TECHNICAL
RESEARCH
REPORT

SYSTEMS
RESEARCH
CENTER

*Supported by the
National Science Foundation
Engineering Research Center
Program (NSFD CD 8803012),
Industry and the University*

Adapting Treemaps to Stock Portfolio Visualization

by W-A. Jungmeister and D. Turo

TR 92-120

CAR-TR-648
CS-TR-2996
SRC-TR-92-120

Sept. 1992

ADAPTING TREEMAPS TO STOCK PORTFOLIO VISUALIZATION

Walter-Alexander Jungmeister*f*, David Turo†

Human-Computer Interaction Laboratory
Institute for Systems Research†

University of Maryland
College Park, MD 20742, USA
turo@cs.umd.edu

Union Bank of Switzerland, Dept. BERG*f*
P.O.B., CH-8021, Zürich, Switzerland
Alexander.Jungmeister@ZH007.UBS.UBS.CH

ABSTRACT

Treemap visualization techniques are extended and applied to stock market portfolios via a prototype application. Designed to facilitate financial decision-making, the prototype provides an overview of large amounts of hierarchical financial data and allows users to alter aspects of the visual display dynamically. Treemap concepts are illustrated via examples which address common portfolio management needs.

For single copies
or a list of all HCIL
technical reports
please write to:

Teresa Casey
Human-Computer
Interaction Laboratory
A.V. Williams Building
University of Maryland
College Park MD 20742

email tcasey@cs.umd.edu

ADAPTING TREEMAPS TO STOCK PORTFOLIO VISUALIZATION

Walter-Alexander Jungmeister

Union Bank of Switzerland, Dept. BERG
P.O.B., CH-8021, Zürich, Switzerland

Alexander.Jungmeister@ZH007.UBS.UBS.CH

David Turo

Human-Computer Interaction Laboratory
University of Maryland
College Park, MD 20742, USA
turo@cs.umd.edu

ABSTRACT

Treemap visualization techniques are extended and applied to stock market portfolios via a prototype application. Designed to facilitate financial decision-making, the prototype provides an overview of large amounts of hierarchical financial data and allows users to alter aspects of the visual display dynamically. Treemap concepts are illustrated via examples which address common portfolio management needs.

KEYWORDS: information visualization, hierarchy, treemaps, fisheye views, finance

INTRODUCTION

The electronic age has brought with it access to major stock exchanges throughout the world from desktop computer terminals. Financial advisors and portfolio managers have at their fingertips the information to maintain and trade larger portfolios more efficiently than in the past. Whenever more information is available, the requirement for effective handling and interpretation of this information is key.

Conventional mechanisms for handling large amounts of stock exchange information include traditional electronic ticker tape displays as well as full-screen software applications that take advantage of the graphical capabilities of modern PCs and workstations. These have proven to be quite satisfactory for various portfolio management tasks such as number calculation and small graphical presentations, but they are less adept at synthesizing and visualizing the large volumes of portfolio information which accumulate over even one trading day.

The *treemap* is a visualization technique for displaying large amounts of hierarchically structured information [2]. We believe the field of portfolio management is ideally suited for treemap visualization given its dual hierarchies (a corporate management structure and portfolio structure).

It should be noted that we will address how the visualization of this type of information can be useful to the portfolio manager for *identifying areas of interest in a global context*; additional tools for interpreting and acting upon this information are necessary for a complete portfolio management system.

PRIOR TREEMAP RESEARCH

Research in treemap design has occurred over the past two years in the Human-Computer Interaction Laboratory at the University of Maryland. The treemap is a method for recursively subdividing a 2-D rectangular space to visualize large hierarchies that are cumbersome to display using traditional approaches. The treemap algorithm is space-filling and dubbed "slice-and-dice" -- it slices the initial rectangle in one direction (either horizontally or vertically) to display the first level of hierarchy, then "dices" the resultant rectangles in the other direction to display the second level. Successive levels are formed by repeating this process. Figure 2 is an example of a treemap produced by this algorithm.

Treemap design issues are highlighted in [7]; a sample treemap application is featured in [2] and experimental results are discussed in [8]. Several issues arose with the experimental usage of treemaps; these issues are discussed in a later section and addressed in the prototype.

PORTFOLIO MANAGEMENT TODAY

Two Hierarchies: Management and Portfolio

The complexity of the corporate organization is proportional to the client base of the company (which could be a bank, investment house, brokerage firm, or a smaller local business): large companies will naturally have large organizational hierarchies to cover different geographic regions and different types of markets; smaller companies need less of a support structure to serve their client base. Since the treemap's effectiveness arises from the visualization of large amounts of information, we are concerned with the larger organizational hierarchies found in banks or brokerage houses.

The lower levels of a typical corporate hierarchy usually contain the portfolio managers, group leaders, and section

- performance relative to a fictional "purchase price" for each stock
- performance relative to standard market indices

A treemap display is the focus of the FolioMap application (Figure 2). The corporate and portfolio hierarchies are combined on the display by placing the portfolios directly beneath the individual portfolio managers.

In Figure 2, each colored rectangle represents a single stock. The underlying hierarchy is represented by gray-scale rectangles successively offset based upon hierarchical level. The user is able to manipulate the diagram to provide different "views" of the data and selectively emphasize or de-emphasize objects on the screen. This can be done by modifying one of the following four items:

- **Size Attribute**: Value which determines the area of an individual rectangle; this attribute indicates something of interest to the user.
- **Color Attribute:** Value or range which maps to a particular color. This value can be viewed simultaneously with the size attribute to convey additional information.
- **Offset Quantity:** The amount of pixel space used to offset children from their parents on the diagram. Offsets are produced from the left and top of the parent rectangle.
- **Exaggeration Level:** Amount that differences between sizes are exaggerated to overcome aspect-ratio problems. Users choose from Low, Medium or High exaggeration on the diagram.

The Exaggeration Level is new to this prototype and is discussed by means of example in a later section. Also new to this prototype is a *dynamic color key* (in the upper right corner of the screen) that changes to reflect the color and ranges selected for the chosen color attribute. This flexibility creates an extremely customizable environment, as is demonstrated in the three examples below.

Additional parts of FolioMap are a *feedback area*, in the lower left of the window, which provides one-way textual feedback on the current selected node, and an "animation slider," which animates the diagram over time.

FolioMap is not designed to be a complete system. It allows users to facilitate the process of sifting through large amounts of data by presenting that data in a convenient and customizable way; once the user has identified an area of interest, more traditional systems and approaches for data analysis can be employed quite effectively through "hooks" into other systems.

USING THE PROTOTYPE

The same data can appear radically different on a treemap depending upon the size and color attributes the user specifies. The following examples all use data from March 6, 1992. Each example addresses a specific need.

Example 1: Determining Buy/Sell Conditions

Configuration. Figure 2's configuration helps to determine appropriate times to buy and sell stocks. The size of each rectangle is determined by the stock's value; color is determined by the percent change in value from the preceding trading day. The hierarchical offset is set to 12 pixels.

Using this configuration, stock price changes are seen in the context of value. Stocks with higher values have the potential to cause increased profits or losses and are consequently emphasized here via large areas. Lower-valued stocks are not realized on the diagram.

The color range represents the level of stock stability. Here, gains greater than 4% of value or losses greater than or equal to 5% of value over one trading day determine conditions which exceed the set stability level and map to dark green and red colors, respectively. Other percentage values within the allowable range are assigned varying shades of green and red.

Results. From this diagram, Cabot Corporation's stock is clearly a winner, increasing in value by 4.56% (up $1.75 to $40.12) for the day (Cabot is selected in the lower left corner of the figure; a white box surrounds its rectangle). Vencor Incorporated stock is colored dark red, reflecting a loss of 5.56% (down $1.75 to $29.75). At a glance, then, problem areas in a global context can be determined.

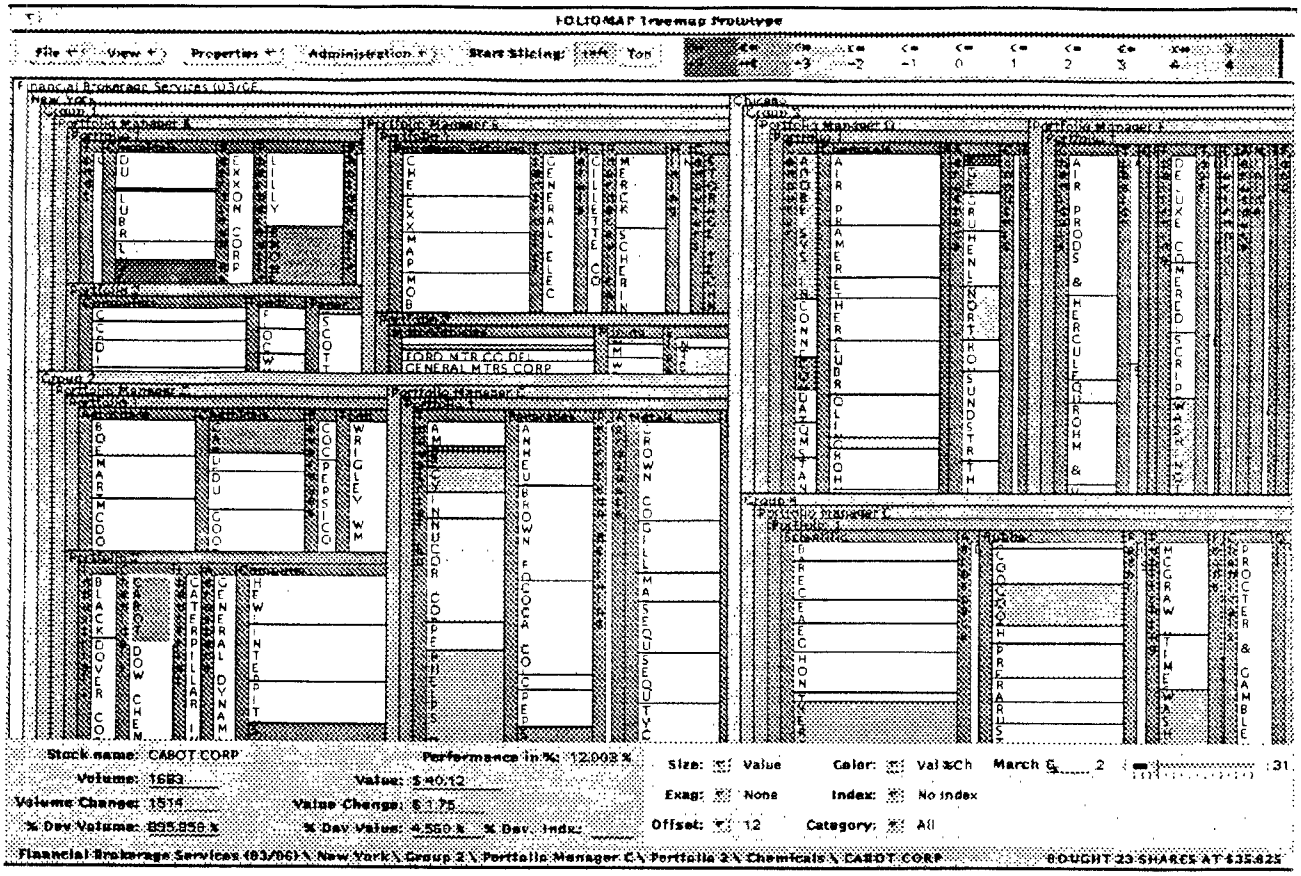
This example uses 4% and -5% as limits for stock stability. The ranges and colors are modifiable, creating a diagram which accurately reflects the portfolio manager's notion of risk. Blinking colors may be used to identify changes deemed critical, while a gray-scale may be used to address monochrome monitors or color deficiencies.

Comments. A deficiency noted above is the fact that stocks with lower values are not displayed on the diagram, specifically because the partitioning algorithm allocates space based on the size attribute and there is a finite amount of space to allocate. In this example, though, these stocks were deemed "unimportant" to the user.

If a user had a need to view all stocks, FolioMap provides two possibilities:

- the size attribute can be set to "Unit Count", which displays all stocks with equal areas
- the diagram may be "zoomed" to view a particular rectangle and its descendants at full-screen size

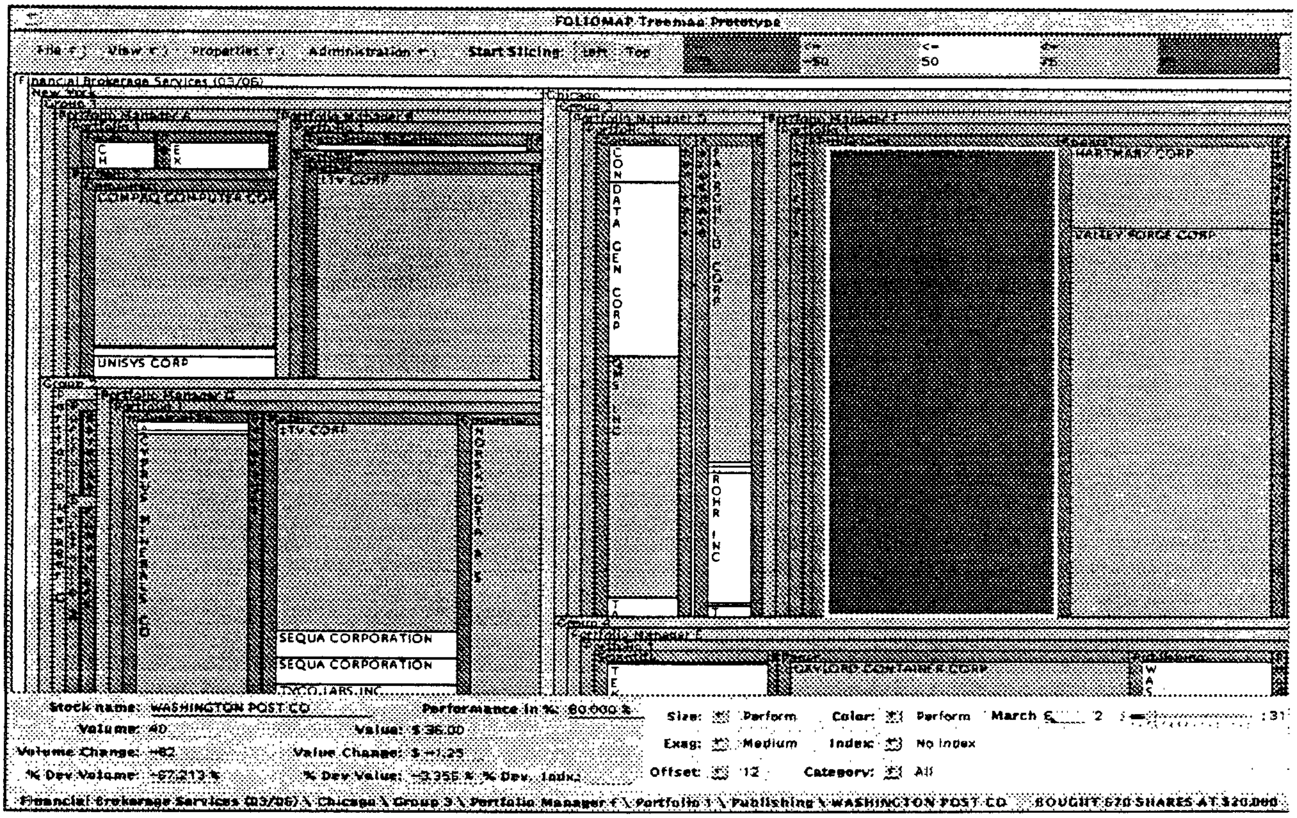
Another concern in the example is that Cabot Corporation's area was small, making its dark green color somewhat



Stock name: CABOT CORP
Performance in %: 12.003 %
Volume: 1683
Value: $ 40.12
Volume Change: 1514
Value Change: $ 1.75
% Dev Volume: 895.858 %
% Dev Value: 4.560 %
% Dev. Indx:
Size: Value
Color: Val %Ch
March 6 2 :31
Exag: None
Index: No index
Offset: 12
Category: All

Financial Brokerage Services (03/06) \ New York \ Group 2 \ Portfolio Manager C \ Portfolio 2 \ Chemicals \ CABOT CORP BOUGHT 23 SHARES AT $35.825

Figure 2: Buy/sell conditions are determined by the above treemap. Value is proportional to area, and Value % Change from preceding day is indicated by color. Cabot's small area indicates low value relative to other stocks; its dark green color indicates a greater than 4% increase from March 5.



Stock name: WASHINGTON POST CO
Performance in %: 60.000 %
Volume: 40
Value: $ 36.00
Volume Change: +82
Value Change: $ +1.25
% Dev Volume: -67.213 %
% Dev Value: -3.355 %
% Dev. Indx:
Size: Perform
Color: Perform
March 6 2 :31
Exag: Medium
Index: No Index
Offset: 12
Category: All

Financial Brokerage Services (03/06) \ Chicago \ Group 3 \ Portfolio Manager F \ Portfolio 3 \ Publishing \ WASHINGTON POST CO BOUGHT 670 SHARES AT $20.000

Figure 3: This treemap indicates portfolio performance. Large areas indicate large differences from the original purchase price; color indicates a positive or negative swing. The Washington Post Co. has a positive difference from its original price.

difficult to see on a cluttered diagram. This is to be expected with the current settings, as the size attribute indicates a primary interest in Value, and Cabot's value was small compared with the rest of the stocks.

If a user was primarily interested in stocks which have *changed* in value and not in the *actual* value of the stocks, setting the size attribute to "Value % Change" would be the correct choice. Area would then reflect *absolute* percentage changes from yesterday's value (attributes which have negative values are mapped on an absolute scale). Cabot's area would be one of the largest on the diagram, making its identification a simple matter. Color, set to the same attribute in this case, would be used to indicate whether the change was positive or negative.

Example 2: Determining Portfolio Performance

Configuration. Performance of a stock is always tied to the price of the stock when purchased. Figure 3 is a representation of the same data as Figure 2, with "performance percentage relative to purchase price" as both the size and color attributes. (This attribute is simply the current price minus the purchase price, the difference divided by the purchase price.)

The color key chosen to represent performance differs from the "Value % Change" key from Figure 2 in three respects: the number of ranges has decreased (from 11 to 5), the colors are different, and the range values are different. In this case, we are interested only in stocks which have changed by 50% above or below the original purchase price. This is reflected by using white for the range -50% to 50%. A change level of 75% determines the end points and is given the darkest red and blue colors.

Another change is the Exaggeration Level: here, it is set to "Medium," which means size differences are exaggerated via a mathematical function. This effect serves to highlight the size attribute, similar to a fisheye effect [1].

Results. Given that color is used only for large-scale percentage changes, the diagram clearly indicates that the Washington Post Company's stock (selected on Figure 3) is performing very well based upon its purchase price of $20.00 a share (its March 6 value is $36.00).

LTV Corporation at first may appear to indicate an anomaly: it is displayed twice, once colored light blue (in the 50% to 75% range) and once colored light red (in the -50% to -75% range). This is due to the fact that the stock was purchased for two different portfolios *at different prices* (and obviously at different times). The box colored red is down from its or:ginal price, and the box colored blue is up from its original price.

Comments. Many of the stocks performing poorly relative to purchase price have rectangles with similar areas, making comparison difficult. This can be countered by increasing the exaggeration level to High, which increases the size distinctions between the rectangles at the expense of decreasing the number of rectangles displayed.

Example 3: Determining Market Activity

Configuration. Market activity for a stock can be interpreted as the level of interest in the stock as determined by the number of shares traded during a particular day. Returning to March 6, Figure 4 displays the same data as Figures 2 and 3, with size changed to Volume (number of shares traded in a particular day) and color changed to "Volume % Change," which indicates the percentage change in volume from the preceding trading day.

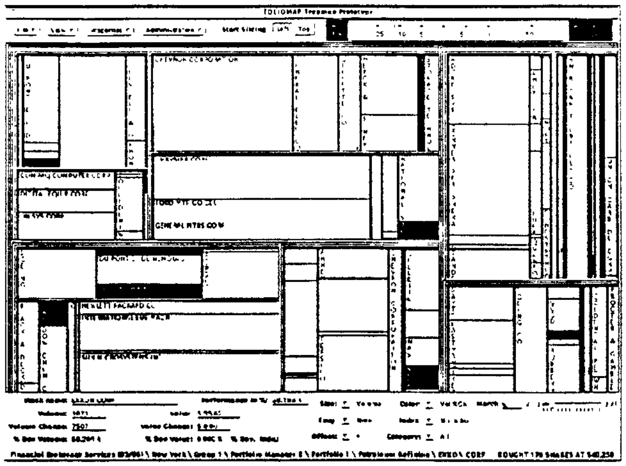


Figure 4: Market Activity Treemap, Size = Volume, Color = Volume % Change

The offset has been changed to 4 pixels to reduce emphasis on the hierarchy and give more space to the leaf nodes. Exaggeration effects have been turned off. As the volume of shares traded over a day tends to fluctuate more than the value, the color key is again modified for "Volume % Change" to reflect smaller ranges over 13 intervals.

Results. The diagram here contains information ("clues") that would lead the user to other systems for further investigation. One item worth investigating might be the cause of the high volume of automobile stocks at the top-center of the diagram. Another item would be ascertaining why a large box is represented by a "positive" color, which indicates the stock is being heavily traded and the number of shares traded has increased from yesterday.

The wise portfolio manager would use these clues to uncover the underlying market conditions (such as a company's quarterly announcement of profits or the federal government's release of economy figures) that cause these volume fluctuations.

Comments. This static picture does not provide a way to determine trends. However, the animation slider may be used to display data from the following days in March to see if the volume level and change represent a trend.

Additional Concepts

Hierarchy Display Level. Figure 5 uses Figure 2's configuration with the display shown down to the Portfolio Manager level. This reduces detail, but is useful for the upper-level manager to gain an overview of performance as well as context for the display of lower hierarchical levels. Aggregate data is displayed in the text area for Portfolio Manager "D" in the figure. The size and color attributes may be altered here as well. (The current FolioMap application does not set colors for interior nodes, however.)

Zooming. Figure 6 uses Figure 2's configuration as well, zoomed in on Portfolio Manager "G"'s rectangle. ("G" has been assigned only one portfolio.) Zooming allows more detail to be shown for cluttered regions. G's 3rd and 4th portfolio industries, for example, appear as thin vertical lines on Figure 2; they have expanded into rectangles on Figure 6.

Taken as a whole, FolioMap provides the user with a visual display of a large hierarchical data space that allows quick location of areas of interest. Mental information routing is thus optimized: the initial impact of large amounts of data is minimized to allow the user to concentrate on particular details within the data.

TREEMAP ISSUES

The prior treemap experiments and prototypes surfaced some issues which were addressed by the FolioMap prototype.

User Customization

Of primary importance in any treemap implementation is user control. Providing the user control over the size and color attributes, offset and exaggeration levels, color ranges and the colors themselves (as FolioMap does) creates a customizable environment that supports information retrieval and the user's sense of what is important.

Color

Color is used in treemaps because it is a powerful tool for conveying information -- perhaps even more powerful than shape, as suggested in [3]. Along with color's benefits come several liabilities: combinations of colors and high saturation levels can create optical illusions and jarring effects on screen [6]; color deficiencies also need to be taken into account.

For FolioMap a palette of 24 colors (from six distinct hues) was provided for use in color ranges. Each hue was divided into four separate colors by altering the brightness at fixed intervals while keeping saturation at a constant, low level to avoid undesirable color side effects.

The colors chosen for the examples in this paper all resemble a spectrum sequence as that sequence type was found to be more effective than other types, such as a saturation sequence or gray-scale [9]. The gray-scale representing the internal hierarchy emphasizes leaf node colors. Users are not limited to these choices and may select colors differently depending upon their individual needs.

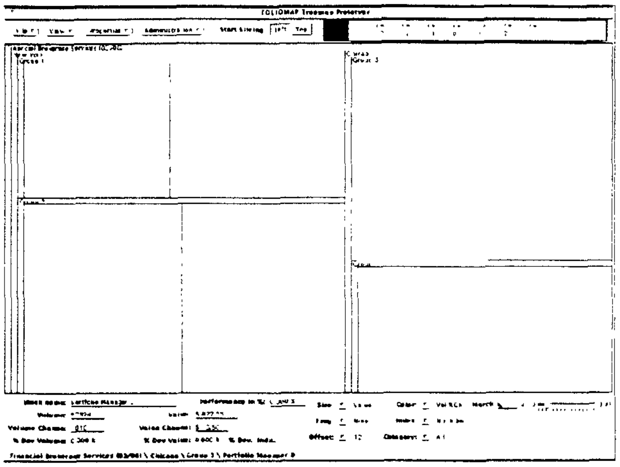


Figure 5: Treemap displayed to Portfolio Manager Level

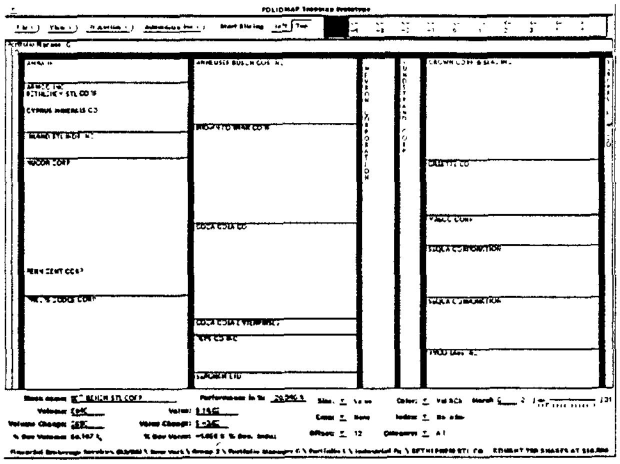


Figure 6: Treemap zoomed in on Portfolio Manager "G"

Exaggeration for Aspect-Ratio Problems

Subjects in a treemap experiment discussed in [8] were observed to incorrectly compare the areas of similar rectangles with differing heights and widths. This aspect-ratio problem prevents proper size ranking of items.

To overcome these limitations, the size quantity was filtered through a mathematical power function to exaggerate the differences between rectangles. For user convenience, 3 functions were mapped to exaggeration levels of "Low", "Medium", and "High".

Emphasizing the size attribute leads to greater screen area for nodes with larger size values and little or no screen area for nodes with smaller values, much like a fisheye view [1].

In this case, however, there are multiple points of interest rather than the single focus of a standard fisheye diagram. Given that the size attribute is a reflection of interest in a node, the disappearance of small nodes during exaggeration should be of little consequence. Additionally, high exaggeration may lead to quicker judgments and decisions about the data as "useless" nodes are eliminated from the diagram.

FUTURE RESEARCH

Future research will involve an additional experiment to test the effectiveness of the exaggeration and customization concepts developed for FolioMap.

The following items are research objectives:

- **Comparison Algorithms** - Promoting comparison on treemaps by creating a fisheye view around a particular node or nodes
- **Sorting** - Aids in comparison by sorting nodes underneath a parent based on a selected attribute
- **Information Overload** - Techniques to counteract the information density presented to the user
- **New Domains** - Additional data sources such as budgetary or census data.

CONCLUSION

FolioMap has evolved the treemap concept into a customizable approach for visualizing large hierarchies. A user can visually manipulate these hierarchies in a wide variety of ways, creating diagrams which emphasize items an individual user considers important while retaining the global context of those items. Treemaps can thus be used to find the proverbial "needle in the haystack" and provide visual insight into the data that would not have been possible otherwise.

ACKNOWLEDGMENTS

We would like to acknowledge the support of the members of the Human Computer Interaction Lab (HCIL, University of Maryland, College Park), especially Brian Johnson and Ben Shneiderman, whose work with treemaps over the past two years form the basis of this paper.

The FolioMap application, developed at HCIL, was partly funded by the Human Factors Group of Union Bank of Switzerland, Zürich.

REFERENCES

1. Furnas, G. Generalized fisheye views, in Proc. CHI'86 Human Factors in Computing Systems (Boston, April 13-17, 1986), ACM Press, pp. 16-23.

2. Johnson, B. and Shneiderman, B. Tree-Maps: A Space-Filling Approach to the Visualization of Hierarchical Information Structures, in Proc. 2nd IEEE Conference on Visualization (San Diego, Oct. 22-25, 1991), IEEE Computer Society Press, pp. 284 - 291.

3. MacDonald, L. Using colour effectively in displays for human-computer interface. Displays (June 1990), 129-142.

4. Maginn, J.L. and Tuttle, D.L. (Eds.). Managing Investment Portfolios: a Dynamic Process. Boston/New York, 1983.

5. Reilly, F.K. Investment Analysis and Portfolio Management. San Francisco, 1985.

6. Rice, J.F. Ten Rules for color coding. Information Display (March 1991), 12-14.

7. Shneiderman, B. Tree Visualization with Tree-maps: A 2-D space-filling approach. ACM Transactions on Graphics, January 1992.

8. Turo, D. and Johnson, B. Improving the Visualization of Hierarchies with Treemaps: Design Issues and Experimentation, to appear in Proc. 3rd IEEE Conference on Visualization (Boston, Oct. 19-23, 1992).

9. Ware, C. Color Sequences for Univariate Maps: Theory, Experiments, and Principles. IEEE Computer Graphics and Animation (September 1988), 41-49.