and similar devices will be used by data analysts. That is one important reason why our approach is browser-based. Thus the interface is a natural candidate for multitouch input and future versions will incorporate this enhancement.

The design and implementation of an interface is only the first step; ease of use and effectiveness in discovering relationships of interest is essential. Usability testing will be needed to determine which novel selection and display features should be retained. Usability testing will involve assessments of speed, accuracy, “discoverability”, and user satisfaction [c.f., 18,19].

Acknowledgements

This research was supported by a CAS Faculty Award to Ian Spence and by and by an NSERC Discovery Grant to Michael McGuffin.

About the Authors

Jiang Du (jdu@cs.toronto.edu) is a graduate student in the Department of Computer Science, University of Toronto. His research is in heterogeneous databases, data mining and data warehousing. Ian Spence (ian.spence@utoronto.ca) is a Professor in the Department of Psychology, University of Toronto. His research interests include cognition, perception, enginering psychology, and data visualization. Michael J. McGuffin (mchael.mcguffin@etsmtl.ca) is an Assistant Professor in the Department of Software and IT Engineering of the École de technologie supérieure in Montreal, where he conducts research on user interfaces and information visualization.

References

[1] S. Sarawagi, R. Agrawal, N. Megiddo. Discovery-driven Exploration of OLAP Data Cubes. *Advances in Database Technology - EDBT'98, Lecture Notes in Computer Science*, 1377, 168-182, 1998.

[2] E. Suzuki. Data Mining Methods for Discovering Interesting Exceptions from an Unsupervised Table. *Journal of Universal Computer Science*, 12, 627-653, 2006.

[3] L.Q. Geng, H.J. Hamilton. Interestingness Measures for Data Mining: A Survey. *ACM Computing Surveys*, 38, 1-32, 2006.

[4] C. Joslyn, J. Burke, T. Critchlow, N. Hengartner, E. Hogan. View Discovery in OLAP Databases through Statistical Combinatorial Optimization. *Scientific and Statistical Database Management, Proceedings, Lecture Notes in Computer Science*, 5566, 37-55, 2009.

[5] http://www.tableausoftware.com/

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

[7] J.D. Mackinlay, P. Hanrahan, C. Stolte. Show Me: Automatic Presentation for Visual Analysis. *IEEE Transactions on Visualization and Computer Graphics (TVCG),* 13, 1137-1144, 2007.

[8] E.R. Tufte. *Envisioning Information*. Graphics Press, 1990.

[9] L. Wroblewski. Small Multiples within a User Interface. http://www.uxmatters.com/mt/archives/2005/12/small-multiples-within-a-user-interface.php

[10] D.D. Woods. Visual Momentum: A Concept to Improve the Cognitive Coupling of Person and Computer. *International Journal of Man-Machine Studies*, 21, 229-244, 1984.

[11] L. Bartram. Can Motion Increase User Interface Bandwidth? *Proceedings of IEEE Conference on Systems, Man, and Cybernetics*, 1686-1692. 1997

[12] C. Plaisant, J. Grosjean, B.B. Bederson. SpaceTree: Supporting Exploration in Large Node Link Tree, Design Evolution and Empirical Evaluation. *Proc. IEEE Symposium on Information Visualization (InfoVis)*, 57-64, 2002.

[13] P. Pirolli, S.K. Card. Information Foraging. *Psychological Review*, 106, 643-675, 1999.

[14] Pirolli, P.L.T. *Information Foraging Theory: Adaptive Interaction with Information*. Oxford University Press, 2007

[15] N. Elmqvist, T.-N. Do, H. Goodell, N. Henry, J.-D. Fekete. ZAME: Interactive Large-Scale Graph Visualization. *Proceedings of the IEEE Pacific Visualization Symposium*, 215-222, 2008.

[16] N. Elmqvist, P. Dragicevic, J.-D. Fekete. Rolling the Dice: Multidimensional Visual Exploration using Scatterplot Matrix Navigation. *IEEE Transactions on Visualization and Computer Graphics (Proc. InfoVis 2008)*, 1141-1148, 2008. https://engineering.purdue.edu/~elm/projects/scatterdice/scatterdice.pdf

[17] M. Friendly. Corrgrams: Exploratory Displays for Correlation Matrices. The American Statistician, 56, 316-324, 2002.

[18] I. Spence. Visual Psychophysics of Simple Graphical Elements. *Journal of Experimental Psychology: Human Performance and Perception*, 16, 683-692, 1990.

[19] I. Spence, S. Lewandowsky. Displaying Proportions and Percentages. *Applied Cognitive Psychology*, 5, 61-77, 1991.