

- [Home](#)
- [About](#)
- [Resume](#)
- [Virtualization Jobs](#)
- [Virtualization Resources](#)

## VMtoday

VMware News, Views, & How-To's from Josh Townsend

« [Vote Now for Top Virtualization Bloggers](#)

[Right-sizing Your Power and Cooling](#) »

## Storage Basics – Part III: RAID

January 6th, 2010 | Author: [Joshua Townsend](#)

This is the third in a multi-part series on storage basics. I've had some good feedback from folks in the SMB space saying that the [first](#) couple posts in this series have been beneficial, so we'll be sticking with some basic concepts for another post or two before we dive into some nitty-gritty details and practical applications of these concepts in a VMware environment. In the [second post](#) of this series I introduced the concept of IOPS and explained how the physical characteristics of a hard disk drive determine the theoretical IOPS capability of a disk. I then noted that you can aggregate disks to achieve a greater number of IOPS for a particular storage environment. Today, we will look at just how you combine multiple disks and the performance impact of doing so. Remember that we are keeping this simple; the concepts I present here may not apply to that fancy new SAN you just purchased with your end-of-year money or the cheap little SATA controller on your desktop's motherboard (not that there's anything wrong with it) – we're more in the middle ground of direct attached storage (DAS) as we firm up concepts.

[6](#)  
[tweets](#)  
[retwee](#)

Enterprise servers and storage systems have the ability to combine multiple disks into a group using Redundant Array of Independent Disks (RAID) technology. We'll assume a hardware RAID controller is responsible for configuring and driving storage IO to the connected disks. RAID controllers typically have battery-backed cache (we'll talk cache in a future post), an interconnect where the drives plug in, such as SCSI or SAS (we'll talk about these too in a future post), and hold the configuration of the RAID set including stripe size and RAID level. The controller also does the basic work of reading and writing on RAID set – mirroring, striping, and parity calculations. There are several different types of RAID level – rather than rehash the details of them, [read the Wikipedia entry on RAID](#) and then come back here....

Ok, great. So you now know that RAID is implemented to increase performance through the aggregation of multiple disks, and to increase reliability through mirroring and parity. Now let's consider the performance implications of some basic RAID levels. As with many things in the IT industry, there are trade-offs: security vs. usability, brains vs. brawn, and now performance vs. reliability. As we increase reliability in a RAID array through mirroring and parity, performance can be impacted. This is where the **more disks = more IOPS** bit starts to fall apart. The exact impact depends on the RAID type. Here are some examples of how RAID impact the maximum theoretical IOPS using the most common RAID levels, where:

$I$  = Total IOPS for Array (note that I show Read and Write separately)

$i$  = IOPS per disk in array (based on spindle speed averages from [Part II: IOPS](#))

$n$  = Number of disks in array

$r$  = Percentage of read IOPS (calculated from the Average Disk Reads/Sec divided by total Average Disk Transfers/Sec in your Windows Perfmon)

$w$  = Percentage of write IOPS (calculated from the Average Disk Writes/Sec divided by total Average Disk Transfers/Sec in your Windows Perfmon)

### **RAID0 (striping, no redundancy)**

This is basic aggregation with no redundancy. A single drive error/failure could render your data useless and as such it is not recommended for production use. It does allow for some simple math:

$$I = n * i$$

Because there is no mirroring or parity overhead, theoretical maximum Read and Write IOPS are the same.

### RAID 1 & RAID10 (mirroring technologies):

Because data is mirrored to multiple disks

$$\text{Read } I = n * i$$

For example, if we have six 15k disks in a RAID10 config, we should expect a theoretical maximum number of IOPS for our array to be  $6 * 180 = 1080$  IOPS

$$\text{Write } I = (n * i) / 2$$

### RAID5 (striping with a single parity disk)

$$\text{Read } I = (n - 1) * i$$

Example: Five 15k disks in a RAID 5 (4 + 1) will yield a maximum IOPS of  $(5 - 1) * 180 = 720$  READ IOPS. We subtract 1 because one of the disks holds parity bits, not data.

$$\text{Write } I = (n * i) / 4$$

Example: Five disks in a RAID 5 (4 + 1) will yield a maximum IOPS of  $(5 * 180) / 4 = 225$  WRITE IOPS

Again, these formulas are very basic and have little practical value. Furthermore, it is seldom that you will find a system that is doing only reads or only writes. More often, as is the case with typical VMware environments, reads and writes are mixed. An understanding of your workload is key to accurately sizing your storage environment for performance. One of the workload characteristics (we'll explore some more in the future) that you should consider in your sizing is the percentage of read IOPS vs. the percentage of write IOPS. A formula like this gets you close if you want to do the math for a mixed read/write environment in a RAID5 set:

$$I = (n * i) / (r + 4 * w)$$

Example: a 60% read/40% write workload with five 15k disks in a RAID5 would provide  $(5 * 180) / (.6 + 4 * .4) = 409$  IOPS.

The previous examples have all been from the perspective of the storage system. If we take a look at this from the server/OS/application side, something interesting shows up. Let's say you fired up Windows perfmon and collected Physical Disk Transfers/sec counters every 15 seconds for 24 hours and analyzed the data in Excel to find the 95th Percentile for total average IOPS (this is a pretty standard exercise if you are buying enterprise storage array or SAN). Let's say that you find that the server in question was asking for 1000 IOPS at the 95th Percentile (let's stick with our 60% read/40% write workload). And finally, let's say we put this workload on a RAID5 array. That's saying a lot of stuff, but what does it all mean? Because RAID5 has a write penalty factor of 4 (again, Duncan Epping's posted [a great article here](#) which I referenced in Part II that describes this in a slightly different way) we can tweak the previous formula to show the IO's to the backend array given a specific workload.

$$I = \text{Target workload IOPS}$$

$$f = \text{IO penalty}$$

$$r = \% \text{ Read}$$

$$w = \% \text{ Write}$$

$$IO = (I * r) + (I * w) * f$$

Our example then looks like this (remember work inside parenthesis first, and then My Dear Aunt Sally):

$$(1000 * .6) + ((1000 * .4) * 4) = 2200$$

Simply stated, this means that for every 1000 IOPS that our workload requests from our storage system, the backing array perform 2200 IO's, and it better do it quickly or you will start to see latency and queuing (we call this performance degradation, boys and girls!). Again, this is a very simplistic approach neglecting factors like cache, randomness of the workload, stripe size, IO size, and partition alignment which can all impact requirements on the backend. I'll cover some of those later.

As you can hopefully see, the laws of physics combined with some simple math can provide some pretty useful numbers. A basic understanding of your array configuration against your workload requirements can go a long way in preventing storage bottlenecks. You may also find that as you consider the cost per disk against various spindle speeds, capacities and RAID levels that you are better off buying smaller, faster, fewer, more, slower.... disks depending on your requirements. The geekier amongst us could even take these formulas and some costs per disk and hit up Excel Goal Seek to find the optimal level, but that's more than this little blog can do for you today.

Before I wrap up this post, I want to leave you with a few more links that I have bookmarked around the topics of IOPS and RAID over the past several years:

- DB sizing for Microsoft Operations Manger, includes a nice chart with formulas similar to the ones I provided in this article: <http://blogs.technet.com/jonathanalmquist/archive/2009/04/06/how-can-i-gauge-operations-manager-database-performance.aspx>
- An Experts Exchange post with some good info in the last entry on the page (subscription required) [http://www.experts-exchange.com/Storage/Storage\\_Technology/Q\\_22669077.html](http://www.experts-exchange.com/Storage/Storage_Technology/Q_22669077.html)
- A Microsoft TechNet article with storage sizing for Exchange – a bit dated but still applicable: [http://technet.microsoft.com/en-us/library/aa997052\(EXCHG.65\).aspx](http://technet.microsoft.com/en-us/library/aa997052(EXCHG.65).aspx)
- A simple whitepaper from Dell on their MD1000 DAS array – easy language to help the less technical along: <http://support.dell.com/support/edocs/systems/md1120/multlang/whitepaper/SAS%20MD1xxx.pdf>
- A great post that uses some math to show performance and cost trade-offs of RAID level, disk type, and spindle speed. <http://www.yonahruss.com/architecture/raid-10-vs-raid-5-performance-cost-space-and-ha.html>
- Another nifty post that looks at cost vs. performance vs capacities of various disks speeds in an array <http://blogs.zdnet.com/Ou/?p=322>

### Related posts:

1. [Storage Basics – Part V: Controllers, Cache and Coalescing](#)
2. [Storage Basics – Part VI: Storage Workload Characterization](#)
3. [Storage Basics – Part IV: Interface](#)
4. [Storage Basics – Part VII: Storage Alignment](#)
5. [Storage Basics – Part II: IOPS](#)

Posted in [VMware](#) | Tags: [calculate](#), [calculations](#), [disk](#), [IOPS](#), [perfmon](#), [performance](#), [raid](#), [SAN](#), [sizing](#), [Storage](#), [virtualization](#)

### 9 Responses to “Storage Basics – Part III: RAID”

- [Tweets that mention Storage Basics – Part III: RAID | VMtoday -- Topsy.com:](#)  
January 7, 2010 at 2:18 am

[...] This post was mentioned on Twitter by joshuatownsend, WillHamann. WillHamann said: RT @joshuatownsend: New VMtoday.com post: Storage Basics – Part III: RAID <http://cli.gs/Hyt2m> #vmware [...]

[Reply](#) [More from author](#)



- [Sumesh:](#)  
[January 7, 2010 at 2:20 am](#)

Excelent one dude,i was lookign for this

[Reply](#)



- [Matthew Shuter:](#)  
[January 7, 2010 at 11:08 am](#)

Raid 0 portion, you have “This is basic aggregation with now redundancy”  
now should be no

[Reply](#)



- [Joshua Townsend:](#)  
[January 7, 2010 at 11:32 am](#)

Good catch – I have updated the post.

[Reply](#) [More from author](#)

- [uberVU - social comments:](#)  
[February 3, 2010 at 2:14 am](#)

#### **Social comments and analytics for this post...**

This post was mentioned on Twitter by joshuatownsend: New VMtoday.com post: Storage Basics – Part III: RAID  
<http://cli.gs/Hyt2m> #vmware...

[Reply](#) [More from author](#)

- [Storage Basics – Part V: Controllers, Cache and Coalescing / VMtoday:](#)  
[March 24, 2010 at 7:41 am](#)

[...] I introduced IOPS and the math that goes into calculating the IOPS capacity of a disk array. In Part III we considered a RAID implementation’s impact on performance and availability. And most [...]

[Reply](#) [More from author](#)

- [cmdln.org \(a sysadmin blog\) » Blog Archive » Analyzing I/O performance in Linux:](#)  
[May 17, 2010 at 10:41 am](#)

[...] <http://vmtoday.com/2010/01/storage-basics-part-iii-raid/> [...]

[Reply](#) [More from author](#)



- [karthik:](#)  
[July 23, 2010 at 7:07 am](#)

Hi,

In the above IOPS calculation begining (RAID5) its been said for 5 disk you take it as four reason being one disk for Parity.

But actually RAID 5 doesnt have a separate disk for just holding the Parity. It takes a junk of each disk in RAID 5. So in the case the calculation should be  $5 \times 180 = 900$  IOPS (ideally) and not  $(5-1) \times 180 = 720$ .

Am I right?

[Reply](#)[Joshua Townsend:](#)[July 24, 2010 at 4:00 pm](#)

Karthik – Thanks for the comment. RAID5 is indeed distributed parity – no single disk holds all parity data. Parity bits are read only during rebuilds and so they do not impose a penalty during normal read operations. This means that for reads, your calculation of a RAID5 set with 5 15k disks ( $5 \times 180 = 900$  IOPS) is correct.

Your formula for writes, however, is not correct. For writes, the actual work of writing parity imposes a penalty – 4 IOPS are consumed to write data and parity for each I/O operation requested by your server/OS/application/workload – the “factor of 4” I wrote about. This penalty takes away from the IOPS available to your real workload (we call this overhead). In reality, your 5-disk RAID5 could write a total of 900 IOPS ( $5 \times 180 = 900$ ) before queuing. However, your workload can’t take advantage of all 900 IOPS because most of the IOPS go to reading-modifying-writing the parity bits. Put another way, the disks are busy reading, modifying and writing parity bits instead of writing your workload’s “real” data. Your workload only gets the left-over IOPS after the parity writing overhead has been consumed.

Summing it up, your formula for calculating RAID5 **read** IOPS is  $(\text{Number of Disks}) \times (\text{IOPS per disk}) = \text{READ IOPS}$ . Your formula for calculating RAID5 **write** IOPS available to your server/OS/application/workload is  $(\text{Number of Disks}) \times (\text{IOPS per disk}) / 4 = \text{WRITE IOPS}$ .

Keep in mind this is all theoretical – many arrays use cache, coalescing, and other tricks to minimize the RAID5 write penalty.

Hope this helps,  
Josh

[Reply](#) [More from author](#)

## Leave a Reply

<input type="text"/>	Name (required)
<input type="text"/>	Mail (will not be published) (required)
<input type="text"/>	Website

Additional comments powered by [BackType](#)

Recent Posts

- [VMware vExpert](#)
- [EMC Virtual Storage Integrator Update](#)
- [Storage Basics – Part VII: Storage Alignment](#)

- [Free SAN Monitor for DS3300, MD3000i and others](#)
- [Washington DC VMware User Group – April 29th](#)
- [Storage Basics – Part VI: Storage Workload Characterization](#)
- [Update: SVGA Drivers on Windows 2008 R2 and Windows 7](#)

## Tags

[3.5](#) [2009](#) [best practices](#) [client](#) [compatibility](#) [disk](#) [DL380](#) [ecosystem](#) [EMC](#) [ESX](#) [esxi](#) [esxtop](#) [fusion](#) [G3](#) [HP](#) [Hyper-v](#) [I/O](#) [IOPS](#) [memory](#) [Microsoft](#) [NetApp](#)  
[network](#) [Patch](#) [performance](#) [powershell](#) [reporting](#) [SAN](#) [security](#) [sizing](#) [sql](#) [statistics](#) [Storage](#) [Update 3](#) [vcenter](#) [VI3](#) [virtual](#) [virtual center](#)  
[virtualization](#) [Virtual Machine](#) [VM](#) [VMUG](#) [VMware](#) [vmworld](#) [vsphere](#) [windows](#)

## Categories

- [Documentation](#) (2)
- [EMC](#) (2)
- [General IT](#) (10)
- [Issues & Troubleshooting](#) (18)
- [Microsoft](#) (8)
- [NetApp](#) (3)
- [News](#) (6)
- [Personal](#) (3)
- [Product Releases](#) (2)
- [Storage](#) (8)
- [Storage Basics](#) (3)
- [Uncategorized](#) (3)
- [VMUG](#) (2)
- [VMware](#) (46)
- [VMware How To](#) (18)
- [VMworld](#) (3)

## Recent Tweets

- No public Twitter messages.

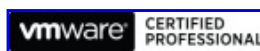
## Recent Bookmarks

- [Windows Guest Defragmentation, Take Two « Pivot Point](#)
- [Dan Weiss Blogsite » NetApp performance testing engineer tells it like it is](#)
- [Understanding how storage design has a big impact on your VDI! - Ruben Spruijt - BrianMadden.com](#)
- [Performance Troubleshooting VMware vSphere – Storage | virtual insanity](#)
- [PowerCLI – Where do I start ?! « Virtu-Al](#)
- [CPU/MEM Reservation Behavior » Yellow Bricks](#)
- [Refreshing Test Dev SQL Environments using NetApp Technologies · Rajeev Karamchedu](#)
- [Finding thin-provisioned virtual disks with PowerShell | VCritical](#)
- [ESXi4 Console - Secret Commands for Emergency Maintenance](#)
- [VMware Server on a Domain Controller \(DC\) « Derek's Musings](#)
- [How to Enable SSH in ESXi – VMware « Code Ghar](#)
- [VMware ESX, EMC CLARiiON Arrays, and Multiple Protocols - blog.scottlowe.org - The weblog of an IT pro specializing in virtualization, storage, and servers](#)
- [PVSCSI and Low IO Workloads « Pivot Point](#)
- [VDI.com - Why should I provision VDI VM's with less RAM and other resources than my physical machines?](#)
- [The old pain in the DAS « StorageTexan's Blog](#)

## About Me



Hello, and thank you for visiting VMtoday. My name is Josh Townsend. I am a technology professional with a strong background in VMware Virtualization, Storage, and Microsoft technologies. I am a Sr. Systems Administrator at [Tiber Creek Consulting](#) in Fairfax, VA, and hold several technical certifications, including VMware Certified Professional. I am also a 2010 VMware vExpert.



I am also leader of the [Washington DC Metro Area VMware User Group \(VMUG\)](#).



The opinions expressed on this site are my own and may not reflect the views of my employer, VMware, or any other party unless otherwise stated.

Please feel free to follow me on Twitter  
[@joshuatownsend](#)

## Archives

- [July 2010](#) (2)
- [June 2010](#) (1)
- [May 2010](#) (1)
- [April 2010](#) (2)
- [March 2010](#) (3)
- [February 2010](#) (1)
- [January 2010](#) (4)
- [December 2009](#) (3)
- [November 2009](#) (3)
- [September 2009](#) (7)
- [August 2009](#) (2)
- [July 2009](#) (1)
- [June 2009](#) (6)
- [March 2009](#) (4)
- [January 2009](#) (4)
- [December 2008](#) (2)
- [November 2008](#) (2)

## VMware Related Sites

- [boche.net – VMware Virtualization Evangelist](#)
- [Chris Wolf](#)
- [Gabes Virtual World](#)
- [Gerben's Blog On Virtualization](#)
- [IPMer](#)
- [Ken's Virtual Realty](#)
- [Mike Laverick's RTFM Eduction](#)
- [Planet V12n](#)
- [Rene Jorissen's Booches.nl](#)
- [Run-Virtual](#)
- [Scott Lowe's Blog](#)
- [The VMguy](#)
- [vinternals](#)
- [Virtu-Al](#)
- [Virtualization.info](#)
- [VM /ETC](#)

- [VM Guru](#)
- [VMtoday Virtualization Resources](#)
- [VMware](#)
- [VMware Land](#)
- [VMware vLaunchpad](#)
- [vSphere Land \(Eric Siebert\)](#)

## Virtualization Jobs

**Virtualization Jobs**

[Server Administrator](#)  
[New Albany, IN](#) - Floyd Memorial Hospital & Health Services

[DAS - SDCOE CLOUD Infrastructure Architect](#)  
[Louisville, KY](#) - IBM

[DRA - Advisory \(SCON\) IT Architect](#)  
[Louisville, KY](#) - IBM

[DRA - Sr. \(SCON\) IT Architect](#)  
[Louisville, KY](#) - IBM

[DRA - Executive \(SCON\) IT Architect](#)  
[Louisville, KY](#) - IBM

[System x FTSS](#)  
[Louisville, KY](#) - IBM

[Custom Migration Windows Architect](#)  
[Louisville, KY](#) - IBM

[Field Technical Sales Specialist \(Must Reside In the Boston / Hartford Area\)](#)  
[Louisville, KY](#) - IBM

[Midrange Storage Virtualization Solutions Technical Consultant](#)  
[Louisville, KY](#) - IBM




[Server Systems Administrator](#)  
[Jeffersonville, IN](#) - Heartland Payment Systems

[See All Jobs](#)




[Post a job for \\$25](#)

Jobs by [SimplyHired](#) [+ Get Widget](#)

## Virtualization Resources

	<a href="#">Windows Server 2008 H...</a> Rand Morimoto, Jeff Guillet... <b>\$32.84</b> ★★★★☆
	<a href="#">Windows Server 2008 H...</a> Robert Larson, Janique Ca... <b>\$37.79</b> ★★★★★
	<a href="#">MCTS: Windows Server ...</a> William Panek (Paperback ... <b>\$40.49</b> ★★★★★



	<a href="#">Virtualization: The Complete Guide</a> Ivanka Menken, Gerard Blom <b>\$26.95</b> ★☆☆☆☆
	<a href="#">Virtualization for Security</a> John Hoopes (Paperback - ...) <b>\$53.95</b> ★★★★☆
	<a href="#">Win Svr Std with o Hyp...</a> Microsoft Software <b>\$1,169.99</b>
	<a href="#">Windows Server 2008 H...</a> John Kelbley, Mike Sterlin... <b>\$31.49</b> ★★★★★
1 2 3 4 >	
 <a href="#">Get Widget</a> <a href="#">Privacy</a>	

Powered by [WordPress](#) and [WordPress Theme](#) created with Artisteer by [Josh Townsend](#).