

Treemap Presentation as a Corporate Dashboard

Larry Day & Richard Dickinson
Corporate Performance Measures
BNSF[®] Railway

Treemap data presentation as an exception-based reporting tool for corporate dashboards has many advantages over tabular data presentation. These advantages include a more compact information display, simultaneous presentation of multiple data dimensions (such as size and color), and, as a direct result, faster identification of problem areas for further investigation. This paper will discuss treemap display technology, how it was implemented by BNSF Railway, and the role SAS[®] has played in this implementation.

INTRODUCTION

Treemaps were conceived in the early 1990s by Ben Shneiderman and developed at the University of Maryland Human-Computer Interaction Lab (HCIL) [1]. Treemap visualization represents a very powerful means of displaying hierarchical information that allows rapid assimilation and interpretation of voluminous data in size-constrained environments. In 2003 our group began exploring treemap technology as a vehicle for displaying departmental performance measures. The utility of treemap display for exception-based reporting was immediately obvious. Our efforts led to the development of CORR (Condition of the Railroad). Powered by SAS, this web-based application currently displays 43 marketing department measures. Extensive drill-in capability allows each measure to be broken-down in many ways, such as by customer, personnel reporting hierarchy, geography, equipment type, and commodity. Literally billions of data views are possible.

WHAT IS A TREEMAP?

In its most basic form a treemap is a series of rectangles displayed in a defined space with the size of the rectangles representing some dimension of interest (such as total number of units transported year-to-date). Additionally, another dimension can be represented by the color of the rectangles indicating an increase or decrease in units transported compared to the same timeframe in the previous year. The rectangles themselves can be any measureable entity such as files on a computer's hard drive, Fortune 500 companies, or, as displayed in Figure 1, commodity groups being transported.



Figure 1. Basic treemap visualization.

In years past, discussion of treemap technology was relatively hard to find. This situation has changed dramatically in the last five years. A Google search of “treemap” will return a mass of hits, attesting to the increased research and interest in treemap visualization. Version 7 of JMP® contains a treemap view as part of its graphics module.

TREEMAPS AT BNSF RAILWAY

As stated above, our group was interested in creating a dashboard application displaying departmental measures as part of an exception-based reporting tool. We wanted it to be intuitive, easy to navigate, and to quickly draw the user’s attention to potential problem areas. The solution developed was to present the dashboard information using two different forms of treemap display: *Overview* treemaps and *Detail* treemaps.

Overview Treemaps

As can be seen in Figure 2, the Overview treemap serves to display all dashboard measures in a single display format. No attempt is made to size these rectangles by another dimension. The purpose of the Overview treemap is simply to display all measures against some yardstick (in this case, Year-to-date vs Last Year at the overall System level). Users can mouse-over any of the 43 measures and a gray box will appear showing statistics associated with that measure. In this case the gray information box indicates that Year-to-date Units are 1.99% less than that for the same timeframe last year (the actual Units values have been “x’d-out” due to the sensitive nature of the data). At this point the user can manipulate the dropdown choices to obtain a different view of the Overview treemap, or the user may click on a rectangle and be taken to a *Detail treemap* for that measure.

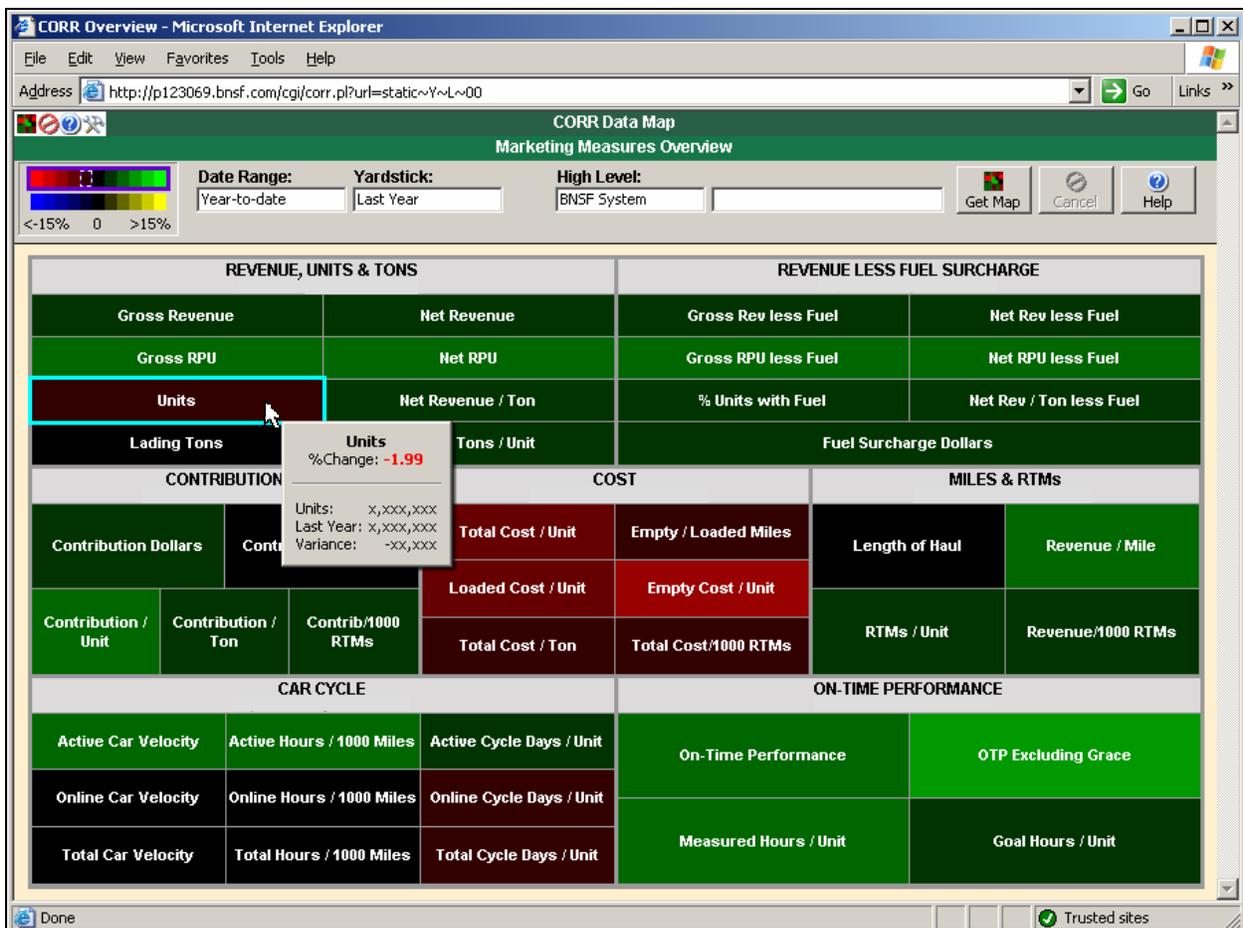


Figure 2. Overview treemap showing all measures at once.

Detail Treemaps

The display shown in Figure 3 is a single-level Detail treemap. CORR is capable of displaying up to three levels at a time, nested within each other. The rectangles represent forecast groups (primarily commodity groupings), the size of which is determined by the sum of units YTD for both this and last year (the size of a rectangle can be interpreted as a function of its "importance" or "impact"). Eleven gradations of color are provided that highlight the magnitude of percent difference in the measure (Units) between the date range (Year-to-date) and yardstick (Last Year). Many date range/yardstick combinations are available. As with the Overview treemaps, mousing-over a rectangle causes a gray information box to appear that contains statistics on the rectangle in question.

There are various algorithms for drawing the rectangles. We chose the squarified algorithm[2]. This method results in rectangles with low aspect ratios (height and width are more nearly equal) making them, in our opinion, easier to see and mouse-over.

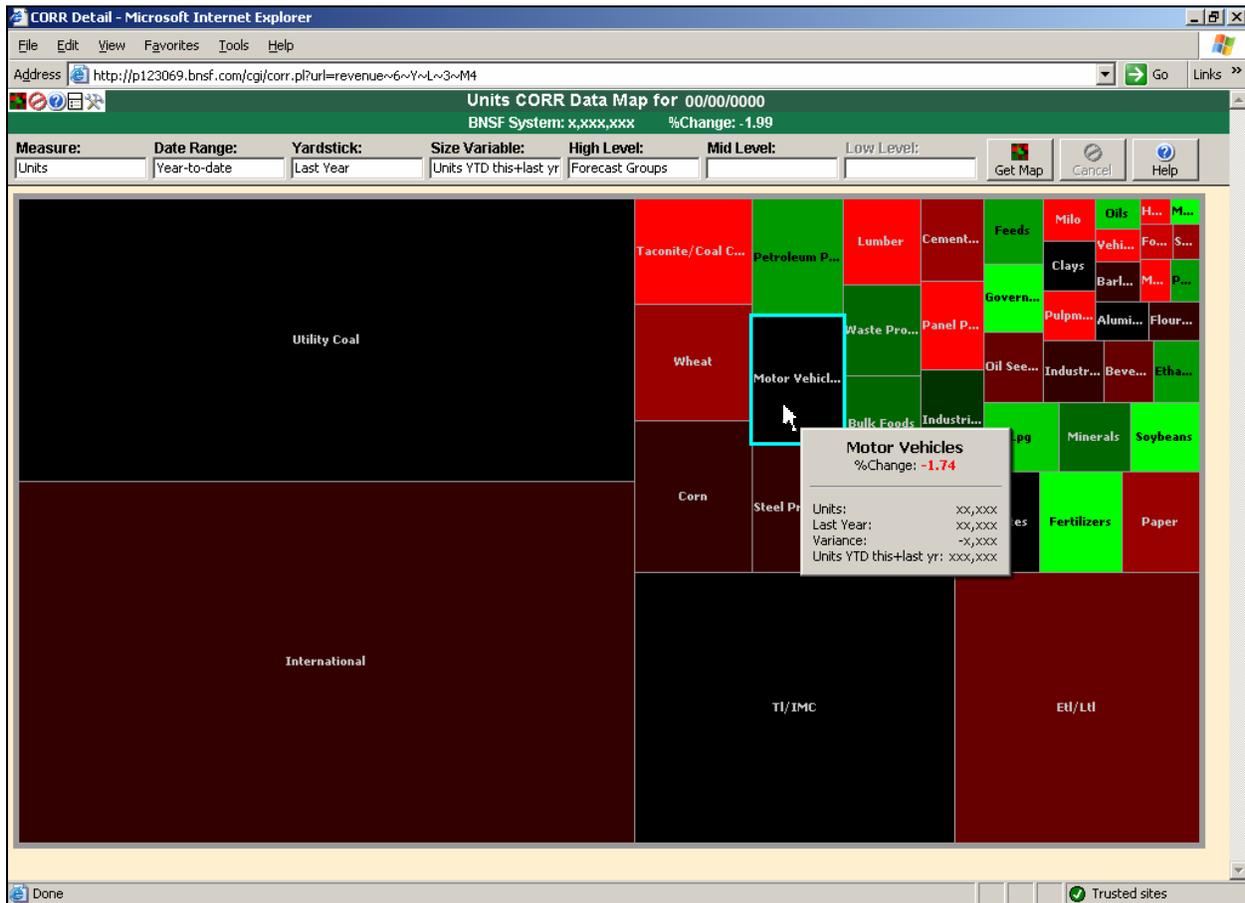


Figure 3. Detail treemap.

NAVIGATING THE CORR UNIVERSE

One of the more unique and powerful features of CORR is that a Detail treemap not only provides an efficient visualization of the data, but that it also provides the *primary mechanism for navigating through the data*. The following scenario will illustrate this.

Imagine you are a manager working for the railway and one of your areas of responsibility is Motor Vehicles. The Detail treemap shown in Figure 3 indicates that units for Motor Vehicles are down 1.74% from the number of units last year. Where is this difference coming from? CORR's drill-in capability (which we call *isolation*) makes the investigation fast and easy.

The first step is to isolate on Motor Vehicles by clicking on that rectangle. The Motor Vehicles rectangle will animate to fill the screen as shown in Figure 4. The next step is to choose what you would like to see within Motor Vehicles. Clicking on the High Level dropdown box presents a list of choices with which to further break-down the isolated rectangle. As can be seen, many choices are available (not all are shown in Figure 4). Depending on what is of interest, Motor Vehicles can be broken-down by, among others, marketing or sales personnel, customers, geography, rail car ownership, and equipment types.

What is being demonstrated here is the ease of navigating within CORR. Single-, double-, and triple-level views are possible simply by choosing from dropdown lists. At any point, the measure, date range, yardstick, or size variable values can be changed. Additionally, a right mouse click on any displayed rectangle presents the user with a choice of report and graphical displays of the data.

Much attention was given to grouping the level choices into an easily recognizable and intuitive hierarchy. This has resulted in an interface that is quickly understood and mastered. Numerous times we have been told by users that even after months away from the CORR system, they have no trouble navigating when reintroduced to it.

Returning to our navigation example, the Motor Vehicles rectangle was isolated and the break-down choices were offered to the user. Suspecting that geography may prove useful in the investigation, the user requests that Motor Vehicles be broken-down by Origin-regions. Once this high level choice is made the user clicks the Get Map button and initiates the query.

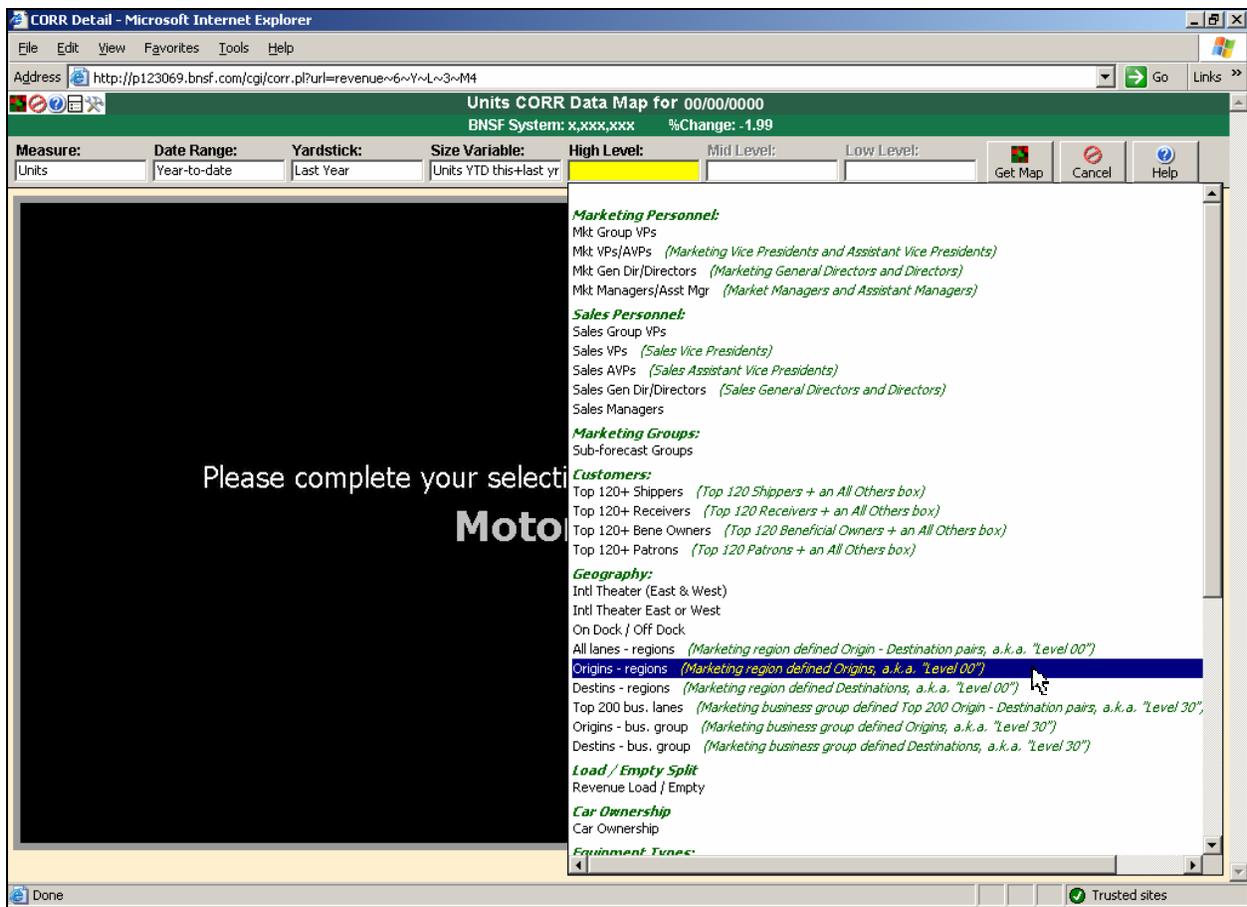


Figure 4. Navigating through CORR using isolation.

Figures 5 and 6 demonstrate further isolation and drill-down into the data. Isolating on Motor Vehicles and displaying Origins—regions immediately shows that the Chicago origin is significantly contributing to the negative disparity in units this year versus last year. Isolating on the Chicago origin and displaying the destination regions (Figure 6) shows which ones are impacting Chicago.

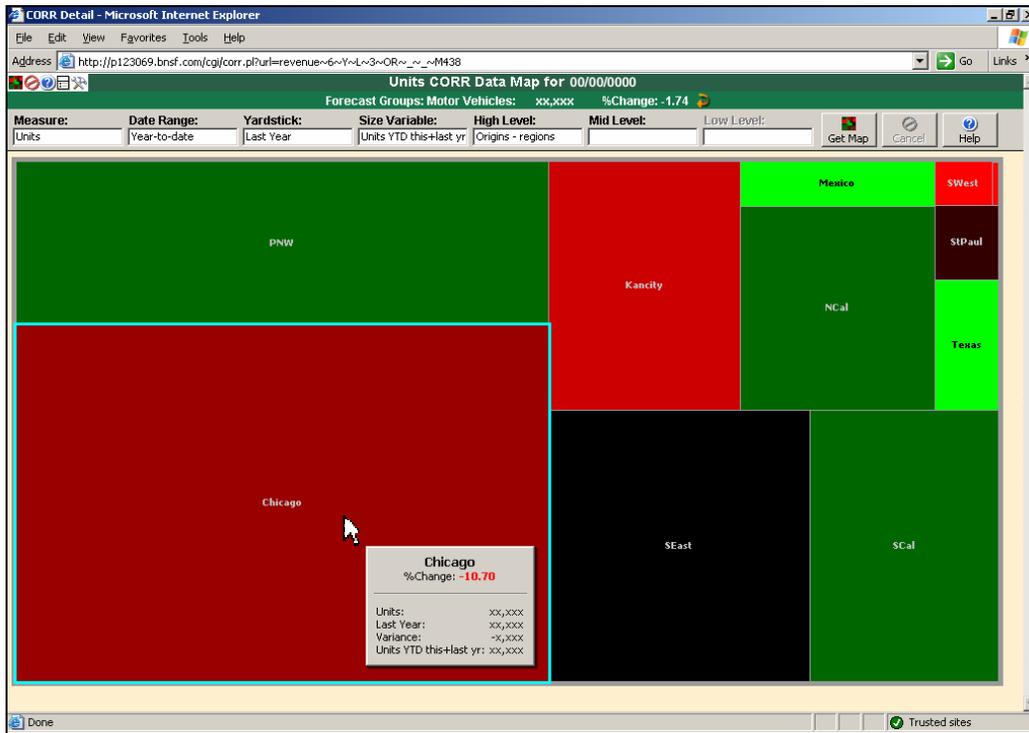


Figure 5. Origins-regions within Motor Vehicles.

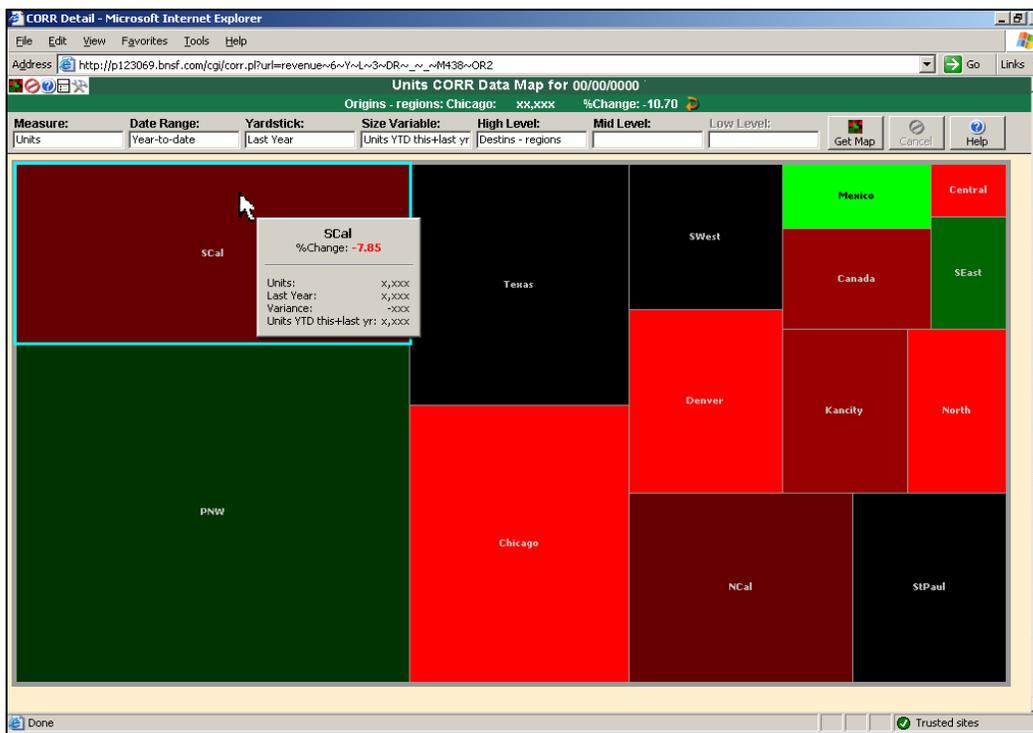


Figure 6. Destins-regions within origin Chicago.

In Figure 6 (previous page) it can be seen that the SCal destination region is experiencing a 7.85% drop in units compared to the same period last year. A look at equipment types may prove enlightening. Isolating on the SCal destination region and choosing BNSF Car Kind (3 char) produces the treemap shown in Figure 7. The M3A Trilevel car kind unit values are more than 15% below the same period year.

There are many things that could account for this result. Costs associated with operating M3A Trilevel car kinds may have increased, lessening their utilization. Route changes may have precluded the use of this car kind. Further investigation down different navigation paths or looking at different measures will provide more information in the process of informed decision-making.

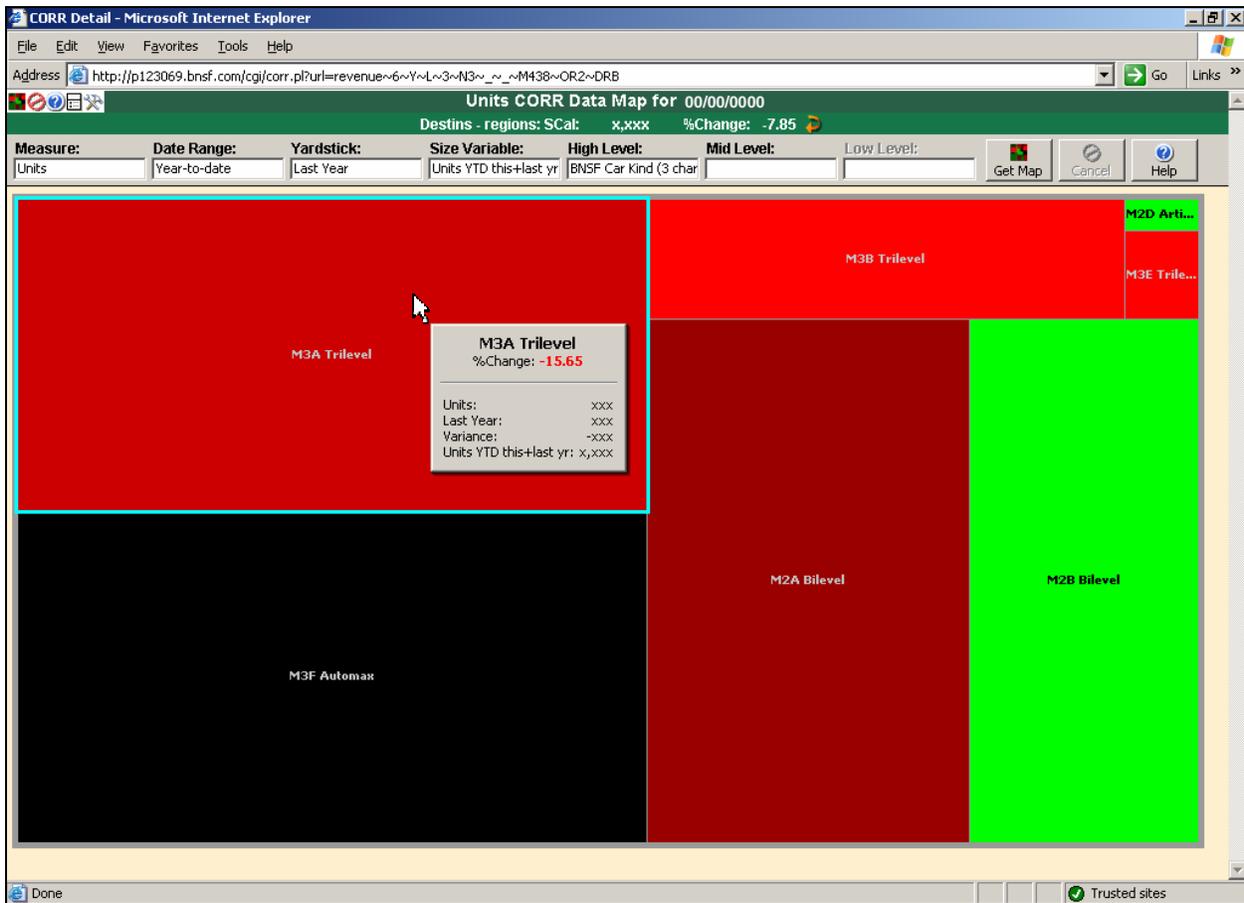


Figure 7. BNSF car kinds within SCal destination region.

Before leaving the discussion of navigating through the CORR system, a short discourse on multi-level Detail treemaps is in order. Figures 5 and 6 illustrated isolating on a specific origin region and then displaying the destination regions within it. As mentioned previously, CORR is able to produce two- and three-level nested treemaps. As such, the information shown in the two figures can be combined into a nested treemap that is requested from a single query.

The display shown in Figure 8 demonstrates just such a two-level nested treemap. In this example, Motor Vehicles has been isolated and CORR has been requested to simultaneously display all of its destination regions nested within all of its origin regions. Origin regions constitute the high level and destination regions constitute the mid, or nested, level. If you look at Figure 5, the rectangles exactly conform to the shape and location of the area bordered by the high level origin regions displayed in the figure below. In a multi-level treemap, each level is visually differentiated by the size of the border (in this example the high level has a 3 pixel border while the mid level has a 1 pixel border). All rectangles are appropriately labeled, with the high level labels displayed slightly larger than the mid level labels.

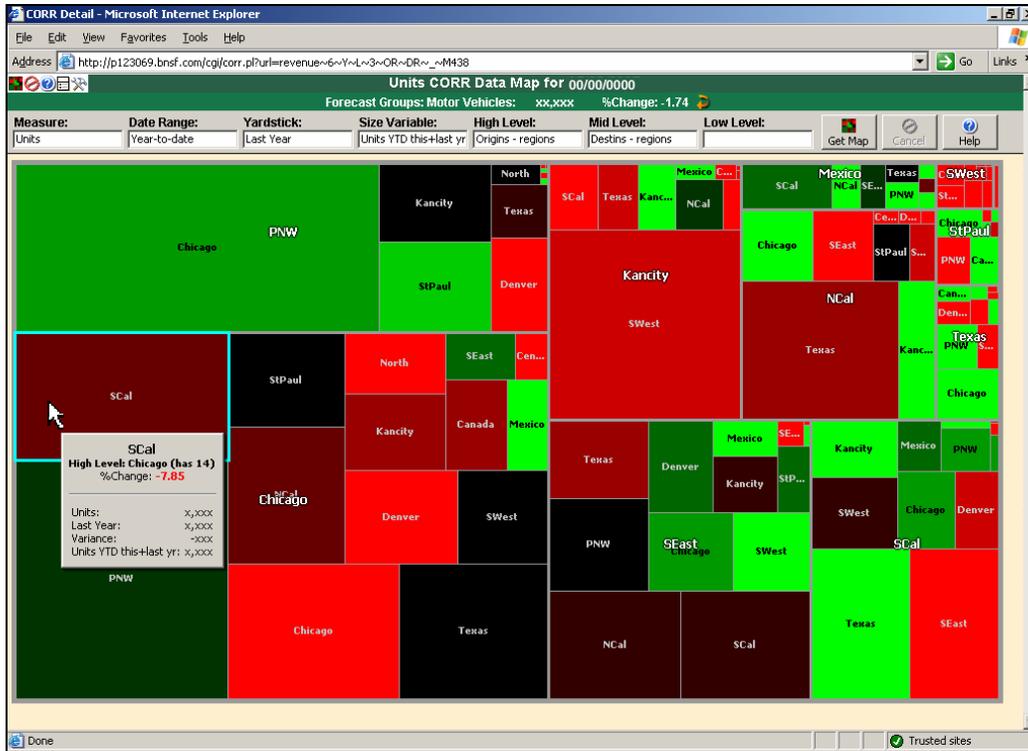


Figure 8. Two-level nested treemap.

HOW DOES CORR WORK?

The CORR system consists of three primary components: a data preparation component, a query generation and data retrieval component, and a user interface component. The first two components are written in SAS. The user interface component is written in JavaScript.

Data Preparation

The basic data for CORR resides in a Teradata® warehouse. Twenty seven months of data are pulled from the warehouse as part of the morning data preparation process. The data in the warehouse is stored at the individual freight bill level. As this level of granularity is beyond our needs the warehouse performs an NWAY grouping of the data, collapsing the data at the level of summarization required by CORR.

Following warehouse data extraction, the data preparation programs come into play. Date ranges are attached to the data (YTD, MTD, etc.), along with yardstick and size variable data. Additionally, files containing data for graphical support are created. It is at this point that any unique business rules are applied to the data. The data is left at the NWAY level of summarization, except for any data that can appear at the overall system level. This system level of summarization obviates the need to pass the entire set of data simply to get numbers at that level. Last, necessary Indexes are generated and the data preparation component is finished. Currently the size of all data (including indexes) is approximately 10 gigabytes.

Query generation / data retrieval / user interface

With data preparation finished, all is in place for the two components that are the heart of CORR. The query generation / data retrieval component and the user interface component work in concert with each other. Looking closely at the URL in Figure 5 you will see the text:

revenue~6~Y~L~3~OR~_~_~M438

This URL text provided the information to the query machinery to retrieve the data that ultimately resulted in the treemap displayed in Figure 5. The text itself was generated by the user's response displayed in

Figure 4. Buried within the HTML are codes associated with dropdown choices as well as the rectangles themselves. The choices made are captured by JavaScript code which gets passed to a Perl script that, in turn, initiates a SAS session that executes a job containing the URL text as a macro variable. The passing of URL text is all performed with CGI environment variables.

The first thing SAS does is parse the URL text that was passed to it, breaking it up into individual tokens. From the structure of the text, SAS determines that a Detail treemap is being requested. In terms of the tokens of interest to this discussion, the token “revenue” tells SAS from which file to read the data, “6” defines the measure as Units, “Y” defines the date range as Year-to-date, the token string “OR-_-_-” indicates that this is a single-level treemap displaying Origins – regions and, finally, “M438” indicates that a commodity forecast group was isolated (M4) and that the isolation was for Motor Vehicles (38).

Obviously, the tokens represent codes of some kind. The query machinery consults a format library that converts the codes into meaningful variable names and values to use in data retrieval WHERE clauses. For example, the URL text example results in the following WHERE clause being generated:

```
WHERE (DATERNGE='Y') and SFG in ('381', '382')
```

Notice that the WHERE clause contains the text: `SFG in ('381', '382')`. The treemap being requested was for the commodity forecast group Motor Vehicles (M438). The data are stored at the commodity *sub-forecast* group level, which is at a more granular level of detail. Retrieving the data for Motor Vehicles requires extraction of the data for the sub-forecast groups 381 and 382. The query machinery knows this (a format library tells it which sub-forecast groups make-up which forecast groups) and builds the WHERE clause accordingly.

Following retrieval of the data, it is necessary that the data be summarized at the level required to satisfy the user’s request. The request was for a treemap displaying all the originating regions for the forecast group Motor Vehicles. The dynamically generated PROC MEANS performs the summarization:

```
PROC MEANS NOPRINT;  
  CLASS ORIG00;  
  TYPES ORIG00;  
  OUTPUT OUT=TREEMAP SUM=;  
RUN;
```

At this point, SAS writes the information (such as rectangle labels, date range, yardstick, and size variable statistics, etc.) to a file which is then served to the user’s browser by a Perl script. JavaScript interprets this information and builds the HTML that results in a treemap, calculating the coordinates of the rectangles.

When the treemap request is for a two- or three-level map involving multiple isolations, the logic for building the query can become much more complex. Additional machinery comes into play for determining the best index to use and, depending on the request, unique business logic may have to be applied. Hopefully, the discussion has shed some light on the role SAS plays in the CORR system.

CONCLUSION

CORR (Condition of the Railroad) was implemented as a treemap-based corporate dashboard for the Marketing department at BNSF Railway. Acceptance within the department has been a very satisfying experience for the authors. Only minimal training has been required for an individual to begin productively navigating through the data. Users of CORR range from summer interns all the way to the highest levels of the organization. This is a testament to the multi-purpose nature of the system, and of treemap visualization technology in general.

Treemap presentation as a corporate dashboard can serve as both an Executive Information System and as an investigative mechanism for data at a very granular level. Treemaps are very supportive of interactive exploration, quickly bringing attention to outliers and exceptions. This aspect of treemaps, along with the robust drill-in and nesting capability built into CORR, encourages contemplation of different

data perspectives and promotes thought experimentation about industry issues. As an added benefit, few presentation technologies will more rapidly confirm or exclude a hypothesis.

The software used to develop CORR had to meet a very demanding specification. The delivered system had to easily handle the voluminous data reporting requirement. It had to provide quick response to a user's request. It had to be built in a reasonable amount of time. Last, it had to be scaleable, upgradable, and maintainable. In regard to the above specification, SAS, with its flexibility, speed of execution, and large file handling capability, met these requirements.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the contributions made by the following people. Their work has been instrumental in the success of CORR.

Design and development of the front-end user interface:
Page Nix

Data infrastructure and file preparation:
Bill Baum
Mario Hernandez
James Mayhar
Kevin Williams

REFERENCES

1. Shneiderman, B. (2006), "Treemaps for space-constrained visualization of hierarchies"
<http://www.cs.umd.edu/hcil/treemap-history/index.shtml>
2. Bruls, M., Huizing, K., and van Wijk, J. (2000). Squarified Treemaps. *Proceedings of Joint Eurographics and IEEE TCVG Symposium on Visualization (TCVG 2000)* IEEE Press, pp. 33-42.

CONTACT INFORMATION

Your comments and questions are encouraged. Contact us at:

Larry J. Day
Director, Special Projects
Corporate Performance Measures
BNSF® Railway
Larry.Day@bnsf.com

Richard W. Dickinson
Manager
Corporate Performance Measures
BNSF® Railway
Richard.Dickinson@bnsf.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® Indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.