

Realtime  
publishers

Resolving VMware vSphere's Six  
Biggest Performance Issues  
The Essentials Series

# What Ten Counters Quantify Those Behaviors?

sponsored by



Greg Shields

---

# Introduction to Realtime Publishers

---

by **Don Jones, Series Editor**

For several years now, Realtime has produced dozens and dozens of high-quality books that just happen to be delivered in electronic format—at no cost to you, the reader. We've made this unique publishing model work through the generous support and cooperation of our sponsors, who agree to bear each book's production expenses for the benefit of our readers.

Although we've always offered our publications to you for free, don't think for a moment that quality is anything less than our top priority. My job is to make sure that our books are as good as—and in most cases better than—any printed book that would cost you \$40 or more. Our electronic publishing model offers several advantages over printed books: You receive chapters literally as fast as our authors produce them (hence the “realtime” aspect of our model), and we can update chapters to reflect the latest changes in technology.

I want to point out that our books are by no means paid advertisements or white papers. We're an independent publishing company, and an important aspect of my job is to make sure that our authors are free to voice their expertise and opinions without reservation or restriction. We maintain complete editorial control of our publications, and I'm proud that we've produced so many quality books over the past years.

I want to extend an invitation to visit us at <http://nexus.realtimepublishers.com>, especially if you've received this publication from a friend or colleague. We have a wide variety of additional books on a range of topics, and you're sure to find something that's of interest to you—and it won't cost you a thing. We hope you'll continue to come to Realtime for your educational needs far into the future.

Until then, enjoy.

Don Jones

Introduction to Realtime Publishers..... i

What Ten Counters Quantify those Behaviors? ..... 1

    Ten Behavior-Quantifying Counters ..... 2

        Counter #1: CPU.ready.summation ..... 2

        Counter #2: CPU.usagemhz.average..... 2

        Counter #3: mem.active.average ..... 2

        Counter #4: mem.consumed.average ..... 2

        Counter #5: mem.swapped.average..... 2

        Counter #6: mem.vmmemctl.average..... 3

        Counter #7: disk.busResets.summation ..... 3

        Counter #8: disk.totalLatency.average ..... 3

        Counter #9: disk.usage.average ..... 3

        Counter #10: net.usage.average..... 3

Counters Aren't Everything..... 4

## **Copyright Statement**

© 2011 Realtime Publishers. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtime Publishers (the "Materials") and this site and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtime Publishers its web site sponsors. In no event shall Realtime Publishers or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtime Publishers and the Realtime Publishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtime Publishers, please contact us via e-mail at [info@realtimepublishers.com](mailto:info@realtimepublishers.com).

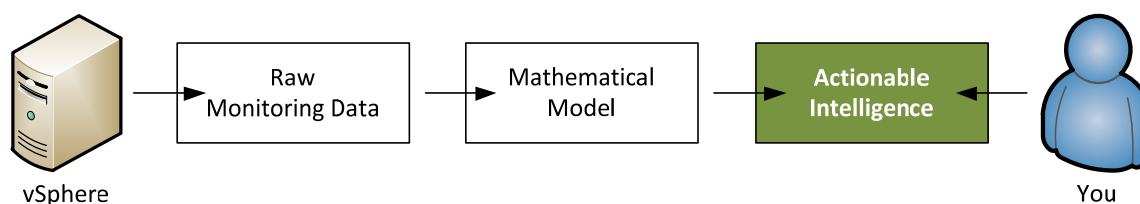
# What Ten Counters Quantify those Behaviors?

A virtual environment is by nature an invisible environment. You simply can't crack the case on a vSphere host and expect to "see" the behaviors going on inside. That's why its counters are so important. They represent your only way to understand the behaviors and quantify potential resolutions.

But counters by themselves are very scary things. A counter is by definition just a number. Put together enough of those numbers, and you'll create a graph not unlike Figure 1 in the previous article. Divining meaning from the points on that graph, however, is another thing entirely. Studying charts and graphs is an activity that can consume every part of your workday. With those graphs constantly evolving with a virtual environment's behaviors, just keeping up is a challenge all its own.

*Yet monitoring virtual machine performance is a virtual environment's most important activity.* That's why this series' previous article suggested that an unaided person can never effectively convert raw data into actionable intelligence. Oh, yes, in a tiny environment with just a few interdependencies, you probably might. But most of our VMware vSphere data centers are large and distributed. Finding the source of a performance issue isn't easy when you're starting at its unending integers.

Relating those numbers to the actions you should take is what you really want. Figure 1 shows that line of thinking in relation to the first article's notion of actionable intelligence. In it, you can see how the raw monitoring data from a vSphere system can flow through some kind of built-by-somebody-else mathematical model that converts raw data into suggested actions.



**Figure 1: How actionable intelligence is generated.**

What does that raw data look like? What top-ten counters might you plug-in to such a model to represent virtual machine and virtual host behavior? These ten are the topic of this second article. In the next section, you'll gain an appreciation for the amount of effort it takes to convert just ten integers into answers.

## Ten Behavior-Quantifying Counters

What follows are ten counters considered to be most important in determining the rough behaviors within a vSphere environment. Think of them as an equation with ten variables. With the right equation, plugging in these values will net you an approximation of your vSphere environment's behaviors.

### Counter #1: CPU.ready.summation

CPU ready time relates to the percentage of time a virtual machine was ready to use a physical CPU but could not get scheduled to run on it. A high CPU.ready.summation time means that virtual machines are waiting for physical CPU resources that aren't available. A high count here tends to indicate that physical CPU is a bottleneck to performance.

### Counter #2: CPU.usagemhz.average

The CPU.usagemhz.average counter measures average CPU usage in megahertz during a configured interval. Measured over all physical processors, this counter is a primary indicator of the amount of CPU load being placed on the host.

### Counter #3: mem.active.average

mem.active.average represents an estimation of how much memory is actively being used by virtual machines. This estimation is made by the VMkernel and is based on recently-touched memory pages. This memory counter references the quantity of guest memory the guest is actually using to accomplish its workload requirements.

### Counter #4: mem.consumed.average

Slightly different than the mem.active.average counter, mem.consumed.average measures the amount of guest physical memory consumed by a virtual machine. When measured for a virtual machine, this level of memory includes shared memory and memory that is reserved but not used. This counter can also be measured for hosts and clusters. When measured for a host, the counter measures the amount of machine memory used on the host. For a cluster, it measures the amount of memory used by all powered on virtual machine in the cluster.

### Counter #5: mem.swapped.average

Host swapping is a last-ditch approach used by the VMkernel during periods of contention to ensure virtual machines never run out of available memory. Host swapping requires transferring memory from RAM to disk, which significantly reduces its performance. The mem.swapped.average counter references the current amount of guest physical memory that has been swapped out to its swap file. A value here greater than zero can indicate that memory is a bottleneck to performance.

**Counter #6: mem.vmmemctl.average**

Over-commitment in vSphere environments is handled through a process called ballooning. The process reclaims unused memory from running virtual machines to make it available for others. An excessive value for mem.vmmemctl.average means that virtual machines have been assigned too much memory that they are not using but is needed by other virtual machines. A large amount of ballooning can have an impact on overall host performance.

**Counter #7: disk.busResets.summation**

SCSI bus resets occur when a read or write command cannot be completed within an acceptable amount of time. These resets often indicate an underlying performance issue within storage hardware and are measured using the disk.busResets.summation counter. A non-zero value here can indicate that storage is a bottleneck.

**Counter #8: disk.totalLatency.average**

Total latency refers to the absolute quantity of time elapsed between submitting a command to storage, processing that command, and receiving the anticipated response. Measured by the disk.totalLatency.average counter, its information helps to identify the total amount of virtual machine processing delay caused by storage hardware, and is another indication that storage is a bottleneck.

**Counter #9: disk.usage.average**

Related to the disk.totalLatency.average counter, disk.usage.average measures the total disk I/O rate. This information useful for identifying when storage—either within the storage itself or the connection to that storage—is a bottleneck to virtual machine performance.

**Counter #10: net.usage.average**

This final counter measures the combined send and receive rates for network traffic during a configured interval. The net.usage.average counter can be measured against a single virtual machine or the entire host, and is used to identify how much data traffic is passing in and out of the measured host or virtual machine.

## Counters Aren't Everything

These counters might create that ten-variable equation, but they by no means fully approximate every behavior your vSphere environment experiences. The actual set of counters is far higher, making the model far more complex. That said, knowing these critical ten gets you started down the road of quantifying vSphere's behaviors. It also gives you a much better appreciation for how necessary assistive support is in resolving VMware vSphere's biggest performance issues.

That said, counters aren't everything—nor are mathematical models. Both counters and the models they feed only work when the environment remains predictably stable. Although virtualization is by nature a dynamic architecture, you can gain the right kinds of stability by observing a set of good practices. The third and final article in this series explains twelve practices that will ensure your actionable intelligence is in fact intelligent.