

Design storage for desktop virtualization properly

I've spent the last month or so discussing the top 10 mistakes seen on desktop virtualization implementation so you can learn from other's mistakes. I've discussed 9 different things so far and they were:

10. [Not calculating user bandwidth requirements](#)
9. [Not considering the user profile](#)
8. [Lack of Application Virtualization Strategy](#)
7. [Improper Resource Allocation](#)
6. [Protection from Anti-Virus](#)
5. [Managing the incoming storm](#)
4. [Not Optimizing the Desktop Image](#)
3. [Not spending your cache wisely](#)
2. [Using VDI Defaults](#)

And now it is time for the #1 thing that people mess up with desktop virtualization? Storage (**SURPRISE**).

Actually, this shouldn't be a surprise at all if you pay any attention to the blogosphere. There have been numerous discussions on this exact topic. If you haven't yet, I'd suggest you read the white paper by [Ruben Spruijt](#) and [Herco Van Brug](#) [Storage Deep Impact](#).

Why all of the confusion and misconfiguration of storage? When it comes to storage, most people think only about size... How much space do I need to allocate per user? With desktop virtualization, storage goes beyond simple size calculations. Virtual desktops rely on the storage infrastructure to load different parts of the operating system and user environment. Each one of the requests impacts the storage infrastructure. Without a properly designed storage subsystem, a user's virtual desktop will slow down to the point of becoming unusable because the storage becomes a bottleneck.

In order to properly design the storage infrastructure, the architect must first be able to calculate the expected Input/Output Operations per Second (IOPS) requirements. The IOPS calculations must take the following into account:

Parameter	Description	Values
Disk Speed	The speed that the disk spins has a direct impact on how fast the disk can read the correct sector.	15,000 RPM: 150 Random IOPS 10,000 RPM: 110 Random IOPS 5,400 RPM: 50 Random IOPS
Read/Write	IOPS are broken down into reads and writes. Certain processes are read intensive while others require more writes. The ratio between reads and writes impacts of the overall IOPS.	It has been shown that most desktop implementations result in the follow read/write ratios: Reads: 20% Writes: 80%
RAID Level	The RAID configuration impacts how many actual write IOPS are available due to the different types of redundancy in place. The write penalty reduces the overall IOPS for each disk spindle.	RAID 0: No RAID Penalty RAID 1: Penalty of 2 RAID 10: Penalty of 2 RAID 5 (4 disks): Penalty of 4
Desktop Lifecycle	Each desktop goes through six phases, with each phase incurring different hits on the storage subsystem.	Boot: 26 IOPS Logon: 14 IOPS Work:

		Light: 4-8 IOPS
		Normal: 8-12 IOPS
		Power: 12-20 IOPS
		Idle: 4 IOPS
		Logoff: 12 IOPS
		Offline: 0 IOPS

Taking the six different parameters into account, an architect can calculate the IOPS requirements on a server-by-server basis and for the entire desktop virtualization architecture. The formula is as follows:

Total Raw IOPS=Disk Speed IOPS*# Of Disks

For example, if there are eight 72GB 15K SCSI3 drives in a RAID 10 configuration, the storage would have 720 functional IOPS.

Functional IOPS=(((Total Raw IOPS×Write %))/(RAID Penalty))+(Total Raw IOPS×Read %)

With the functional IOPS calculated for the disk subsystem, the number of desktops that can be supported is based on the phase of the desktop lifecycle. Identifying how many desktops can be simultaneously logged in with this configuration is as follows:

Total Raw IOPS=150×8=1200

Functional IOPS=(((1200×.8))/2)+(1200×.2)= 720

These calculations help identify what is possible when all desktops are doing the same activity. However, this is not likely to be the case. In fact, on each hypervisor, a portion of the desktops will be in one of the defined phases. So you, as the architect, must figure out what each server will require based on the loads experienced by each of the desktops combined. With your calculations, you might even realize you don't need a SAN. You might be able to get by with local storage (as I've spoken about before).

Daniel - Lead Architect - Worldwide Consulting Solutions

Follow Me on twitter: [@djfeller](#)

My Blog: [Virtualize My Desktop](#)

Questions, then email [Ask The Architect](#)