

Realtime
publishers

Tactics in Optimizing Virtual Machine Disk IOPS

The Essentials Series

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
Article 1: Poor Practices that Hinder VM Disk IOPS.....	1
Poor Practice #1: Overextending SAN-to-Server Connections.....	2
Poor Practice #2: Using Poorly-Performing Disks in High-Load Situations.....	2
Poor Practice #3: Creating VMs with the Wrong Disk Format.....	3
Poor Practice #4: Disk Misalignment.....	3
Poor Practice #5: Neglecting Spindle Count.....	3
Poor Practice #6: Excessive Snapshotting.....	4
Fragmentation: The Hidden Drag on IOPS.....	4
Article 2: The Impact of Fragmentation on VM Disk IOPS.....	5
Fragmentation in Virtual Environments: It Only Gets Worse.....	5
Doesn't Windows Compensate for This?.....	7
Article 3: Defining Requirements for a VM Disk Optimization Solution.....	8
Requirement #1: Fragmentation Prevention.....	8
Requirement #2: Virtual Environment Orchestration of Activities.....	9
Requirement #3: Free Space Optimization.....	9
Requirement #4: Support for Special Disk Types.....	10
VM Disk Optimization: Often Forgotten, Always Necessary.....	10

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.

Article 1: Poor Practices that Hinder VM Disk IOPS

Spend time in enough IT shops, and you'll eventually discover that the same mistakes are made everywhere. At least that's the feeling I get when pondering all the virtual environments I've seen in my consulting travels. From large to small, simplistic to highly advanced, you'd be surprised how often the same poor practices are incorporated into people's designs.

Most interesting about those mistakes, particularly in the case of virtual machine (VM) performance, is how unnoticed they often go. IT shops with heavy-duty hardware experience the classic signs of poor performance *and often don't even realize it*. Others might realize performance isn't to par but focus troubleshooting attentions on entirely the wrong things, such as resources like processing and memory that comprise virtual environments, incorrect configurations, or omitting key technologies the lack of which creates big problems down the road.

Your storage represents one of those oft-forgotten areas where poor VM performance can come from. Too often, storage itself is thought of only in terms of capacity: "I have fifteen terabytes of storage I can provision to virtual machines." Yet today's storage and the demands we put on it requires a second metric that's just as important: *performance*.

Input/Output Operations per Second (IOPS) is a common measurement for quantifying storage performance. In general terms, a unit of IOPS represents how many "things" a storage device can accomplish in a given unit of time. Those things might be reading from a disk or writing to it, deleting data from it, or performing storage maintenance tasks.

The amount of IOPS you have to work with—your supply—is greatly driven by your design. Incorporate faster disks, more storage processors, or a wider connection bandwidth, and you'll see IOPS go up. It is also driven by the collection of decisions you've made in configuring hosts and VMs. Overload your connections, ask too much of your disk spindles, or configure VMs in ways that require more-than-necessary storage attention, and you'll quickly find that IOPS suffers. And when IOPS suffers, so do your VMs.

In my travels, I've seen plenty of poor storage practices. They're laid into place by well-meaning administrators who simply forget that *storage performance is as important as storage capacity*. Let me share a few of my favorite stories from those travels. In the telling, hopefully you'll learn to avoid common poor practices that hinder VM disk IOPS.

Poor Practice #1: Overextending SAN-to-Server Connections

One of my favorite poor performance stories begins during the days of the Great Hypervisor War. Back then, a common conversation among virtual administrators was the debate between Microsoft Hyper-V and VMware ESX as hypervisor of choice. During that time, each side found itself seeking reasons for their side's superiority over the other. It was a raucous time in our industry's past.

Back then, I visited a client's data center to help them track down a performance difference between VMs in their VMware ESX environment and those atop comparable Hyper-V hardware. Spending a day tracing the similarities between the two configurations, I was baffled about why their Hyper-V VMs were an order of magnitude slower than those atop ESX.

It wasn't until late in the day when I realized the difference—one so slight in the client's eyes that they neglected to bring it up until day's end. During their comparison, this client was also introducing themselves to the network implications of iSCSI SAN storage. Their previous experience in TCP/IP networking had them concerned primarily about *connectivity*. The focus on that concern had them forgetting completely the impact of *throughput*.

Turns out their Hyper-V servers in Building A were in fact connected to storage in Building B, traversing a single fibre pair and sharing the bandwidth with that entire building's regular network traffic. Their Hyper-V VMs' demand for IOPS far exceeded their storage connection's available supply.

An easy fix, but the moral of that day is to always remember *storage networking requires more from a network than traditional networking*. Segregating traffic where appropriate and monitoring utilization is critical to preventing an IOPS bottleneck.

Poor Practice #2: Using Poorly-Performing Disks in High-Load Situations

Another client, this one a hospital, found themselves developing an interest in virtualization. Like all hospitals, storage of patient records mandated early on (at the time) powerful SAN equipment. A business that embraced technology's leading edge, this hospital's previous-generation SAN was given a second life in hosting VMs the day its replacement arrived.

When I arrived to troubleshoot the ensuing performance issues, I reminded them that not all SANs are built alike—nor will all SANs perform alike. No matter how many processors or disks you provision to virtual hosts, VMs won't perform well atop previous-generation SATA drives that lack the IOPS virtualization requires. The resolution here: Dump the old SAN and acquire one with an IOPS supply that exceeds VM demands.

Poor Practice #3: Creating VMs with the Wrong Disk Format

Virtual platforms like VMware vSphere's early versions didn't support thin provisioned disks. This was for a reason: Although requiring the use of "thick" disks added costs in wasted disk space, those disks were guaranteed to operate with best performance. It wasn't until much later that waste-conserving thin provisioning was eventually made available.

Yet saving on space with thin provisioned disks doesn't come without a cost. That cost is paid with a slight performance loss, particularly when disks are expanded to add space. The performance difference between thick and thin grows smaller with each new virtual platform version, but some difference still remains today.

Even more insidious are linked disk clones, which begin one disk's life based on the configuration of another. Though linked clones may garner even greater space savings, they do so by paying a tax on performance. Forcing disk activity to exist across what are now two disks instead of one means adding to a VM's IOPS demand.

Poor Practice #4: Disk Misalignment

A physical disk is broken into blocks of data, as is a virtual disk. A block represents the smallest unit of data that can be read from or written to a virtual or physical disk. Blocks can be linearly read from a disk, not unlike a needle following grooves on a record. Sometimes, though, a virtual disk's blocks aren't laid down in alignment with those of its physical host. Instead, they're offset by just a bit, sitting a VM's block now atop two physical blocks. When this happens, reading from or writing to that virtual disk requires extra effort across those two physical blocks.

With the right software, misaligned disks are becoming less of a problem in today's virtual platforms. Not paying attention to them, however, means their extra effort becomes a source of reduced IOPS. Worse yet, they're difficult to track down and even more difficult to fix with native tools alone. Pay particular attention to the approaches used by your software and storage device or suffer the pain of double effort at every read and write.

Poor Practice #5: Neglecting Spindle Count

Another of my favorite stories highlights the peril in focusing on capacity to the exclusion of performance. This tale deals with another client delving into desktop virtualization. The skills required for success here are very much the superset of those for simple server virtualization. There are just so many extra activities required to assure a good experience when users are provisioned virtual desktops.

During the design phase, this client got too excited about recent improvements in storage capacity. Their excitement is understandably warranted, if misguided. With virtual desktops often incurring huge storage costs over the traditional model, bigger disks usually mean smaller dollars-per-gigabyte. Yet compressing more data into the same form factor also compresses more data onto the same number of disk spindles. Insufficient IOPS supply is the natural result, as virtual desktop users vie for data access and disk thrashing ensues.

Disk thrashing will be a problem with desktop virtualization (or, really any workload) when enough spindles aren't brought to bear. This client learned the hard way that dense storage can also be slow storage when placed under too heavy a load.

Poor Practice #6: Excessive Snapshotting

One of virtualization's early promises was the career-protection device VM snapshots could provide. You remember this storyline, "Are you about to install a patch, or change a configuration that could create a problem? Just snapshot the VM first and you've got an instant time machine!"

Snapshots still provide this functionality today; however, snapshots were never intended as a long-term storage mechanism. Repeat this statement to yourself.

One reason for their short-term nature centers around the same problems discussed earlier with linked clones. Creating a snapshot automatically creates another location across which data must be managed. That doubling of data locations adds to storage effort, eventually reducing performance. Layering multiple snapshots atop each other reduces it even further. Reduce unnecessary storage effort by eliminating snapshots. Use them sparingly, and only for short-term needs.

Fragmentation: The Hidden Drag on IOPS

There's a final IOPS impact that any review of poor practices can't conclude without. This hidden drag relates to the extra effort placed on storage when volumes become fragmented. Fragmented volumes, as any Windows administrator should know, shatter individual files and folders into hundreds (or even thousands) of tiny pieces, each of which requires attention and reintegration during every disk access.

That added attention impacts IOPS, and can significantly reduce VM performance. In fact, the extra attention fragmentation requires leads directly into this series' next article, which discusses specifically the impact fragmentation can have on VM IOPS.

Article 2: The Impact of Fragmentation on VM Disk IOPS

Virtualization's early years found many an administrator focusing attention on processor utilization as primary bottleneck. "Insufficient processing power," we thought back then, "creates a shortfall condition. That shortfall translates directly to poor performance."

Those assumptions weren't necessarily incorrect. Lacking processing power, you will experience performance issues. *Today, however, we realize that disk I/O has a far greater impact than ever before realized.* You probably know that VM performance suffers when hardware doesn't supply enough disk IOPS, or when VMs demand too much. But were you aware that VM and virtual environment configurations can have an impact as well? One critically-important facet of the overall configuration story centers around disk fragmentation's impacts on IOPS.

Fragmentation in Virtual Environments: It Only Gets Worse

You've surely heard the fragmentation story before. Fragmentation as an IT problem has been around since, well, IT has been around. That's because fragmentation is a natural byproduct of normal file system operation. It occurs when a file or folder on disk must be spread across multiple, non-contiguous areas. Figure 1 shows the classic example of a disk at three units of time, starting at the top and moving down.

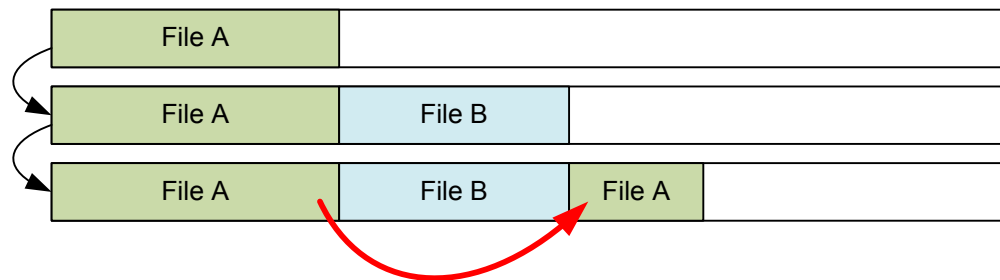


Figure 1: File A must fragment as it grows past its available space.

In this example, File A needs to grow. Perhaps additional data was added or the file was opened, modified, and then closed. File A can't grow contiguously, however, because File B happens to sit in the way. As a result, File A must fragment itself to the next available free space if it is to store its new data.

As files are read and written on disk, this process repeats itself literally tens of thousands of times every week. Files are constantly being added, modified, and deleted, creating “holes” of free space across the entire disk. A deleted file’s hole gets plugged with some other file’s data. When holes aren’t big enough, new data fragments to the next available free space. The problem is a cascading one.

Without protections in place, data can become immediately fragmented as it is written to disk. Existing data fragments further as it evolves, creating a cascading problem where files require incrementally more effort to read and write over time. Figure 2 shows another disk representation, where data in red has grown fragmented from write, modify, and delete operations. Notice the holes in grey. They’re the free space, the “holes” where data will end up next.

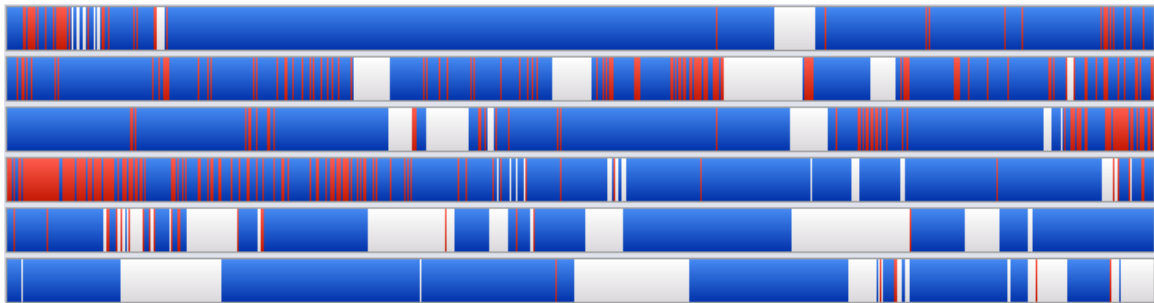


Figure 2: File fragments, represented here in red, get worse over time.

This situation is obviously problematic when experienced across a single computer system. Without compensation, its impact will grow to reduce storage performance, and eventually become a bottleneck for that server’s operations. Now add virtualization to the mix. *This problem multiplies when computer systems are used to host other computer systems*, the exact configuration that defines virtualization.

A VM’s virtual disks can experience fragmentation just like any physical server. The same dynamics of file creation, modification, and deletion that create fragmentation’s performance impact on physical computers follow the same behavior inside VMs as well.

This situation is particularly insidious because it layers fragmentation atop more fragmentation. Figure 3 shows a representation of this multiplicative effect. There, the virtual host is experiencing disk fragmentation. Its files are written at the same time the virtual disks of other computers, combining host fragmentation with VM fragmentation. Significant performance loss is often a result, with VM files requiring extra attention by the VM’s file system—which in turn requires extra attention by the host’s file system.

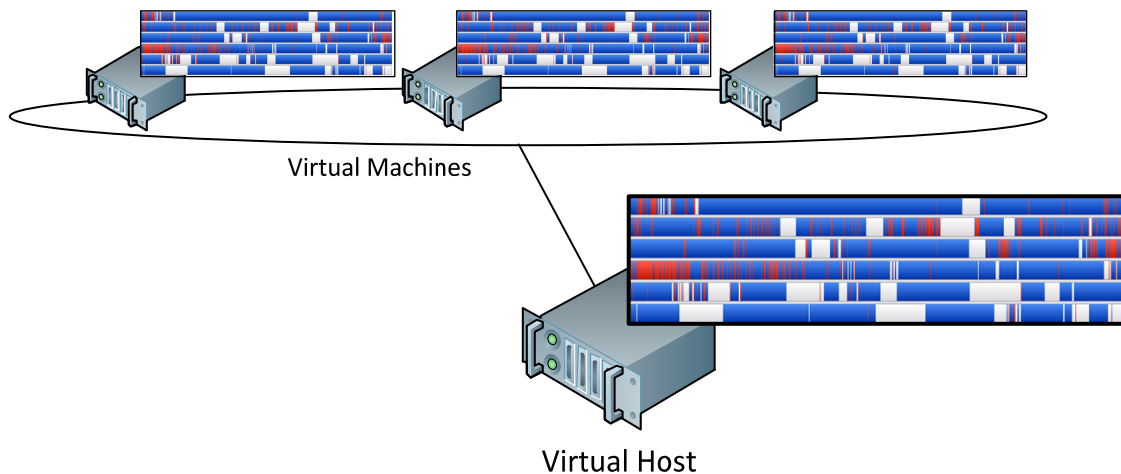


Figure 3: Fragmentation can occur at the host and inside the VMs.

Doesn't Windows Compensate for This?

Doesn't Windows operate with this idea in mind? Indeed it can. Built directly into Windows is a defragmentation feature that will schedule regular passes to reduce fragmentation. This feature is enabled by default on desktop versions of Windows such as Windows 7. It is not enabled, however, on server versions like Windows Server 2008 R2.

This disabled-by-default configuration should beg a question: If fragmentation is such a problem, why would the feature that "fixes" it be disabled by default on server OSs? A primary reason centers around the resource load defragmentation creates. *Eliminating fragmentation by focusing on defragmentation sometimes taxes server resources too greatly.* The extra resources required to even perform defragmentation can themselves impact server operations.

Now multiply these necessary resources across each VM residing on top. You can immediately see how Windows' native approach might not make sense for resource-constrained virtual environments. Of course, all is not lost. This series' final article attempts to find that best-fit solution. In it, you'll learn more about the holistic requirements a virtualization-friendly disk optimization must fulfill.

Article 3: Defining Requirements for a VM Disk Optimization Solution

This series has highlighted the design tactics for best optimizing VM disk IOPS. Following the principles presented in the first two articles will ensure your hardware best meets the demands of your VMs. This series has also explored the specific issue of fragmentation that exists irrespective of how your design is ultimately constructed. Getting to maximum IOPS for VMs requires seeking that best design. It also requires a careful look at the configurations you apply to VMs and virtual hosts so that their activities don't foment poor performance. Fragmentation and its elimination are components of those configurations. More importantly, however, is the recognition that solutions for disk optimization may be a necessary part of your overall design.

Disk optimization, particularly with virtual environments, is more than just defragmentation. Performing disk optimization correctly also requires optimizing free space—those proverbial “holes” inside disks. It requires fragmentation prevention, stopping the problem before it happens. It also demands an orchestration of activities across host and collocated VMs, ensuring that optimization activities themselves don't become an impact on performance. As you look towards options for disk optimization, consider the following four important requirements as your specifications for a virtualization-friendly solution.

Requirement #1: Fragmentation Prevention

Defragmentation is by nature a reactive solution to a naturally-occurring problem. The “de” in defragmentation highlights the fact that such solutions must first wait for the problem to occur. Only after fragmentation has occurred can a defragmentation solution begin cleaning up the mess.

A central problem with the reactive approach lies in the effort required to reverse the damage once done. Run a defragmentation pass weekly, and you've got a week's worth of harm to undo. Just cleaning up the mess requires disk attention that impacts IOPS supply. That attention will get in the way of regular VM operations.

Unfortunately, *timing isn't everything with the reactive approach*. Run it hourly, and the problem's scope might grow smaller, yet the effort requires more regularity. Those are still resources lost. The tradeoff between the amount of damage done and the period between resolution can never really find a functioning balance.

These inefficiencies in balancing time period and effort suggest that a proactive approach might be superior. In layman's terms, a proactive approach is akin to running defragmentation constantly, with the elimination activities occurring at the very moment data is changed. This fragmentation prevention approach reduces the extra effort placed on storage by simply laying down data correctly the very first time. Your selected optimization solution will benefit from being proactive.

Requirement #2: Virtual Environment Orchestration of Activities

Alas, fragmentation prevention alone cannot eliminate every fragment. Even the most-intelligent software solution can never know exactly what "holes" will be necessary at which locations every time. Computers are deterministic, and sometimes users find themselves creating large files or making unexpected changes. Although fragmentation prevention by itself should resolve many issues, there occasionally comes the need for a small amount of extra effort fostered through classic defragmentation.

That classic defragmentation will always have an impact on server operations, whether physical or virtual. Rare, however, is the server that finds itself at 100% utilization all the time. Rarer still is the virtual host with VMs doing the same. It stands to reason, then, that a second solution requirement mandates an intelligent orchestration of activities across virtual host and collocated VMs.

Such a solution should analyze existing server activities to find the time periods of least use. That same approach can analyze activities among all VMs on a host, ensuring that optimization activities in one VM won't cause impacts across others. Physical resources are finite. As such, the best optimization solution will perform its job with a holistic awareness of activities across every part of the virtual environment.

Requirement #3: Free Space Optimization

Recall from the previous article's Figure 1 that fragmentation happens when available free space isn't large enough to support a file's expansion or in writing a new file. When the "hole" on disk hasn't been properly sized, a fragment occurs along with the subsequent need for defragmentation.

A third solution requirement recognizes that free space optimization doesn't necessarily mean creating large unused areas at the back of the disk. It means intelligently sizing holes on disk so that files can naturally expand and be added without automatically fragmenting.

Requirement #4: Support for Special Disk Types

Virtual environment disks also have special needs beyond the physical. Virtual disks come in many forms, each of which requires additional attention if they are to be fully optimized. Two of these forms merit special attention, as lacking compensation for their behaviors can create performance issues no less problematic than fragmentation.

The first disk type requiring attention is the thin provisioned disk discussed in the first article. Thin provisioned disks are designed to start small, only growing when new data requires it to expand. They're great for conserving storage space, at least at first. Yet one problem not well understood by many administrators relates to *what happens when data is removed from these disks*. Thin provisioned disks are indeed designed to grow, but they're not designed to shrink when data removal occurs. Lacking a solution for shrinking such disks, your thin provisioned disks will only keep getting bigger over time. Thus, the first half of Requirement #4 suggests seeking a disk optimization solution that resolves this gap. Such a solution will compact virtual disks after data is removed, ensuring the lowest quantity of wasted space on expensive SAN disks.

A second special disk type is the linked clone mentioned earlier, sometimes also called a differencing disk. These special disks aren't for every application, but they do provide specific benefit in certain circumstances. A common use is for hosted virtual desktops. Being able to provision hosted virtual desktops as linked clones of a central reference image enables the rapid deployment of similar VMs.

Linked clones indeed begin their lives as extremely small files. How different really are two computers that are similar in everything but name? What's not well known is that these disks can quickly grow in size, sometimes even growing to equal the size of their parent disk. This rapid sizing of linked clones stems from many factors, including temporary file storage, creation and deletion of profiles, in addition to fragmentation. Thus, for environments making use of these special disks, the second half of Requirement #4 advises seeking a disk optimization solution that compensates for the rapid sizing behavior of linked clones.

VM Disk Optimization: Often Forgotten, Always Necessary

Disk optimization in virtual environments is absolutely a necessary activity. That optimization comes in many forms. A proper design goes far in ensuring hardware is ready to support the IOPS demand of needy VMs. Correctly configuring those VMs during operations represents another facet.

Incorporating disk optimization tools and tactics comprises the third—and too often forgotten—piece to this story. Lacking disk optimization tools, a virtual environment can find itself losing performance and *you might not even realize it*.

This Essentials Series was brought to you by: Diskeeper

Diskeeper's family of innovative products are relied upon by more than 90% of Fortune 500 companies and more than 67% of The Forbes Global 100, as well as thousands of enterprises, government agencies, independent software vendors (ISVs), original equipment manufacturers (OEMs) and home offices worldwide. Inventors of the first automatic defragmentation in 1986, Diskeeper pioneered a new breakthrough technology in 2009 that actually prevents fragmentation.