# DEATH TO DASHBOARDS:
## ALARMING, PERFORMANCE MANAGEMENT BASED ON VARIANCE, SYSTEM PRIORITIZATION AND OTHER THOUGHTS ON DATA VISUALIZATION

Peg McMahon and Justin A. Martin
Capacity and Performance Management
Sprint Nextel Corporation

*Performance Analysts' tool belts are getting ever heavier, but are the tools getting better? The job revolves around finding the magic in the metrics to find the proverbial needle in the haystack. However, in the world of system performance, the haystack is growing faster than the pitchfork. Advances in measurement technology are creating endless streams of metrics spilling from our systems, which would seem a Performance Analyst's dream. In reality, it is becoming an entangled nightmare. Our dashboards and stoplights, once thought to be the end all way to weave through the maze of metrics, now seem to be sending us down dead ends and straight into information traffic jams.*

**INTRODUCTION**

Dashboards are a noble attempt in simplification. As humans we endlessly seek to build intelligent machines that can figure out the answers for us. In this effort, as Performance Analysts, we want our tools to be smart enough to point us in the right direction. But are conventional dashboards the best way to get the answers we seek? There are likely many definitions of a dashboard, but one that seems to explain the purpose of a dashboard best, as written by Stephen Few of Perceptual Edge is the following:

> "A dashboard is a visual display of the most important information needed to achieve one or more objectives; consolidated and arranged on a single screen so the information can be monitored at a glance." [Few, 2004]

Stephen Few refers to the paper *Are You Drowning in BI Reports?* where it is mentioned that the appeal of dashboards comes from their ability to "Provide intuitive indicators such as gauges and stoplights. . ." which ". . . can be likened to a dashboard in an automobile. It provides an 'at-a-glance view' of the current operational state of the vehicle." [Rivard, 2004]

In other words, the purpose of a dashboard is to provide the operator with just enough information to make quick decisions in order to effectively operate the system. While the dashboard is useful in assisting in the operation of the vehicle, one could argue that it is practically useless in the diagnosis and correction of the vehicle's performance problems.

In all fairness to dashboards, they have been a comfortable bench on the journey towards the ultimate visualization for analyzing system performance. However, the time has come for Performance Analysts to demand that our system performance management tools keep moving forward. We need to look towards a better way to monitor and manage the performance of our systems at the enterprise level. To know how to improve, we should first understand what we are seeking to accomplish.

- Identify systems which are underperforming.

- Understand the level of underperformance i.e. “is it somewhat poor, or is it really bad?”

- Prioritize which systems should be addressed first. i.e. “how much is the underperformance impacting the business

In order to accomplish these objectives, the use of dashboards and arbitrary performance thresholds are not sufficient. It is the purpose of this paper to provide the reader with examples where the use of arbitrary thresholds and dashboards can hinder the performance analyst and highlight the need for better performance data visualization.

**The Thresholds Pitfall**

With the torrent of performance metrics now flowing from our systems, we are in a quandary as to understanding which metrics are important and when we should pay attention to them. Using dashboards we attempt to create a global view of “How is everything doing?” Countless off-the-shelf performance monitoring tools attempt to highlight system problems by using thresholds. However, the thresholds are typically not good indicators of a performance issue and they do not enable intuitive prioritization.

The typical dashboard makes use of thresholds by presenting indicators, such as red or green lights, which inform the user as to which metrics are above or below a predefined threshold. The problem with this is that the thresholds are often arbitrary and the indicators are usually defined within too narrow of a range in order to make intuitive decisions.

For example, imagine having a dashboard that is representing the performance of several dozen systems. You then find that there are a half dozen red lights blinking. How do you prioritize which issues to investigate first?

**Figure 1: Example of a conventional stop light dashboard**

Alarming based upon arbitrary thresholds can also cause problems. The conventional idea behind alarming is to send out some kind of notification when a threshold is breached. However, unless the threshold provides a meaningful indicator of poor system performance, the alarms will likely be disregarded or altogether turned off.

One specific example of a poor use of arbitrary thresholds is when a CPU run queue or utilization threshold is set for a database server that has varying workload demands. Since the majority of database systems undergo a backup operation during the overnight hours, it is highly likely that there will be significant CPU utilization. This change in workload may breach a threshold which was originally intended for a different workload. An alarm is then unnecessarily triggered or the performance dashboard lights changed from green to red. In this case it can be seen that there is a need for establishing smarter thresholds and step away from our reliance on the conventional dashboard.

**Alarming Based on Variance**

The next step towards better performance monitoring is the use of a *baseline* approach to performance management. Using the power of statistics, the performance metrics can be analyzed to create upper and lower control limits based on the normal variance of the system’s performance. From this analysis, dynamic thresholds can be set based on the normal variance represented by the data.

Implementing dynamic, variance based thresholds takes into account the system’s typical workload characteristics. Now, when a back up occurs in the middle of the night, as long as the same back up has occurred at the same time for the past several nights, the CPU threshold is not breached. In theory, an alarm will only occur when the system utilization is above or below a dynamic threshold which outlines the “normal” processing range of the system.

This method of monitoring will provide a more refined approach to alarming as it will help to better identify actual performance issues. This is important when the analyst is responsible for monitoring many systems. However, when implemented across thousands of systems, there will likely be several that will have at least one hour which exceeds the variance threshold, and thus triggers alarms. When using a conventional dashboard, this improved level of monitoring creates the same problem as found earlier. How do you prioritize the order in which to resolve the performance issues?

In today’s business environment the number of systems is increasing while there are fewer people to manage them. Each system has a unique impact on the business. Understanding a system's business impact and addressing system performance issues in the correct priority will save a company significant dollars. Using a conventional stoplight dashboard for system performance management will often confuse and delay critical decision making.

One way to help address the prioritization problem, using the performance variance data, could be to create a sorted list. This type of report would present the servers having the most performance variance appearing at the top. Using this list, cross-referenced with a list of systems prioritized by their business criticality would be one way to determine which problems need to be addressed first. This method is not intuitive since it requires the analyst to jog between reports. However, it is a way to use the available data in order to make the most business impacting decision.

The problem in determining how to quickly prioritize system performance issues is not necessarily due to a lacking in performance data, but rather the lack of a way to properly visualize the performance data.

**The Need for Better Data Visualization**

Over one-hundred years ago a social satire name *Flatland* written by Edwin Abbott, describes a 2D world in which its inhabitants are shapes which have an unknowingly limited view of their surroundings. The story describes the inhabitants' view of their world from only a 2D perspective. Each shape appears to one another simply as lines with no dimension. Within *Flatland* everyone was perfectly accepting of their world because they did not know any different. It wasn't until the author, who portrays himself as a square, has an extraordinary experience which enables him to rise up from the plane of his world. Looking around, outside of the plane of *Flatland* he realizes that a third dimension exists and that he is in fact a cube and not just a square. Suddenly his view of his world has changed. He gains a new understanding of problems that he had faced within *Flatland* due to his unknowing of another visual dimension [Abbott, 1884].

Within the world of systems performance management, we are in fact living in *Flatland*. The multi-dimensional reality within which our servers, applications and networks function is being represented to us in a severely limited form. In the paper entitled, *Seeing it all at Once with Barry* by Gunther and Jauvin, a radical new way to visualize system performance is offered using a hypothetical tool called 'Barry' [Gunther, 2007]. With a tool based on this concept, it appears feasible that the performance of possibly thousands of systems could be observed within a single view. Currently available performance management tools are amazing in the amount of data that they can gather and analyze. However, we are handicapped in our ability to visualize this data in such a way that enables us to make intuitive decisions with regard to "How is everything doing?"

In order to take steps in this direction, the authors have taken a step back from their traditional toolsets and made attempts to find a better way to visualize their system's performance data. One example of this is the creation of a rudimentary, but functional treemap. A treemap is typically a display of space-constrained squares which are varied in size and shades of color. The size of the squares and the shading of their color are set according to the attributes of the subjects being represented.

One example of using a treemap, would be to represent the size of a system by the size of the square, while the color shading of the square would relate to the system's CPU utilization. The squares could also be grouped within hierarchies such as by datacenter and/or business unit. Compared to the green, yellow, and red stoplights of a conventional dashboard, this example of data visualization could enable the performance analyst to quickly assess and prioritize which systems' performance should be analyzed further. Of course, this would assume that the size of the system correlates with its business criticality.

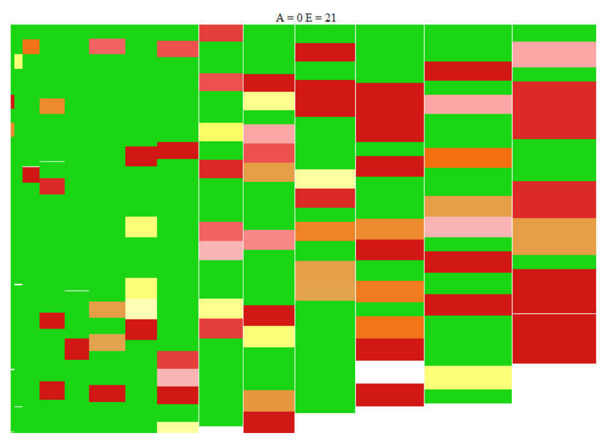


**Figure 2: Example of treemap view of many systems CPU utilization where size is based on number of CPUs and shading represents level utilization**

Another proposed implementation of a treemap would be to set the size of the square according to a metric which specifically relates to the system's business criticality. This would naturally draw the analyst's eye towards systems with a greater degree of business impact. Then using the performance variance data, shade the color of the square according to the performance variance of the system. For example, as the color goes from a lighter shade of red to a darker shade, it would indicate that variance in the system's performance, as compared to its baseline, is increasing. Those systems which appear larger and redder would be quickly identified as having a high priority performance issue.

Our use of treemaps has just begun and we are already noticing unique trends in how our systems are performing. One such example was the realization that that our Windows OS platforms as a whole were running much less utilized than other OS platforms. Using this information we are learning where we need to put more focus to gain value from our systems.

As we progress in our discovery of better ways to visualize our performance data, we are looking forward to using treemaps to enable us to gain a more intuitive view into the performance of our systems. Our goal is to implement a blended view of the business metrics, such as costs, combined with system metrics, such as response time and throughput. With this type of real-time analysis capability we expect to enable faster identification of business impacting performance issues prioritized by criticality.

## CONCLUSION

It is not the authors' intent to drive performance analysts away from their performance dashboards and to turn off their threshold based alerting. It is rather our hope to drive demand towards new and better ways to visualize performance metrics. Technology is quickly advancing our capability to obtain more and better performance data; however, there does not currently seem to be parallel growth in new and better ways to visualize the data.

When the human mind is presented with information in the right way, its capability to infer and assess the information is unmatched. In much the same way, we can interpret the multitude of cloud patterns in the sky and identify a coming storm. Wouldn't it be interesting, with the right data visualization, if the same could be done using our performance metrics?

### Disclaimers

Any opinions expressed in this paper are those of the authors only. They do not represent any official opinions or positions of their respective employers.

### References

[Few, 2004]
"Dashboard Confusion"
Stephen Few, Perceptual Edge
http://www.perceptualedge.com/articles/ie/dashboard_confusion.pdf

[Rivard, 2004]
"Are You Drowning in BI Reports?"
Kurt Rivard and Doug Cogswell
April, 2004 issue of DM Review Magazine

[Abbott, 1884]
"Flatland"
1953, ISBN 0-691-02525-8

[Gunther, 2007]
"Seeing It All at Once with Barry",
Neil J. Gunther, Performance Dynamics Company
Mario Francois Jauvin, MFJ Associates
CMG 2007