QUEST SOFTWARE®

Guessing Is Not a Strategy: The Science of Capacity Management

The process and pitfalls of satisfying performance requirements in a cost-effective manner

BY JONATHAN KLICK, VKERNEL VIRTUALIZATION EVANGELIST

WHITE PAPER

Contents

Introduction

The layers of abstraction in virtualized servers establish at once the technology's biggest advantage and its most frustrating challenge. The server's resources become, in effect, universal, allowing them to be shared among multiple operating systems and applications. This dramatically improves resource utilization, which can be as low as 10% for dedicated servers. But in the process, important details about the resources become somewhat obscured from view. This can cause virtual machine (VM) capacity management to become more of a guessing game or trial-and-error process than a science.

By reading this whitepaper, VM administrators will:

- Get answers to some common questions. How am I utilizing current capacity? How much more can I do with the existing capacity? How can I reclaim underutilized capacity? When will I run out of capacity?
- Learn why capacity management gets less effective as the number of servers grows
- Understand why guessing inevitably underutilizes resources through over-provisioning
- Benefit from following a proven six-step process for capacity management
- Discover ways to address the common difficulties in managing capacity
- Improve results when sizing and resizing virtual machines

It is important to note that this white paper is not a definitive guide to capacity management. No white paper could possibly provide so much knowledge in so few pages. What is covered here is both a proven process to follow and the common pitfalls to avoid, which together help make capacity management more of a science and less of a guess.

The Capacity Management Challenge

Capacity management in a virtualized environment is a balancing act between performance and cost savings. The cost savings derive from the ability to run multiple virtual machines on each physical server. But because the VMs compete with one another for the server's finite resources, application performance can degrade when the environment is not configured well.

To guarantee good application performance, most VM administrators over-provision resources. But over-provisioning undermines the cost-savings available through consolidation and virtualization by under-utilizing the host's resources. This tendency to over-provision is a symptom of a lack of visibility into how the virtual machines are actually utilizing the physical resources.

There is also a tendency for over-provisioning to get worse as the number of hosts grows. Figure 1 shows the results of a study VKernel conducted of a half-million virtual machines across 2,500 virtualized environments. As shown, the greater the number of hosts, the fewer the number of VMs per host. The number of VMs drops precipitously at first, and then stabilizes at around half of that achievable on a small scale.

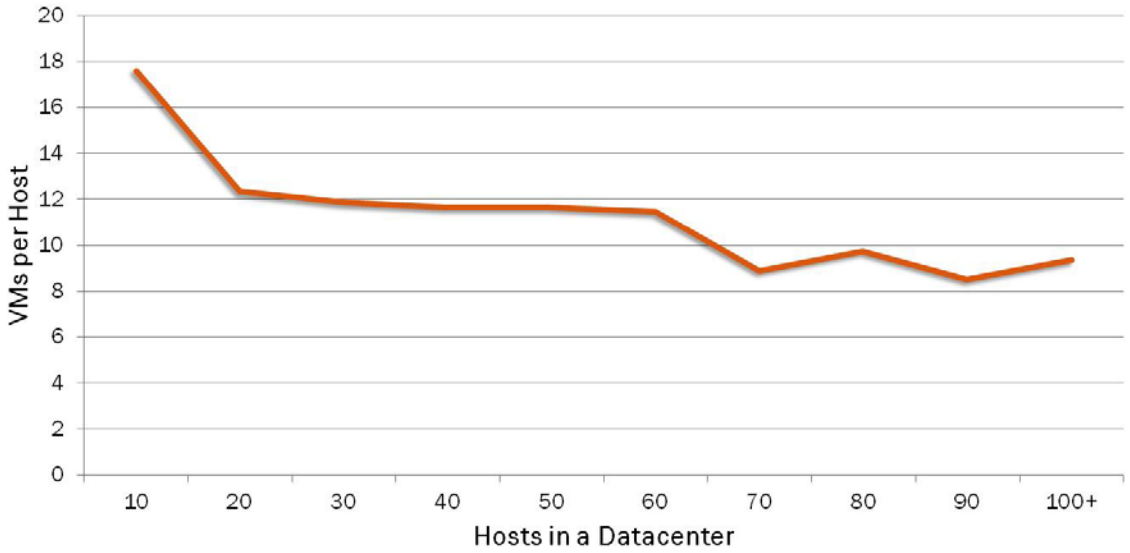

Figure 1 – Benefits Diminish as Virtualization Scales

On a small scale, the lack of visibility into utilization of the underlying physical resources is a manageable challenge. VM administrators can use spreadsheets and other manual processes to track and tune resource allocations reasonably well. But because manual processes fail to scale, they are incapable of continuing to produce satisfactory results in environments with around 20 hosts or more, according to VKernel's study.

The study also reveals there is way too much guessing going on in VM capacity management today, and that those guesses are not achieving the full cost savings possible with virtualization. But the problem goes deeper than not fully utilizing the physical hosts' resources. Growth without efficiency could cause an organization to outgrow its datacenter's physical space, or power or cooling capacity.

Consider just a modest capacity management goal of increasing the number of VMs per host from 10 to 12. That may seem trivial, but it represents a 20% better return on the investment in hosts. And what if a large-scale environment could be managed just as effectively as small one? That would enable nearly doubling the number of VMs running on existing hosts.

By following the capacity management guidelines outlined in this paper, VM administrators should easily be able to achieve the modest 20% improvement, and with the right tools and techniques, might be able to realize a 50% improvement or more—all while maintaining satisfactory levels of performance and honoring all service level agreements. Replacing the guessing all begins with a lesson in Capacity Management 101.

Capacity Management 101

The science of capacity management involves a sequential workflow with six steps, as shown in Figure 2. Although each step is relatively straightforward (with one exception), it is important to follow them in the order shown, as each builds on the results of the previous ones. It is also necessary to repeat the entire process periodically, especially after the virtualized infrastructure and/or application workload change.

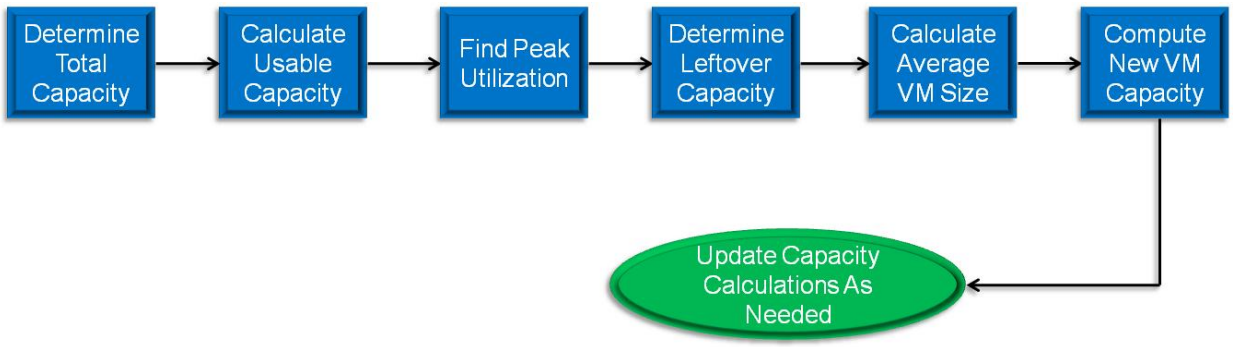

Figure 2 – Capacity Management Workflow

What follows is an example of the capacity management workflow for a cluster of hosts with HA (high availability) enabled. The workflow operates in a similar fashion for other use cases, including for a single host. The HA use case is employed here because it is one of the more difficult ones in capacity management.

Determine Total Capacity – The cluster in this example consists of four hosts in an HA configuration. To be able to recover from the complete failure of any single host, the available total capacity should be based on only three of the hosts. In other words, only 75% of the resources in the cluster should be considered to be available for the total capacity calculation. Note that this assumes all hosts have the identical configuration. In clusters with hosts of varying sizes, it is important to account for the case where the largest one fails.

Calculate Usable Capacity – Having a capacity buffer is prudent to avoid performance problems during periods of peak workloads. For an HA cluster, owing to the built-in "buffer" from the equivalent of an additional or "spare" host, a 5-15% capacity buffer should be adequate. For individual hosts and non-HA clusters, a slightly larger buffer of 15-20% is recommended.

Find Peak Utilization – The critical consideration in this step is to use a period of when a peak workload will actually occur. If that peak will occur beyond the planning horizon (e.g. at the end of a quarter or fiscal year), it may be necessary to make an estimate of the peak resource utilization. It is also necessary to determine peak utilization for all VM resources, including memory, CPU, storage I/O, storage space and the network. Of these five shared resources, memory is the one that is normally the most constrained—and the most difficult to assess.

It is important to note that this is both the most important and the most difficult step in the capacity management workflow. Indeed, many white papers have been written about, and much hypervisor documentation has been devoted to, assessing resource utilization, and there are many third-party products available to complement the capacity management capabilities found in VMware's vSphere, Microsoft's Hyper-V and Red Hat Enterprise Virtualization (RHEV). So the insight gained in this step depends on the tools and techniques used, making further discussion beyond the scope and intent of this white paper.

Determine Leftover Capacity – This simple calculation reveals unused capacity available for other VMs. Leftover capacity is determined by simply subtracting peak utilization from physical capacity for each resource.

Calculate Average VM Size – In most virtualized data centers, it may be necessary to use different averages for different types of applications, such as database, virtual desktop, email, etc. For similar applications, use existing configurations that deliver satisfactory levels of performance to calculate the average VM size needed. For new applications, use the software vendor's recommendation or an industry average (discussed below) to determine the average VM size.

Compute New VM Capacity – This simple calculation reveals the number of additional VMs that can fit in the leftover capacity. It is determined by simply dividing the leftover capacity by the average VM size.

In this simple four-host example, the six-step capacity management workflow is not at all arduous. But what about an environment with 50 or 100 hosts? Owing to the complexity of managing capacity on a larger scale, it will be necessary to consider another factor: how frequently the environment changes. In relatively static environments, reassessing VM capacity can be performed only occasionally, perhaps only once a year during budgeting. But in dynamic environments, it may be necessary to reassess capacity on a monthly or even a weekly basis.

The process can be streamlined, of course, by using good capacity management tools, which also achieve more accurate results. Some tools available automate many of the tasks, enabling a complete reassessment of capacity in just a few hours. Most also provide special capabilities to address the common difficulties encountered when managing capacity.

Difficulties Managing Capacity

While each virtualized environment present its own unique set of difficulties, the three addressed here are fairly common and involve the two resources that normally constrain the number of VMs it is possible to place on any single host: memory and CPU.

Lack of Memory Metrics

As mentioned above, memory is the resource that is normally the most constrained and the most difficult to assess. The reason is the lack of good metrics available in the hypervisors. Merely knowing how much memory is being used can be misleading. While this shortcoming exists in all virtual server solutions, it is useful to consider an example with two such memory metrics in vCenter: average memory active and average memory consumed.

The average memory active metric is collected at the VM level and measures the amount of memory pages that are being actively used by a VM at any given time. The problem is: A VM may not be actively using all of its allocation at a given time. Because the typical VM has more memory allocated than it actually uses at a given time, this metric is not truly representative of how much memory a VM needs.

The average memory consumed metric is a more accurate means for gauging a VM's total memory footprint, but it also has limitations. A portion of the memory consumed always holds data or code that has been used recently, but is not being actively accessed. The problem is some VMs will consume all of the resources allocated to them whether they actually need that allocation or not. So while this may be a more accurate metric, average memory consumed alone does not provide a complete answer for capacity management needs.

Some of the other typical memory metrics available, such as those used to indicate swapping or ballooning, can be quite useful for troubleshooting memory contention problems. But these too have only limited value in capacity management because a complete absence of swapping or ballooning usually means one of two things: The allocation is either perfectly sized (pretty rare!), or it is over-provisioned. And without sufficient metrics, a trial-and-error process of continuing to reduce memory allocations until contention occurs may be the only way to know.

Caching and Dynamic Allocations

Certain desirable features available in most virtual server solutions can also make capacity management more difficult. Consider caching in a database application, for example, that has been allocated 8 gigabytes of memory, but maybe only actually needs 2 or 4. Because the application can cache, it will, which means it will routinely be using its full allocation of 8 gigabytes. This has the effect of skewing the average memory consumed metric in the opposite direction. So while average memory active sometimes understates the amount of memory actually needed by a VM, average memory consumed can overstate the need with caching enabled.

The ability to dynamically move VMs among hosts to better balance overall resource utilization is a popular capability in both vCenter with its Dynamic Resource Scheduler feature and in Hyper-V with its Performance Resource Optimization feature. But from a capacity management perspective, the only purpose these features fulfill is to obscure sub-optimal VM allocations. Moving a VM currently experiencing memory contention to another host that is not, can temporarily alleviate an acute performance problem, but it does nothing to diagnose or cure the underlying chronic problem.

These and other capacity management difficulties are exacerbated by the nuances introduced through the use of allocation shares, reservations and limits in resource pools. The most critical of these for capacity management purposes is the reservation, which is a minimum resource allocation with a default value of zero. Any VM configured with a non-zero reservation dedicates use of that particular resource exclusively to the VM, making it unavailable to all other VMs on its shared host.

CPU Ready—or Not

While memory is the most common cause of contention in host servers, applications also contend for CPU resources. Because CPU utilization at both the VM and host levels is often low, this can mislead administrators into believing there is no CPU contention occurring. There is a second type of contention, however, that has nothing to do with the GHz available, but more to do with the number of virtual CPUs (vCPUs) that are trying to access the much more limited number of physical CPUs (pCPUs). In a typical virtual environment, the ratio of vCPUs to pCPUs is normally in the range of 2:1 to 3:1, but this can vary based on the nature of the applications.

Fortunately there are some metrics available to determine if the virtual-to-physical ratio might be causing performance degradation. In a VMware environment, for example, the best metric to use is CPU Ready. Unfortunately, hypervisor features that dynamically balance workloads across hosts often mask this potential cause of performance problems. VMware's Dynamic Resource Scheduler (DRS), for example, does not take into account the ratio of virtual to physical CPUs while balancing the load. Consider a simple cluster with two hosts, each with dual quad-core processors. Both are operating at 30% CPU utilization, but one host might have 20 virtual CPUs allocated, while the other has 40 running on the same eight physical cores. VMs on the first host might be performing well, but VMs on the second host would be placing a much higher demand on the physical cores, resulting in a higher probability of CPU contention. This lack of awareness by DRS of the number of virtual CPUs is yet another example of why right-sizing virtual machines can be as important as it is difficult.

(Right-)Sizing Virtual Machines

When VMs are correctly sized, it does not matter if they are consuming all of the memory they have been allocated. In fact, a right-sized VM should consume all of its allocated resources and never be in need of any more. Such perfection is rare for all VMs, of course, and is forever unobtainable with over-provisioning. While it may seem counterintuitive, over-provisioning can actually degrade overall performance. This is particularly true with CPU allocations, where VM performance can often be improved by reducing the number of CPUs.

VKernel's survey of a half-million VMs across 2,500 virtualized environments also explored RAM and CPU commitments and virtual-to-physical ratios. The results are depicted in Figure 3, and reveal that most VM administrators are doing pretty well by over-committing (rather than over-provisioning) CPU resources. When it comes to RAM, however, VM administrators are not doing so well.

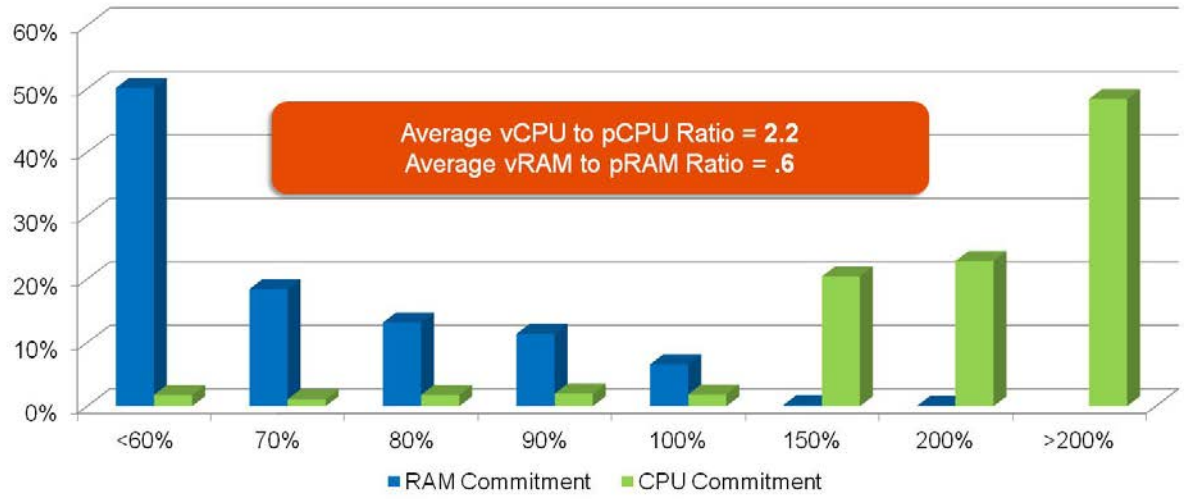

Figure 3 – Industry Allocation Ratios

The results reveal a sort of industry average for allocating VM RAM and CPU resources. While the 2.2 virtual-to-physical CPU ratio is quite good, it is generally acceptable to go as high as a ratio of 3.0, or even higher with applications that are not CPU-intensive.

The more interesting finding here is that the virtual-to-physical for RAM ratio shows that organizations, on average, are committing only 60% of the physical memory available. Even taking into account HA configurations and buffering, this level of commitment falls far short of fully utilizing available memory. With utilization typically being about half of the commitment, this level means that the average organization is actually utilizing only about 30% memory.

Before discussing the workflows for sizing and resizing VMs, it is important to consider one other factor: overhead. Consider the example of a VM allocated with four CPUs and 8 gigabytes of RAM. Such a generous allocation requires an additional 400 megabytes of memory for overhead, and the more the resources, the greater the overhead. Cutting the allocation in half to two CPUs and 4 gigabytes of RAM, also reduces the overhead by nearly one-half. While that may seem trivial for a single VM, for a host, cluster or entire data center, the reduction in overhead can be significant, especially when no additional memory can be added because the hosts are already configured with the maximum supported. The combined effect of right-sizing just a few VMs could make room for one or two additional VMs per host.

Sizing New VMs

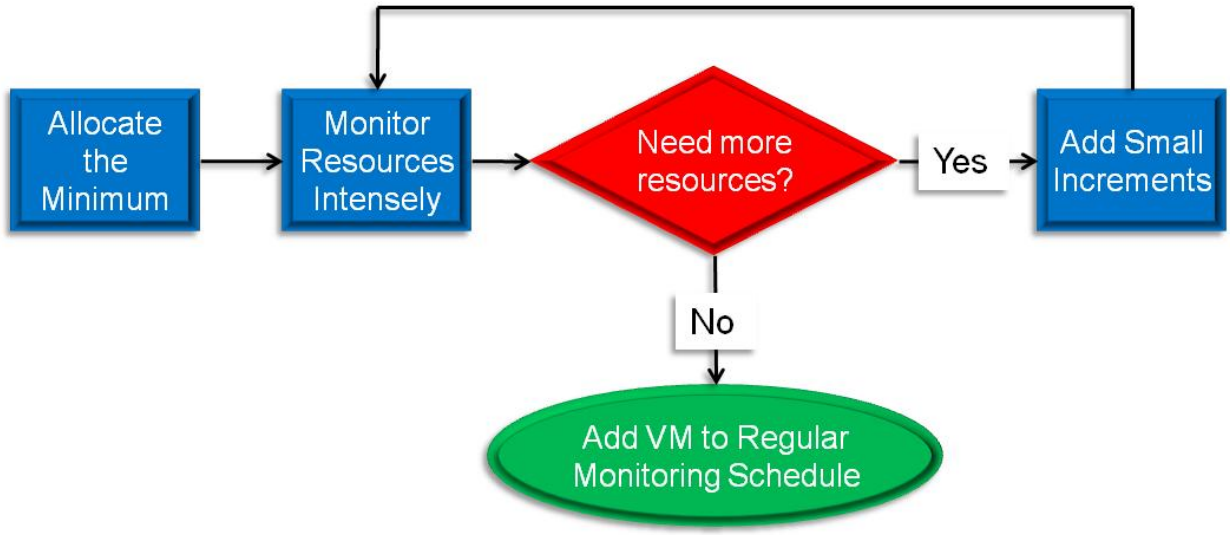

Figure 4 – Sizing New VMs

Perhaps the most important step shown in Figure 4 is the first one encouraging minimal initial allocations. Why? Because users never complain when their allocations are increased, but taking away resources is certain to be met with protest. So when sizing new VMs, allocate a bare minimum of memory and CPU resources. That is easier said than done, of course, because there is tremendous variation in different types of applications running on different operating systems, and users initially request what the application vendor may recommend, which not too surprisingly, is always a generous allocation.

If possible, start out as low as one CPU and 2 gigabytes of memory. If that seems too low, try doubling it to two CPUs (in line with the industry average) and 4 gigabytes of memory. Then monitor resource utilization immediately and intensely when the application goes live, or better yet, in a test setup before going live. If more resources are needed, that should become fairly obvious fairly quickly. Look at the peak utilization values, making sure that the buffers are sufficient so that utilization does not exceed 90%. If more memory or CPU (or other resource) is needed, add it in small increments.

What constitutes “small” will also vary. For example, if peak utilization in an application allocated 1 gigabyte of memory is at 90-95%, another 512 megabytes should be sufficient. But if the VM has 8 gigabytes with the same 90-96% peak utilization, then another 2 to 4 gigabytes might be warranted to provide an adequate buffer and somewhat “future-proof” an application that is expected to grow.

When the application is performing well, add it to a regular monitoring schedule. Whether that regular schedule is weekly, monthly or quarterly should be determined by the nature of the application, its anticipated peak demand and, of course, the time and tools the IT staff possesses to monitor the environment. Also be on the alert for any and all alerts that indicate a performance problem is occurring, and take action before the users start to complain.

While this might seem like a trial-and-error process, it is really more of the iterative and controlled one often used in scientific endeavors. Unknowns abound in any environment, and it is rational to deal with unknowns in manageable steps. For if the objective of capacity management is to maximize resource utilization, this incremental approach is the best way to avoid over-provisioning.

Resizing Existing VMs

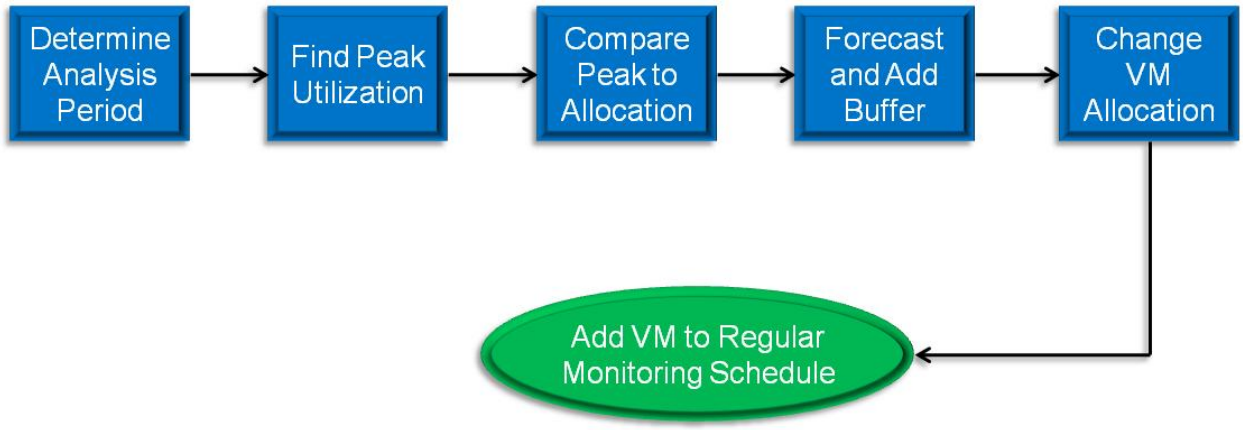

Figure 4 – Resizing Existing VMs

Truth be told, resizing is usually about reducing VM allocations. Resizing becomes necessary when the above "minimalist" approach to sizing a VM was not followed originally, or because there has been a substantial change in the application or its use. For whatever reason(s), every environment has VMs that are over-provisioned. Of course, it is also possible that some applications will change in ways that require an increase in one or more resources. The process described here is designed to right-size any VM, regardless of whether that results in a reduction or increase in resource allocations.

The key to getting good results is to use a valid analysis period; that is, one when the peak demand will actually occur for the VM's application. For this reason it may be useful to have different VMs on different monitoring schedules, such as a certain day of the week, the beginning of each month, or the end of every quarter.

The techniques used to find peak utilization will vary by resource. For memory, for example, use average memory active and average memory consumed, as described on the previous page, and examine how the operating system is using memory. Then "triangulate" these less-than-perfect individual metrics, while applying some insight about the application, to provide a valid assessment of peak utilization.

If the VM is consistently below 75% utilization of any resource during its peak, it is a good candidate for downsizing. If utilization is above 75% but below 90%, it may be necessary to look at how the application is trending. For example, has it remained at 80% during the past year, or has it steadily increased from 75% to 85% over the past quarter? Applications that are growing are good candidates for upsizing as utilization approaches 90%, and applications that are growing at a particularly rapid rate may warrant increasing the buffer as an extra measure of precaution and future-proofing.

Finally, while downsizing, be mindful of an application's minimal resource requirements. For example, Java applications must have a minimum heap size, and memory allocations below that minimum are certain to cause problems.

Conclusion

Capacity management in a virtualized environment is a balancing act between performance and cost savings. Most VM administrators, however, tend to favor performance over cost savings by over-provisioning resources. With good capacity management tools and techniques, a better balance can be achieved regardless of existing conditions or past practices.

Capacity must be managed at both the physical and the virtual levels. All too often, VM administrators focus too much at the physical host and cluster level (often at the direction of management), taking the virtual level as a given that must simply be accepted. But this misses a huge opportunity because the virtual level has a profound impact on the physical level. Consider this issue instead as involving co-dependent macro and micro levels, as this provides a constant reminder of the impact VM sizing at the micro level has on the hosts, clusters and data center at the macro level.

By following the techniques outlined here, combined with the right tools, any IT organization can turn capacity management into more of a science. Even modest objectives for right-sizing and reclaiming capacity can deliver meaningful results. Over time, as confidence in the science of capacity management grows, the objectives can get more ambitious. And someday, with patience and persistence, all of those expensive servers in the data center could well be running nearly twice as many VMs each as they are now.

To learn more about capacity management in a virtualized environment, please visit VKernel on the Web at http://www.vkernel.com. There you will find several capacity management resources including additional white papers, podcasts and blogs on a variety of capacity management topics, as well as many case studies about how other organizations are using VKernel's vOPS™ Server and vOPS Storage family of products better balance performance and cost savings in their virtualized environments. These products provide the monitoring and management tools needed to:

- Identify hosts running out of capacity
- Identify datastores running out of capacity
- Predict future performance problems
- Reserve capacity for planned future VM deployments, then auto-deploy these VMs into reserved slots
- Identify optimal VM placements and additional workload capacity
- Predict future hardware resource requirements

While on the site, be sure to download your free 30-day trial of vOPS Server Standard, an award-winning capacity management and planning tool for vSphere, Hyper-V and RHEV environments.

Updated—[September, 2012]