

THE MINIMUM DAILY ADULT

The Right Metrics and the Wrong Metrics

Denise P. Kalm
CA, Inc.

In capacity planning and performance analysis, we are inundated with metrics that purport to measure performance, but how we display them and understand them is what matters. Many metrics we take for granted are actually not that useful, and yet, permeate our world. This paper is intended to help you understand why the same-old, same-old metrics aren't good enough, and what works better.

"If you want to inspire confidence, give plenty of statistics. It does not matter that they should be accurate, or even intelligible, as long as there is enough of them. "

~ Lewis Carroll

Introduction

Without realizing it, we have various rules-of-thumb (ROT) metrics we use every day. The problem is – what do they actually mean? Are they useful to you? Every day when you get up, you encounter one with your daily vitamin, representing the MDR – minimum daily requirements. Dave Barry sheds light on the validity of these metrics by naming vitamins as “*little pills named A, B, C, D, E and K that the government recommends you have certain amounts of. These recommendations are based on the requirements of the Minimum Daily Adult, a truly pathetic individual that the government keeps in this special facility in Washington, D.C. where he is fed things like ‘riboflavin.’*”¹ In fact, for women over 50, calcium requirements are 4x the minimum, unless you prefer to be the Minimum Daily Adult. Would the Average

Daily Adult be better? Average based on what data?

Capacity planning and performance metrics also need to be reviewed for validity and usability, and assessed for the value they provide to a given audience. We generally have drawn the “right” metrics from the tools we have, and from what is simple to obtain. If average CPU busy is available, that is a metric of importance. Over time, we all have accepted some of these ways of viewing data as the right and only way to do so. It may take a little more work to find better numbers (or ways of viewing the data,) but you will be able to see the benefit in being more proactive, being considered business-responsive and positioning yourself as the go-to person for the numbers that matter.

When a Rule-of-Thumb is ROT The Tyranny of Average Response Time

It turns out that any metric can be valid and useful if presented in a way that makes sense. The response time of one specific transaction is a useful number – it expresses the real-life experience of a user. But what happens

¹ Barry, Dave, “Stay Fit and Healthy Until You Are Dead,” St. Martin’s Press, NY, 1985

when you average it, even over a short period of time?

Of all metrics, none is as pervasive as average response time, otherwise known as the arithmetic mean. The mean is calculated simply by adding the metric values together and dividing by the number of observations. Another related metric is median (though rarely used), which is simply that value where $\frac{1}{2}$ the observations live above the median and $\frac{1}{2}$ below. When values are mostly consistent with just a few outlier values, the median better represents the user experience. (Ex. Response times 0.1, 0.1, 0.1, 0.1, 0.5. Mean = 0.18 Median = 0.1)

In many cases, this average is calculated across many hours of a business day, and worse, often averaged again, across a week of data. Often, the average is not for a specific transaction, but for an application, which may include long and short-running transactions; this average has absolutely no meaning to anyone.

“Then there is the man who drowned crossing a stream with an average depth of six inches.”
~W.I.E. Gates

Though having a low average may seem like the best result possible, having consistent response time is what matters to the end user. Studies have shown that consistency trumps a lower, but inconsistent response every time. But what does it mean? How do you calculate the consistency? The next section – The Right Metrics – will consider the way you should be doing this. But first, let’s look at why averages are meaningless.

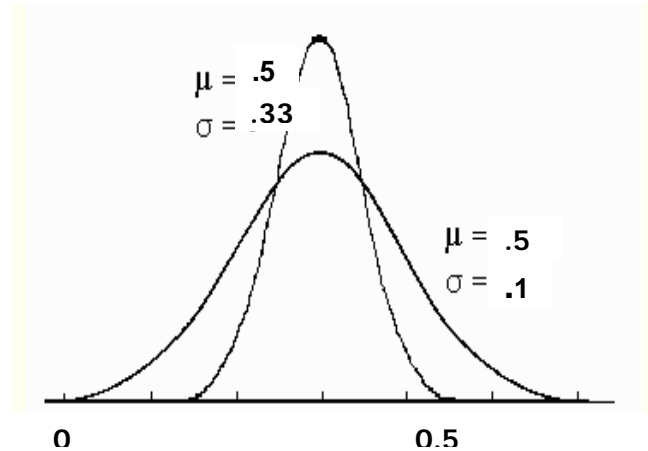


Fig. 1 Two normal distributions both with a mean response time of 0.5

Here are two populations of data, both with the same average response time. The taller one shows much more consistent response times. The broader curve shows much more variability; users would be much less happy with their experience. And yet, the performance reports would look identical – they do not reflect the actual user experience.

Variability is calculated as standard deviation or σ . This value is the root mean square deviation of the values from their arithmetic mean. Or more simply, it describes how spread out the observations (or response times, in this case), are from the calculated average. And with many metrics, such as response time, variability is bad. Two standard deviations from the mean should be approximately 69% of all data points; three encompasses approximately 95% of the data, no matter how the curve is drawn. The same percentages apply to either curve. To manage SLAs, the smaller the σ , the better.

Other Averages to Watch

In fact, few averages really make sense, unless you understand the standard deviation, and it is relatively small. Only then, does average have a meaning to anyone. Even average CPU busy (a common metric of choice) can be useless if the variance (standard deviation squared) is high. In capacity planning, it is a common practice to take averages as the baseline for modeling,

but in fact, if the variance is high enough, your systems may already be in trouble. The same is true of any resource where you track utilization. When variance is high, the next step is to gain a better understanding of the cause of the variance. If the peaks do not represent a problem that can be corrected, they reflect real resource demand, reflective of business patterns. As such, this is a capacity demand that is likely to be ongoing.

The average human has one breast and one testicle. ~Des McHale²

Adding Rabbits & Tortoises Together

Proper performance analysis requires a thorough understanding of the application. It is critical to know which workloads (transactions/processes) are important to an end-user and which do not affect them at all. Without this understanding, performance measurements may include background transactions and daemons, rendering the data pretty meaningless. But before you throw out a unit of work, be sure you understand the impact. Though a background transaction, the efficiency of a temporary storage reader transaction that pushes data to a printer for loan applications may be important to a loan processor. But in the case of that transaction, response time is the wrong metric; you need to look at queues and wait times instead.

Granular data is your best friend in these situations. And if you automate performance data collection, make sure you have created separate populations for each type of work before averaging it. The upfront research to define the populations is worth it when you go to analyze a problem in real time.

“Not everything that can be counted counts; and not everything that counts can be counted.”

~ George Gallup

Avg(Avg)=Garbage

² As there are more women than men, the average human being actually has slightly more than one breast and slightly less than one testicle – Ouch!

Averaging averages is a common strategy when you want to collect a lot of data and represent a user experience more globally. Business day or business week measurements are frequently the report increment, but rarely does the analyst actually average all the individual data points. At the very best, the SMF interval average might be used, but even that is an average. In Fig. 2, we see the problem. Though there was one hour with bad response time (1.0 sec), the average looks okay, and might be within the service levels (SLAs) established. The more data you throw into the averaging “pot,” the worse the result will be, in terms of accurately reflecting anything.

HR	RT (sec)
8	0.5
9	0.4
10	0.3
11	0.6
12	0.3
1	0.4
2	1
3	0.3
4	0.3
	0.455556

Fig. 2 The average of averages

Another problem with averaging (or curve smoothing, as it should be called), is that you miss the spikes that can inform you of problems or changes in demand. If your information on future expected demand is not all that accurate (who’s is?), then you need to look at other ways to predict the future. No hardware vendor ever gave a good discount when you needed the box yesterday.

Percent of What?

Using the Breast Cancer Calculator, women were able to discover their risk of cancer.³ Obviously, anything short of no risk at all was scary. But searching further, they found they could reduce their risk by 18% by exercising. But 18% of what? It sounds like a big number, but if your lifetime risk of breast

³ “Detailed Breast Cancer Risk Calculator” <http://www.halls.md/breast/risk.htm>

cancer is only 3%, this only changes the number to 2.5%. The hormone replacement treatment (HRT) studies were the same; for women on estrogen for 15 years, their risk rose 48%.⁴ A big number, but if your risk is actually only 1%, that really only raises the risk to 1.48%. Often, studies fail to give you the baseline data, so the increase (or decrease) sounds important.

For a metric, the same may be true. If you manage to decrease CPU use of a given transaction by 50%, that sounds like a noteworthy accomplishment. But it doesn't usually translate to a 50% decrease in CPU demand overall. First, you must know how much CPU the transaction used initially. Then, you need to know how many transactions usually run. (The same calculation applies equally to UNIX/Linux or Windows processes). If the transaction is only using .01 CPU seconds and runs only 1000x/day, it might not be a big deal. It's a great way to lie with statistics and make an analyst look like a hero, or save him from being the goat, but it isn't useful.

Percent change is clearly a bad metric, but so is raw percentage, unless it is clear what it is behind the number. We used to have to report hourly % CPU utilization to a credit card business manager, but he never asked what it really meant. We gave him the utilization of the CICS region; I'm not sure how he used it. 50% can represent 50% of a uni-processor, which is somewhat meaningful, or it can be 50% of one engine of a 10-way machine. Then, it is important to know if the work involved can only utilize a single engine or can be processed across multiple engines. In the mainframe world, where pushing the CPU to 100% is common, with no degradation in performance, even 100% busy has no meaning. Would you buy a new CPU just because the system routinely runs 100% busy during the peak hour? In the UNIX and Windows world, the number has no merit, because few can run flat out. If you knew the practical capacity of a server, the utilization based on that might be helpful, but in most cases, this number is application-specific and

⁴ "Study: Estrogen Use Can Increase Breast Cancer Risk"
<http://www.cnn.com/2006/HEALTH/conditions/05/09/estrogen.cancer.ap/>

difficult to calculate without some form of modeling or simulation tool.

"Say you were standing with one foot in the oven and one foot in an ice bucket. According to the percentage people, you should be perfectly comfortable."
~ Bobby Bragan, 1963

"USA Today has come out with a new survey - apparently, three out of every four people make up 75% of the population."
~ David Letterman (1947 -)

The Right Metrics

First, you have to know what you are measuring. For a mainframe application, you want to know the response time of a UOW (unit of work). For other applications, some technique, such as ARM, may be needed to connect processes into an application. Tools exist to help calculate such a number and yes, even in UNIX land, there is nothing better than really knowing the response time for individual types of work. You must be able to separate out work that impacts a user (foreground processing), and work that doesn't (background transactions or daemons or services). Once you achieve this understanding, you can automate performance data collection and consolidation, but not before.

This means understanding the application; the developers can often help here. This effort takes time, but it is paid back in full when you have a performance problem or need to predict capacity growth. Knowing that an increase in credit card charges will impact only certain transactions/processes allows a much more granular forecasting than just projecting a line for the entire application based on 15% expected growth.

Consistency, not Averages

Consistency can be most easily derived by taking the standard deviation of the data. If you have SAS or Microsoft Excel, this can be easily calculated. (See Appendix A for how to get Excel to give you statistical functions). "The **standard deviation** is a statistic that tells you how tightly all the various examples are clustered around the mean in a set of data. When the examples are pretty tightly

bunched together and the bell-shaped curve is steep, the standard deviation is small. When the examples are spread apart and the bell curve is relatively flat, that tells you that you have a relatively large standard deviation.⁵

It can also be useful to have minimum and maximum points from your data, to give the range. Small standard deviations are good; larger ones mean that consistency in this metric could be a problem. Large is relative to the minimum and maximum values in the data. As in Fig. 1, 0.33 may not sound like a large number, but if the maximum value is 1.0, it is pretty large.

After gaining a good understanding of what represents user-sensitive, business-critical work, define it in your performance reports. Put yourself in their shoes – what would they want to know? What has meaning for business people, end users, etc.? Check out your understanding with the community; performance reporting is for them. An interesting aside is the popularity of the Six Sigma standard; few people realize that the Sigma is the symbol for standard deviation and that the goal is to drive to eliminating defects such that the good population of components is 6 standard deviations from the mean, relative to the specific limits.⁶ So be cautious in your use of standard deviation in reporting for others; it is not a well-understood term.

Qualifying Percentages

Despite the pervasiveness of percent, it is rarely helpful without qualification. Even in CMG presentations, it is not uncommon to see a % CPU utilization number that may mean percent of a 4-way, or of just an individual engine. If you use percent as a value, always qualify it, so everyone has a common understanding. As an example, 85% CPU busy sounds reasonably high on a UNIX server, but if that is 85% of one engine on a 10-way, the server needs more work to do. Another instance where confusion is likely

is measuring an entity like CICS or IMS where multiple engine use is likely. If CICSREGA measures 250% of an 8-way processor, to really understand capacity limits, it is essential to understand the demand of the most intensive TCB, usually the QR (quasi-reentrant). Sub-processor capacity constraints in CICS are almost always due to a maxed out QR TCB.

An additional valuable qualification is to indicate usable threshold. If a disk device starts thrashing at 60% busy, indicate that, to give a better perspective in viewing the data. 50% busy may sound like there is plenty of resource available, but if 60% is the threshold, you should already be making plans.

Never use percent change by itself, unless you report clearly on the before and after numbers. It gives little information and is too open to making more (or less) of a change than is strictly warranted. As an example, reducing the CPU demand of a transaction by 50% sounds great, unless the original demand was 0.02 seconds and now it is 0.01. No one will notice, unless this transaction is executed many thousands of times per day. The latter example needs to be stated relative to a meaningful value, such as that this workload comprises 50% of a single engine and by cutting the CPU demand of this transaction by 50%, no upgrade will be necessary for the next four quarters, or that the new workload, Y, requiring Z amount of CPU will easily fit in on this processor now, with no performance impact.

How to Display Your Data

Instead of response time averages (or any other average), 95 percentile can be more informative – what is the experience of a user 95 times out of 100. CA-MICS will display this for you and there are other functions that can give you this number. This is understandable to almost any reader; standard deviation might not be. Mode can also be an interesting number – what is the most frequently occurring data value in your population. Sometimes, there is one hour of the day that everyone is tracking; if so, report on that. But in general, let them know the trends. One assumption underlying percentiles is that outlier data is expected and that by showing percentile, you are basically

⁵ “Standard Deviation,”

<http://www.robertniles.com/stats/stdev.shtml>

⁶ “Six Sigma: What is Six Sigma?”

http://www.isixsigma.com/sixsigma/six_sigma.asp

throwing out the rest of the data. For reporting and to base SLAs on, this is perfectly reasonable, but performance analysts need to understand the outliers, to ensure they do not represent an important subset of transactions (processes), in other words, that they are outliers and therefore, unusual.

Here is an example of a report that you might produce:

Performance Report
ABC Bank
10/18/06

	95%tile	95%tile at 10AM	95%tile Last Week
Credit Card Autho	0.5	0.5	0.5
Online Pay	0.4	0.7	0.4
ATM Txns	0.3	0.3	0.4

Response time is not as good as the prior week, and though it may still fall within the SLA, it helps identify a trend that needs investigation.

Resource utilization is mostly of interest to capacity planners and then, only when it is short supply. What matters most to end users is cost. How much does it cost to run a unit of work? They already have their unit price; this information helps them to understand if they are really making money. Ideally, chargeback can be granular enough to help them understand their costs at this level. Once they have the chargeback, the following report will probably help them keep track of the impact of development change:

Chargeback Report
ABC Bank
10/18/06

Workload	#/day	CPU per Unit	Cost/CPU sec	Total Cost
Credit Card Autho	10,000	0.05	0.05	\$25.00
Online Bill Pay	15,000	0.1	0.05	\$75.00
ATM Transactions	20,000	0.1	0.05	\$100.00

If you do need to report on utilization, make sure it makes sense to the customer, likely others in the IT organization (and Acquisitions). Here is a sample:

PROC	Type	#Eng	Usable %	Pk busy Over All	Q Igth	Pk Busy on Busiest
MFA	2064-102	2	100	98	1	100
SunB	E20K	30	55	40	6	65
AIXC	p5-570	4	60	45	2	50

This report may be showing a problem – at least, it is worth investigating.

These were all Excel spreadsheet reports. For some use cases, graphs, pie charts and other formats may work better. Particularly when you want to display a trend, a line or bar chart may make the point best.

Summary

There is a use for these kinds of statistics – the wrong metrics; for researchers to expand minor variation into a national panic, for politicians to influence voters, for anyone who has an interest in obfuscating the truth. But even those who have no reason or desire to confuse can fall into the trap of using the numbers that are so readily available. In the end, making the extra effort will help you notice trends faster, be more proactive and on top of your game. In addition, your customers will understand what you have to offer and value your services more.

“Without data, all you are is just another person with an opinion.”
~ Unknown

“Do not put your faith in what statistics say until you have carefully considered what they do not say.”
~ William W. Watt

Appendix A – Loading Statistical Functions in Excel (Win XP 2002/ Office 2003)

The Analysis Toolpak is an Excel add-in program that is available when you install Microsoft Office or Excel. To use it in Excel, however, you need to load it first.

1. On the **Tools** menu, click **Add-Ins**.
2. In the **Add-Ins available** box, select the check box next to **Analysis Toolpak**, and then click **OK**.

Tip If **Analysis Toolpak** is not listed, click **Browse** to locate it.

3. If you see a message that tells you the Analysis Toolpak is not currently installed on your computer, click **Yes** to install it.
4. Click **Tools** on the menu bar. When you load the Analysis Toolpak, the **Data Analysis** command is added to the **Tools** menu.

Or go to Help and search on “Data Analysis” for the correct method for your version of software.

Bibliography

1. [Barry] Dave Barry, "**Stay Fit and Healthy Until You Are Dead**," St. Martin's Press, 1985
2. [Dixon] Wilfred J. Dixon & Frank J. Massey, Jr. "**Introduction to Statistical Analysis**," McGraw-Hill Book Company, 1969
3. [Geis] Irving Geis, "**How to Lie With Statistics**," W. W. Norton & Co., 1954
4. [Kalm] Denise P. Kalm, "**The Stork Correlation – Use & Abuse of Statistics in Performance and Capacity Planning**," CMG 2002