Active Continuous Optimization (ACO) for Server Infrastructure Performance and Density Improvement

Enterprise server infrastructure optimization creates higher performance and guest densities within existing infrastructure

Like what you read?
If so, click HERE to get a FREE evaluation of Veloxum's software.

Performance degradation occurred in:

77% of virtualization projects

69% of data center consolidation projects

43% of high-bandwidth Web 2.0 installations

IT organizations face an unfolding crisis that limits their ability to deliver applications across the enterprise. Their top priorities – data center centralization, virtualization and collaboration – are cost-saving measures that negatively impact server performance and capacity, and saturate network bandwidth, creating an expanding drag on application performance. IT must find a way to resolve these issues before they can improve application performance across the enterprise.

Active Continuous Optimization attacks this problem head-on. ACO improves performance, and in some cases, generates order-of-magnitude increases in server performance. Virtualized and accelerated systems benefit as well. Veloxum's ACO solution is a rational alternative to adding servers, to upgrading bandwidth or to tolerating declining application performance.

Application Delivery Challenges

According to a recent survey by CIO Magazine, virtualizing servers, centralizing data centers and adding high-bandwidth collaboration tools are IT's top priorities during this economic downturn[1]. While these projects may make sense financially and tactically, they also have a negative effect on application operation. Consider these results from a recent survey[2]:

- 77% of corporations reported that virtualization had had a significant, negative impact on application performance.
- 69% of companies said data center consolidation was causing application degradation.
- 43% confirmed that high-bandwidth applications, such as video and collaboration, were making it difficult to maintain performance.

Operating System Settings: Lurking Inefficiency

A covert but significant performance obstacle exists within Enterprise server infrastructure: the native inefficiency and static nature of operating system settings.

For example, Microsoft Windows 2003 Server, Exchange 2003 and Windows XP – a common application set commonly deployed together – have over 350,000 permutations of configuration settings, not including the huge number of settings for browsers, or OS settings for CPU, I/O and RAM on servers, along with their associated desktop hardware.

Too many configuration permutations exist to manually optimize system performance

Some of these settings may interact poorly with each other or even conflict. The majority are not well documented, especially in the significant way that they can cause synergistic effects with other settings. Although IT personal can manually tune all hardware and software systems, in larger data centers**,** it is impossible to do so in a realistic timeframe. Virtualized systems magnify this issue as local changes become irrelevant with shifting workloads. As a result, IT personnel encounter performance degradation and reduced capacity within their server infrastructure across the enterprise. According to a study commissioned by the U.S. Department of Energy, headed by Dr. Jonathan Koomey, the average server delivers less than 20% of its performance capacity when deployed in a commercial enterprise.

The Solution: Active Continuous Optimization

Active Continuous Optimization automates optimizing OS settings to unlock systems' potential and increase their efficiency and capacity

Enterprise server infrastructure optimization tools that utilize active continuous optimization (ACO) attack overt and covert causes of degradation. They unlock the potential of the existing server infrastructure thus increasing the efficiency of what you already own, improving application performance across the enterprise without costly hardware upgrades or increases in bandwidth.

Veloxum's ACO approach

Veloxum identifies and fixes problems continuously - including virtualized systems - and does so without voiding support contracts.

Veloxum created its own comprehensive Enterprise server optimization solution specifically to tune server performance and to maximize server capacity. It targets the heart of the server, the operating system, working to maximize the efficient use of all server and client components that depend upon OS interaction including:

- Applications and Databases
- CPU
- Memory
- Network
- Storage

Supported Operating Systems

- Microsoft Windows Server 2003,2008
- Microsoft Windows XP, Vista
- Linux
- Sun Solaris
- Apple OSX
- IBM AIX
- VMware
- Hyper-V

Supported Applications & Databases

- Microsoft Exchange / Outlook
- Microsoft SharePoint
- Lotus Notes
- Citrix
- SQLserver
- Oracle

Veloxum's features make it an exceptional tool by:

- Identifying and fixing performance problems via active remediation through a patent-pending process
- Rapidly optimizing both server and client, often within minutes
- Adjusting existing hardware and OS settings "out-of-band" while staying within the manufacturer's supported ranges
- Periodically optimizing with changes deployed either automatically or through existing change-control procedures
- Supporting Linux, AIX, SUN and Windows OS with supplied OS modules
- Optimizing popular apps such as Exchange, SharePoint, SQLserver, Lotus Notes and Oracle with optional Application Optimization Modules (AOM)
- And deploying quickly and easily

If you would like to know more about Veloxum's architecture, installation and operation, see Appendix B.

VMware defines various terms for describing memory from within the VM context and from the ESX (host) context. These terms are used in the graphs presented below.

Memory Terms in the VM Context

- *Memory Consumed*: Amount of machine memory assigned to the VM, accounting for savings from memory shared between other VMs.
- *Memory Granted*: Amount of guest physical memory mapped to machine memory on the host.
- *Memory Active*: Amount of memory recently used by the VM. (The inverse of this is the amount of idle memory.)
- *Memory Shared*: Amount of guest physical memory which is shared between other VMs.

Memory Terms in the Host Context

- *Memory Consumed*: Tally of all machine memory used by VMs and the Hypervisor.
- *Memory Granted*: Tally of all guest physical memory that is mapped to machine memory.
- *Memory Active*: Tally of all guest physical memory recently used by the VM (statistically sampled).
- *Memory Shared*: Tally of all guest physical memory which is shared through transparent page sharing.

Veloxum modifies such settings as the memory tax rate and copy-on-write buffer sizes. These changes are subtle but can lead to dramatic performance improvements.

In-depth Example: Memory Over-provisioning

Many virtualization platforms allow memory over-provisioning across machines running on a given host. Over-provisioning encourages full utilization of the host's memory, as most virtual machines rarely use the full apportionment of memory granted to them. The keys to making this kind of regime work are two reliable and fair mechanisms, one for reclaiming idle memory and one for sharing pages with identical content between VMs.

VMware has deployed both of these methods. "Memory ballooning" is a technique used to reclaim idle memory while transparent page remapping is used to share pages between VMs. Memory ballooning works by inflating or deflating the pinned memory footprint of a pseudo-device driver in the VM's kernel as memory is either reclaimed from, or allocated to the VM, respectively. The VM's own memory manager kicks in as its "physical" memory expands or contracts.

Sampling methods help the hypervisor support the idle memory reclamation mechanism to determine which pages are idle and to assist with proportional-share calculations that assign proportional value to pages based on the priority given to each VM.

In the Figure 1 below, the graph shows real sample data from an ESX host running six over-provisioned VMs.

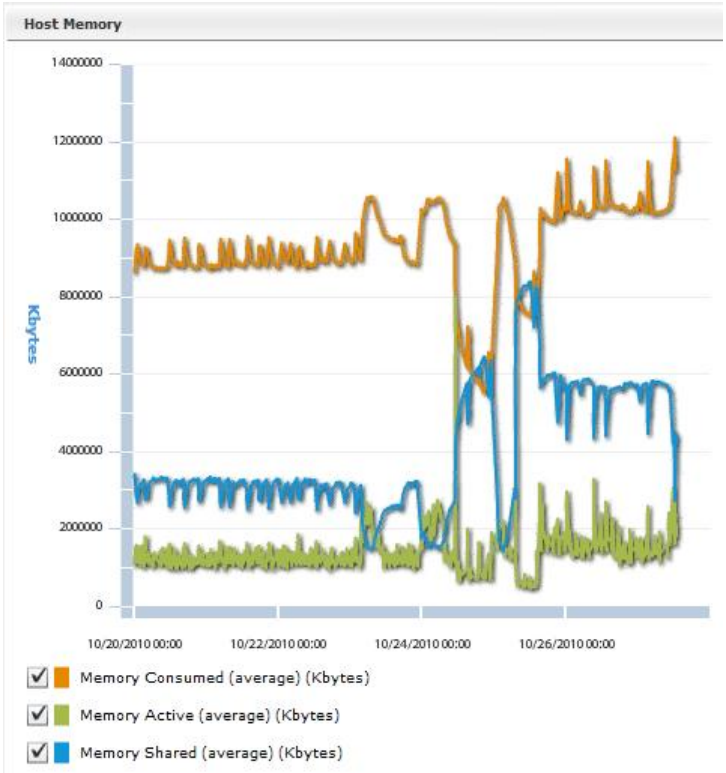


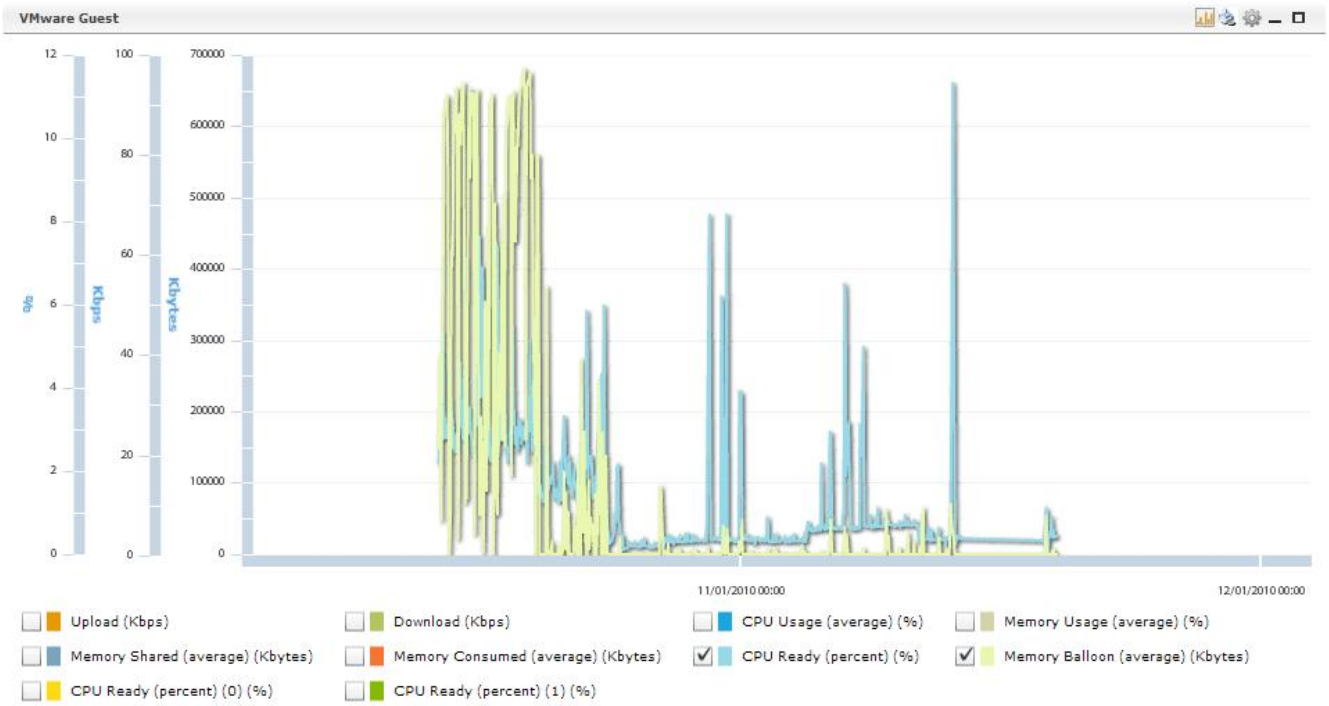
Figure 1: Memory Ballooning Before and After Optimization

It demonstrates the effect of optimizing the idle memory settings and shared memory settings across all of the VMs on this host. High memory balloon rates and large variable-sized memory balloons found on two of the VMs led to high CPU Ready Time %s across all the VMs, high CPU utilization on the host system, and unacceptable performance degradation of all applications running on the host.

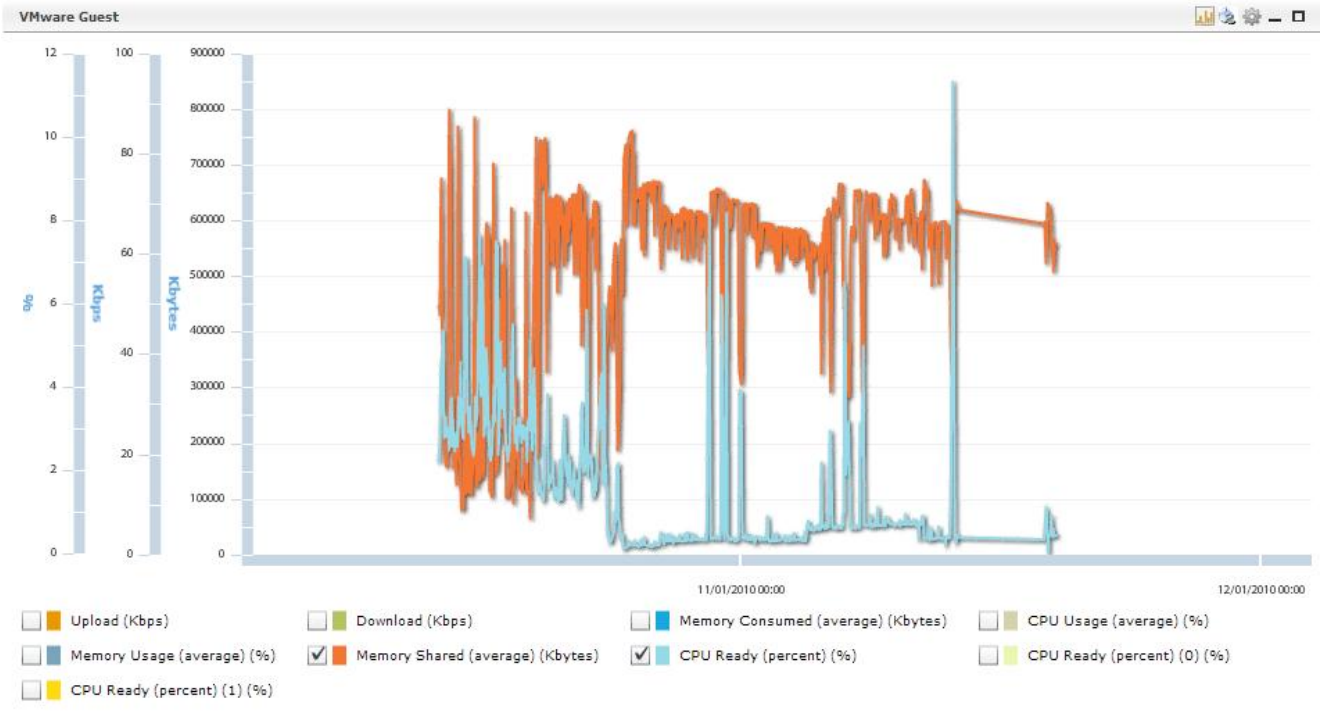
Optimizations (post-10/24/2010 on the previous graph) increased memory sharing without a dramatic increase in overall memory consumption.

Performance Improvements

The following charts demonstrate the performance improvements due to changes in memory ballooning / idle memory reclamation and shared memory settings for a typical VM on the profiled host.



Changes to idle memory settings (such as the memory tax rate or active page sampling rate) had a dramatic, stable, and consistent improvement on the degree to which memory ballooning was active in this VM. Enabling optimizations decreased the amount of ballooning, the variation in balloon sizes, and, most importantly, the CPU Ready time, which dropped to near zero (optimal). Prior to optimization, ballooning caused this VM and others to waste CPU resources on managing a highly variable memory allocation regime.



This graph above illustrates how increasing the degree to which memory sharing was used actually decreased ballooning in the VM and improved CPU Ready times. Modifications to settings, such as the copy-on-write heap size, encouraged the hypervisor to find and share more pages.

Conclusive Results

Figure 2 summarizes the overall results for the VMs running on the host.

Prior to optimization on 10/24/2010, the host reported 60% constant utilization of its CPUs. Once the Veloxum appliance automatically modified memory settings (and others), CPU usage dropped to less than 10% while memory consumption, as described earlier, increased only slightly.

Veloxum also optimized the memory, disk, and CPU characteristics of the individual VMs themselves. Veloxum made these changes in addition to the changes made to the underlying hypervisor layer. This unique linkage of VM changes to hypervisor changes ensures that the inter-related nature of all these settings is both accommodated and maximized.

The net result is that performance for all VMs dramatically increased, while showing that CPU and memory resources could accommodate additional VMs.

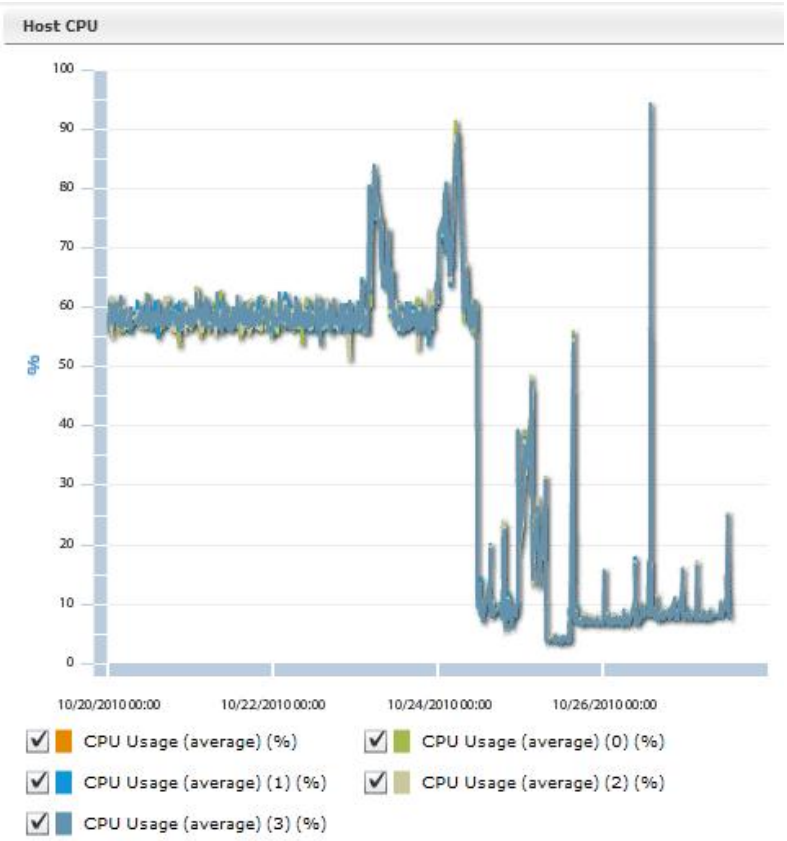


Figure 2: VM CPU Improvement with Optimization

Additional Performance Improvements

The following charts demonstrate typical performance improvements in a number of varied situations gained using Veloxum to tune CPU, memory, and I/O resources.

WAN/LAN Throughput

Figure 3 shows the improvements to WAN and LAN made possible by Veloxum's ACO process. WAN throughput (closer bars) increases from 37.13 Mbps (blue) to 443.99 Mbps (purple), a 1200% improvement.

LAN traffic increases from 46.86 Mbps (blue) to 167.39 Mbps (purple), a 350% increase.

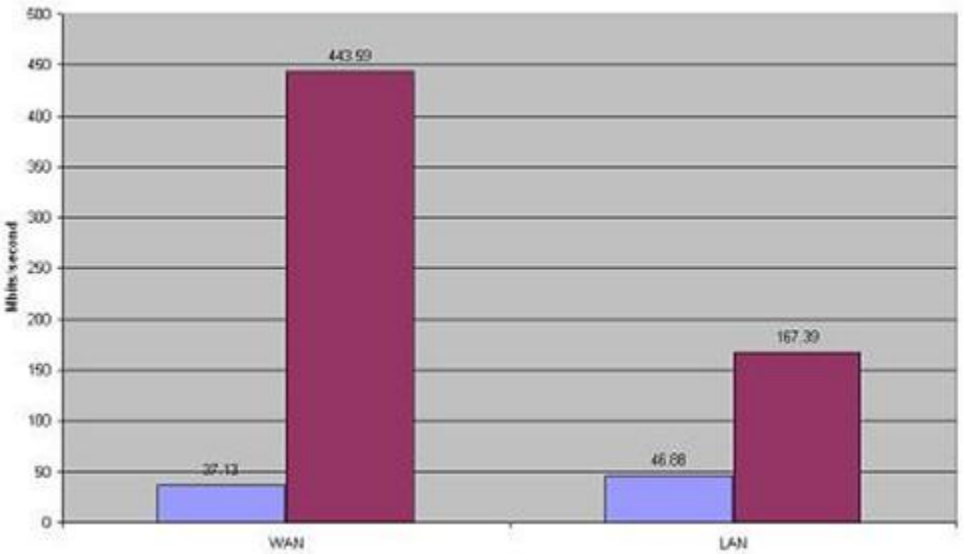


Figure 3: WAN/LAN Traffic Improvement Before and After ACO

VMare/Outlook Performance

Figure 4 shows the improvements to Microsoft

Outlook's responsiveness running on VMware made possible by Veloxum's ACO process. Exchange response times decrease from an original 8 seconds to 1 second, an 800% improvement.

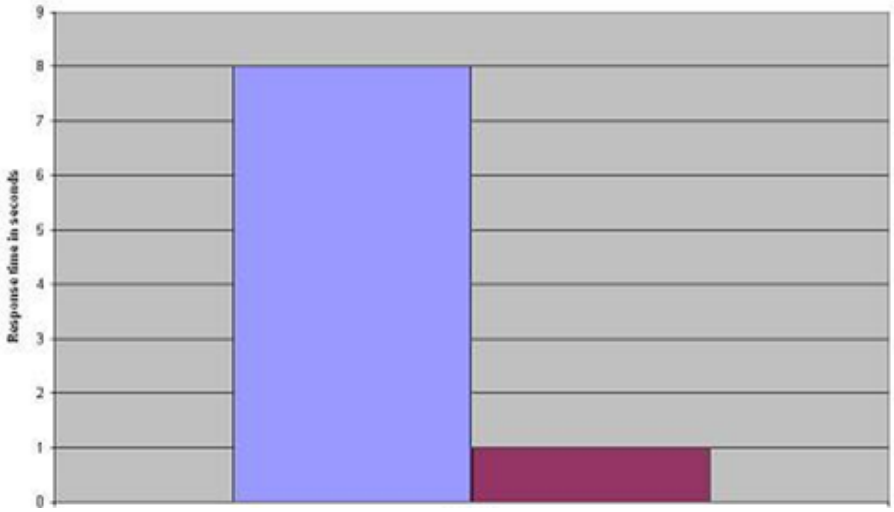


Figure 4: Improved VMware / Outlook Exchange Responsiveness

SQLserver Response

Figure 5 shows the improvements to Microsoft SQLserver responsiveness made possible by Veloxum's ACO process. Server response times decrease from an original 90 seconds to 53 seconds, a 60% improvement.

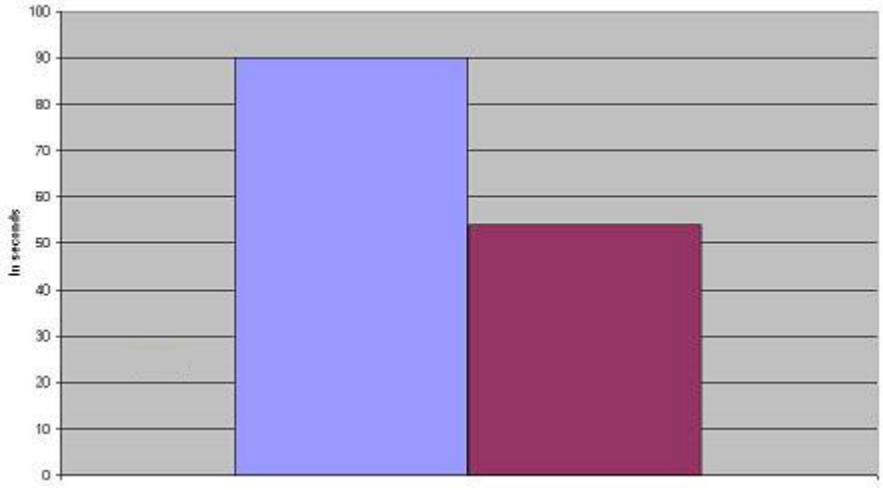


Figure 5: Improved SQLserver Responsiveness

iSCSI Throughput Improvement

Figure 6 shows the improvements to iSCSI throughput made possible by Veloxum's ACO process. Network throughput increases from an original 2475 I/O/s to 3525 I/O/s, a 42% improvement.

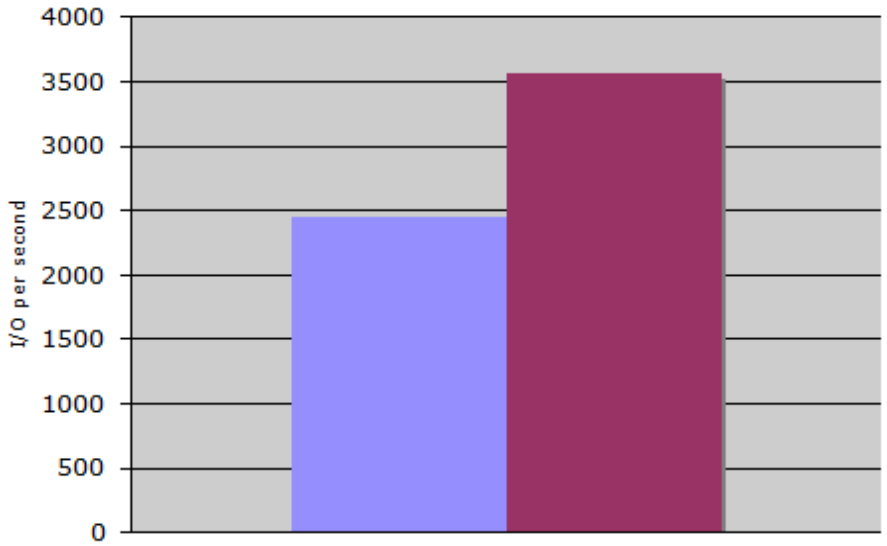


Figure 6: iSCSI Throughput Improvement

Conclusion

Veloxum designed its software's ACO process to maximize the efficiency of your existing infrastructure and to combat performance degradation whether caused by poor OS settings or the side effect of IT priorities. It delivers immediate Return on Investment (ROI) and lets you do more with your existing systems.

Like what you read?
If so, click HERE to get a FREE evaluation of Veloxum's software.

---

[1] "How to Prioritize IT Spending During an Economic Recession" CIO Magazine, November 20, 2008; <http://www.cio.com/article/464725>
[2] "Optimizing Application Delivery over the WAN" Aberdeen Group, August 2008

Appendix A: ACO under the Covers – Multi-variable Optimization

Do you remember the first time you ever used an automatic camcorder? When you pushed the 'on' button, it seemingly came to life, with the lens twisting-and-turning out of the front. When aimed at your subjects, it knew to focus itself on them. If you happened to move into another room or outside, it automatically adjusted the light level to compensate for its new surroundings. Regardless of what you did, it would correct for it and optimize the image. Do you remember your thoughts at the time? Most likely it was, "Cool! This is like magic!" Well, welcome to the world of Active Continuous Optimization (ACO), where machines anticipate your needs and adapt to environmental changes.

To understand ACO operation theory, think about the automatic camcorder's operation. While modern camcorders operate, using highly-advanced algorithms beyond the scope of this paper, for the sake of explanation, this paper presents a greatly simplified operational scenario, focusing on only two functions and their associated camcorder settings – focal-length and light level.

When a videographer points a camcorder at a subject, the camcorder moves the lens back and forth in order to maximize the camcorder's relative focus in regard to the subject. The graph of the relative focus vs. the focal-length setting might look like the graph shown in Figure 1 below. The optimal focal-length range is around the peak of the relative focus value. If the focal-length is to the left of the optimum range, the subject is too blurry because the camcorder focuses too close for the subject. If to the right of the optimal range, the subject again becomes too blurred as the focus is too far away.

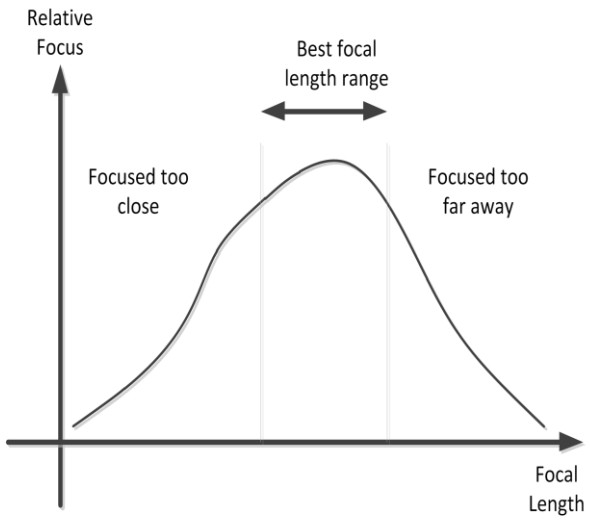


Figure 1: Camcorder Focus vs. Focal Length

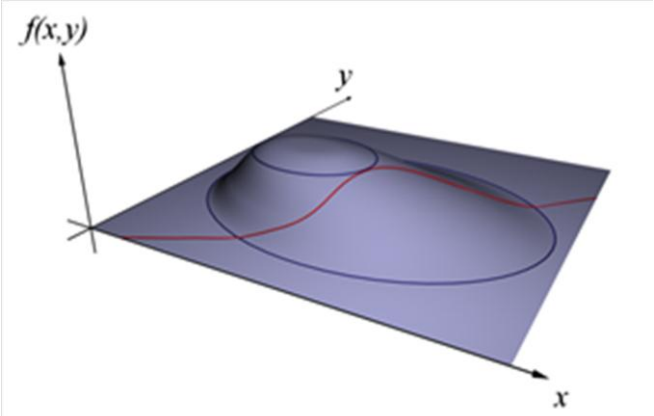


Figure 2: Focal-length optimized for light level

Perhaps it is obvious that the simple focus example above assumes a particular light level. If you constrain focal-length over all light levels settings - as camera manufacturers must - the graph becomes a three-dimensional contour as shown in Figure 2. In this diagram, the purple contour shows all possible focal-length settings for all possible light level settings. The red line represents a particular range of light levels overlaid onto this contour.

Solving for the optimum**,** focal-length setting for a particular light level then becomes a mathematical exercise in determining where the set of focal lengths (the upper contour ring) is tangential to the maximum (optimum) brightness for a given light level. In Figure 2 this occurs near the center of the diagram. When the camera figures out this value, it can zoom the lens in or out to the optimum setting for both variables.

Once you understand the general concept of optimizing one variable constrained by another, it may not come as a surprise that the technique also works just as well for multiple variables. However, for the purposes of this paper, it is important to know only that optimizing variables constrained by others follows generally accepted mathematical principles, and is used throughout everyday activities.

Appendix B: Veloxum's Architecture, Installation, and Operation

Veloxum's Architecture

Three key components make up the Veloxum solution**---**Connectors, Virtual Appliances, and the Management Console.

Connector

The Connector is a lightweight, cross-platform process that identifies the components – applications, databases, hardware, software, and operating systems –installed on the target server or client. The Connector tests and reports the baseline configuration results to the Virtual Appliance. The Connector also implements required configuration changes when it receives instructions from the Appliance. To save resources, Connectors remain dormant until they initiate active tests.

Virtual Appliance

The Virtual Appliance (VA) is a 64bit VMware image that controls Connectors. When first deployed, the VA collects data from the Connectors and automatically creates a baseline. It then compares the baseline against its business rules, identifies optimization opportunities, and creates settings change recommendations. The VA then transmits the configuration change instructions to the Connector. Deployment occurs immediately in automatic mode, or once authorized by IT in manual mode.

Management Console

The Management Console (MC) is a browser-based**,** control center that gives IT complete control over VA instances and provides continuous performance information back from them. Through the MC, IT personnel can set remediation to occur either immediately, on a deployment schedule or when manually set.

The MC provides a rich set of settings-related data not found in other tools. For instance, the MC provides a heat map to instantly inform users which servers are most heavily used and endangering timely performance. In addition, the MC can provide instant graphical readouts on VMware's most critical metrics; % CPU RDY, CPU Usage, Shared Memory Utilization, Network Throughput, and Guest Throughput, to name just a few.

The MC maintains a history of all changes and can roll back any set of changes at any time. Existing enterprise monitoring tools, such as Microsoft's Operations Manager (MOM), can integrate and present MC data.

Installation and Operation

Installing Veloxum within the Enterprise is fast and easy. Installation follows generally accepted procedures and operation requires setting only a few options.

Installation

The Virtual Appliance (VA) Installs as a 1GB 64-bit image that downloads onto an existing VMware server. The Connector installs on all supported devices including local and remote servers, notebooks, and home office systems. It installs via standard software distribution methods and completes within in minutes. Once started, the Connector begins reporting metrics to the VA immediately.

Operation

Veloxum's system and method for optimizing operating systems, and application is a patent-pending process. (An overview of the technology can be found in Appendix A.) In addition to the previously mentioned base-lining, testing, reporting and implementing of fixes, Veloxum also validates implemented improvements, continuously monitors performance, and rolls back changes if desired. This feature combination – active remediation, validation, continuous monitoring, and rollback – set Veloxum clearly apart from other server and application monitoring-only tools.