Spinning out of control

Using flash storage in enterprise environments

Eenvoud in ICT

Author(s) : Marcel Kleine, Herco van Brug

Version: 1.0

Date: June, 2012

REFERENCES

StorageIO Blog	http://storageioblog.com/?p=3025	
Virtuall.eu	http://virtuall.eu/solutions-based/	VDI & Storage: Deep Impact

ABBREVIATIONS

Abbreviation	Meaning
SSD	Solid State Drive
NAND Flash	Not And (electronic logic gate)
RAM	Random Access Memory
VC backing	Venture Capital
HA	High Available
FC	Fibre Channel
iSCSI	Internet SCSI (Small Computer System Interface)
IOPS	Input/output Operations Per Second
PRAM	Phace-change Random Access Memory
FeRAM	Ferroelectric Random Access Memory
MRAM	Magnetoresistive Random Access Memory
De-dup ratio	Ratio of the amount of data before and after de-duplication

CONTENT

1. INTRODUCTION

Solid State Drive (SSD) is definitely one of the buzz words of the last year but it is not by any means a new technology. SSD has been with us in some form or another for quite some time. The real question is, why is it gaining momentum now?

In enterprise environments RAM based SSD's have existed for over a decade to accommodate a niche high performance market that not many of us had a lot to do with. Over the course of that decade however traditional disks have kept on growing in size but not in speed. As a result the balance between capacity and speed has grown warped and confronted us with scenarios where we have needed to put many more drives in our storage systems than we would normally need from a capacity viewpoint only to generate the speed we need. And we most certainly need speed. Consolidation of our server farms, applications and desktops on computing platforms where Moore's law still applies have pushed traditional arrays to their limits.

Does this mean we are going to be replacing our traditional magnetic drives with SSD technology? Not quite, but giving some serious thought to where SSD technology fits in a modern storage landscape is something all of us should be doing.

Over the last two years NAND based SSD's have been making an appearance in the enterprise market and they are doing well. They however have their limitations and we will be discussing those in more detail in this document.

As with any storage design, which SSD should be used and where it should be positioned all depends on your application landscape, and of course your budget. You may not require SSD technology for your entire application set.

Altogether enough reasons to have a serious look at what SSD is and what it can do for you!

1.1 OBJECTIVES

The overall goal of this whitepaper is to share information about the internals of Flash Memory as well as positioning Solid State Disks in solutions like:

- Flash cached arrays
- Flash-tiered arrays
- Hybrid File Systems
- Flash-Only arrays
- Server side flash
- Network flash

1.2 INTENDED AUDIENCE

This document is intended for Architects, System Administrators, Analysts and IT-Professionals in general who are responsible for and/or interested in designing and implementing storage solutions.

1.3 SUGGESTIONS AND IMPROVEMENTS

We did our best to be truthful, clear, complete and accurate in investigating and writing down the different solutions. Although PQR doesn't have a strategic partner relationship with all the storage vendors mentioned in this whitepaper our goal is to write an unbiased objective document where possible, which is valuable for the readers. If you have any comments, corrections, or suggestions for improvements of this document, we want to hear from you. We appreciate your feedback. Please send e-mail to either one of the authors Herco van Brug (hbr@pqr.nl) and Marcel Kleine (mkl@pqr.nl) or PQR's CTO Ruben Spruijt (rsp@pqr.nl). Include the product name and version number, and the title of the document in your message.

2. SSD INTERNALS

The reason SSD's are so hot is because they deliver a massive amount of IOPS, microsecond latency and linear scalability in throughput. Whether it's RAM based, MLC or SLC, the performance of SSD's scales far beyond what a traditional magnetic disk can generate.

2.1 DISK TYPES

Let's have a look at the characteristics of the technologies involved to make the differences a bit clearer.

Traditional magnetic disk

Speed in magnetic disks is created by spinning a disk below a read and write head, the faster the disk spins, the more sectors on the disk can be read or written. Magnetic disks simply can't go faster than they do today because the physical limits have been reached. At 15.000 rpm the edge of the platter moves at almost 4000 meters per second. If it would rotate any faster, the platter would shatter due to a combination of centrifugal forces and drag. It would in effect spin out of control.

The seek time of these 15k rpm disks are as low as 2ms. That means that their theoretical IOPS would be around 500. But with random data and different block sizes, the actual IOPS a disk can deliver is in the 160-200 range. Speed for traditional magnetic disks hasn't changed much over the last 10 years. Aerial density has though; disk sizes are getting bigger and bigger while the platter size stays the same. With the same rotation speed the sequential IO rate will go up somewhat, but the random IO rate is roughly the same as it was 10 years ago.

We can speed up magnetic disk by working around these limitations in a few ways; Serializing or Short stroking. Serializing random writes can speed up the IO rate from 160-200 to 500-600 IOPS per disk. Short stroking is the technique where only the outer part of the disk is used, it will have an impact on capacity but will keep the head away from the slower inner areas of the disk.

RAM based SSD

RAM based SSD's are what you would expect, based on RAM. It's as fast as we expect from RAM (nanoseconds latencies) and has no issues with life expectancy. But it does lose its content when you cut power to it. This means batteries, the bigger your SSD, the bigger your battery. And of course you need to make sure that you restore power to your system before the charge on said battery runs out. There are systems that use battery power to keep the contents of the RAM safe and download the data into a persistent storage device, quite often NAND based Flash.

NAND based SSD (Flash)

NAND based SSD have been around for a while now and started their career in small devices. It has however become a serious player in enterprise storage land. It's relatively cheap to manufacture and although it has some serious flaws with regards to life expectancy, it's a persistent technology (pun intended) and with proper management can be a good addition to your storage landscape. We do however need to be aware of its shortcomings for us to be able to position it appropriately. Because NAND based SSD's are the fastest growing group out there we're going to go a bit deeper into what makes this technology tick.

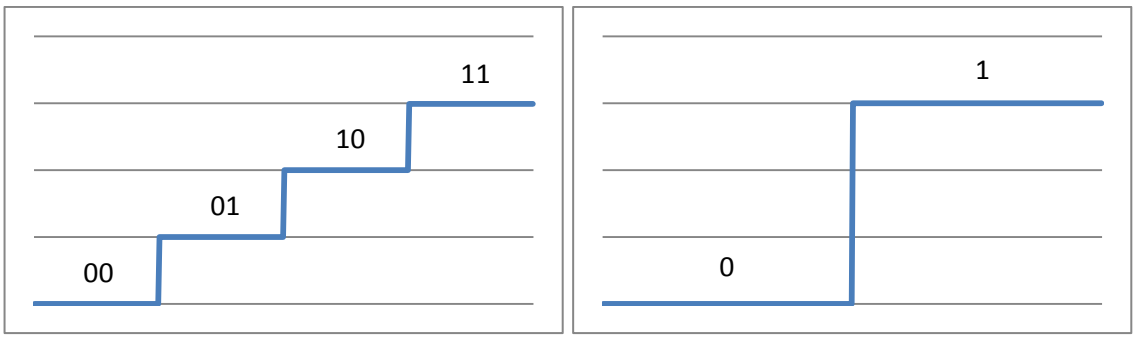
2.2 FLASH MEMORY EXPLAINED

Traditional disks and RAM based SSD technologies are generally well understood, we know the pros and cons. Flash memory, or NAND Flash as it's called officially, is not that well understood.

We know it's fast and also that it wears out after a while. Let's have a look at why this is.

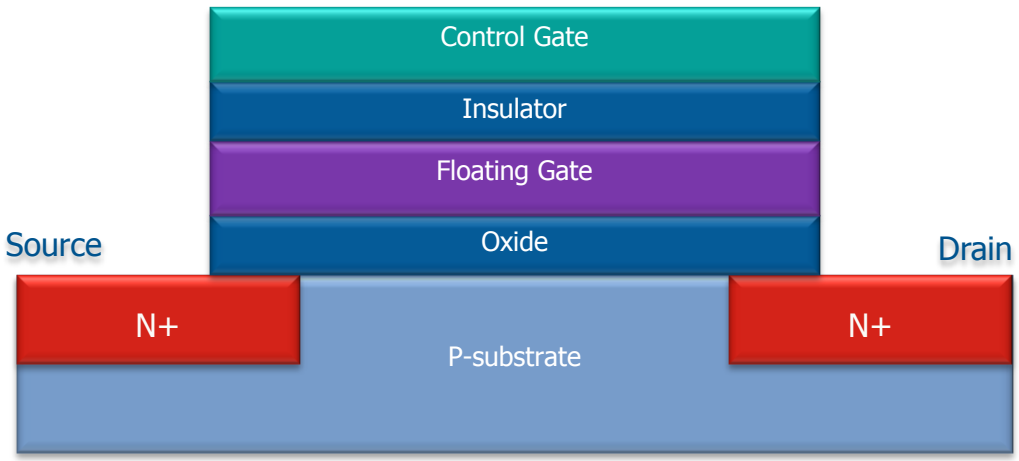
NAND Flash is built up from cells that can trap an electrical charge and thus store binary data. We divide Flash into categories based on the amount of data that can exist in a single Flash cell. Where MLC, or Multi-Level Cell can store 2 bits in one cell, SLC, or Single Level Cell, can store only one bit. We even have TLC where 3 bits can be stored in a cell.

There is a caveat though, the amount of data that that is stored in a single cell determines the amount of times the cell can be written to before it wears out. Because an MLC has less distance between levels, it is more prone to error than an SLC. (see below illustration) And this is the trade off in a nutshell. We either get more write/erase cycles, or higher density. And these numbers have an impact, where SLC can cope with as many as 100k write/erase cycles, MLC can only 5k/10k and TLC a little as 500. As a result the more resilient the type of flash we use is, the lower its density and therefor the higher its cost per GB.



2.3 Why does Flash Memory break down

To understand why Flash cells break down, we have to take a look at the transistor level of a cell. A flash cell is in essence a Field Effect Transistor (FET) with a so called 'Floating Gate' that is electrically isolated by an oxide layer. This floating gate stores electrons and represents the stored information. If there are electrons inside the floating gate, the transistor will not conduct electrons and represents a logical 0. When there is no charge in the floating gate, the transistor is conducting and represents a logical 1.



To write or clear a cell, a relative high voltage is used to force electrons to tunnel through the oxide layer to or from the floating gate. But sometimes electrons get caught in the oxide layer and contaminate the cell. This disturbs the correct readout of the cell. When this happens too much, the cell is broken. And with multiple levels in one cell, the disturbance is a problem much quicker. That's why MLC cells break so much faster than SLC cells.

2.4 Erasing

While Flash Memory can be addressed (read) per byte, it can only be erased per 'block'. These blocks consist of many rows of cells and are usually somewhere between 256 bytes and 16kB or

sometimes 64kB in size. The larger the block size, the higher the density of the Flash Memory is, simply because less space is needed for the connections to all the cells. And the higher the density, the cheaper the module gets.

When a block is erased, all cells are set to '1'. It is possible to write a single byte to memory without affecting its neighbors in the block by only changing the specific '1's to '0's. But to change a '0' back to a '1', the entire block needs to be erased first and then all the '0's need to be put back in.

This means that if you write a random byte over and over to a specific locating in memory, all the cells in the block will get a write hit, decreasing their lifespan as well as the one you intend to write. This is the **Write Amplification** that Flash Memory suffers from.

2.5 CONTROLLERS

The way to deal with the write amplification and more importantly, writes to the same cell, a controller makes sure that data written to the same block is not actually stored in that same block but stored in a (preferably) previously emptied block. This is called **Wear Leveling** and besides spreading the load over all the cells extending the life of the module considerably, it makes writing much faster because cells don't need to be cleared first

To make sure there are empty cells that can be written to, controllers perform a sort of Garbage Collection. All cells that contain non-current data are cleared in the background. That way, when a write IO comes in, there's an empty cell to write to. This is again a factor to take into account though. When writes come in faster than the garbage collection can clear cells, the controller will slow down the data intake and the Flash Memory will suddenly become much slower than it used to be. This is called the **Write Cliff** and can slow down a flash device from 300MBs/ to 16MB/s when not handled correctly.

2.6 INTERFACES

Flash Memory is packaged in several different ways. A lot of people use the term Flash Memory and SSD interchangeably but that is not correct.

SSD uses a SCSI or ATA interface to translate disk commands into Flash Memory commands. It is really from a sort of legacy or backwards compatibility with traditional (spinning) disks that these exist. Or in other words, you don't need a new connector to be able to use Flash Memory.

Interfaces that are currently used range from SATA at 3 and 6 Gbit, 4Gbit FC. These will however slowly disappear into legacy and will be replaced by SAS which is currently up to 12Gbit.

Another way to address Flash Memory is by putting it behind a PCI interface. The reason why these interfaces outperform SSDs is that the PCI interface can handle 16GB/s of data. Plenty to saturate the flash controller instead of the disk controller. This is also the interface closest to the CPU which will result in lower latencies. But the downside is that you need additional drivers to access these devices. In general it's not possible to boot from these.

Violin uses a technique where no SAS or SATA controllers are used, the Flash cells are connected to purpose built intelligent boards that are switched through the mainboard of the array.

The interface most commonly used is probably the USB connector. This makes flash easily accessible. It's just not very fast.

2.7 SPEED

Unfortunately applications and operating systems have not started to generate more sequential workloads. If anything, the use of virtualization and other consolidation techniques has increased random workloads. As a result we're adding more and more spindles when we don't

need the amount of disk capacity all these disks give us. We're wasting space that we don't need for the sake of the IOPS that we *do* need. IOPS we can generate using SSD's.

Where a traditional disk can generate up to +/- 200 random IOPS, a NAND based SSD can easily achieve several thousand IOPS, RAM based SSD's are faster still.

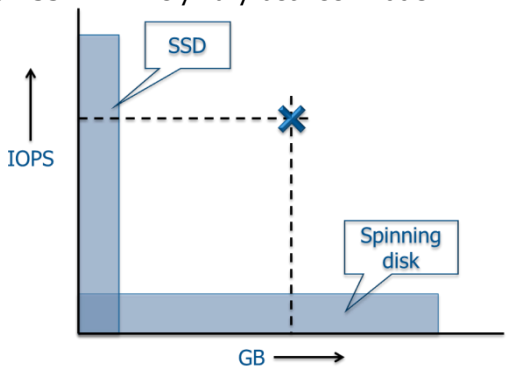
The advantage of SSD is that it contains a lot of Flash Memory modules working together in a matrix to perform at insane IO speeds although the throughput is still limited to 150MB/s or sometimes 300MB/s per drive. They will be severely limited if the controller is designed to drive spinning disks because they saturate backplanes and CPUs in no time. The IO load and throughput that SSD's can handle is enough to make traditional storage vendors limit the number of SSD drives they support in their storage arrays.

When doing a comparison between different SSD vendors be sure to compare apples with apples. What block size is being used and are the writes that are generated actually being committed?

2.8 CAPACITY

SSD drives at present are much smaller than magnetic disks. Where the latter can go up to 900GB at 15k RPM and as far as 4TB on 7.2k RPM an SSD will likely vary between 100GB (usually SLC) and 500GB (MLC) for NAND Flash. RAM based SSD's at present go up to 1TB.

That means that if you need 8TB of storage you would need about 16 regular disks in a RAID 5 solution where for SSD you'd need around 50. With an average price tag for SSD of 5 to 10 times that of an average 15k SCSI disk it's not hard to see the difference. When you need both IOPS and GBs there's nothing in the middle. It's either huge amounts of IOPS or huge amounts of storage. This is where tiering comes in.



2.9 POWER & COOLING

SSD drives, for their lack of moving parts, require less than half the amount of Watts then traditional spinning disks. Where a HDD needs about 7 to 10 Watts, an SSD will need between 2 and 3 Watts.

The lack of moving parts, in particular platters spinning at between 7.2k and 15k RPM, also mean there is hardly any cooling required compared to traditional drives when using the same variable for IOPS.

For example, a system that needs to support 40.000 IOPS will need 40.000/200 IOPS per disk = 200 disks. Compared to an SSD based solution that supports 5.000 IOPS per disk we would only need 40.000/5.000 IOPS per disk = 8 disks. That's 25 times less power, cooling and housing required, something to consider.

2.10 DATA INTEGRITY

Due to the known issues with wear and tear on NAND Flash most SSD manufacturers add multiple levels of protection in their solution. Flash Memory modules are often oversized by 10%-30% or even more. When a cell is marked as broken, the controller allocates a spare cell and marks the broken cell as bad. A counter called Media Wear-Out Indicator (MWI) in the S.M.A.R.T. stack indicates how much of the cells have been reallocated and how much 'life' is left in the device.

Vendors subsequently add (often custom) raid-levels and wear-leveling techniques to their solution to increase life expectancy even further.

3. POSITIONING SOLID STATE DISKS

Where to position SSD technology to best accommodate your storage requirements depends on a number of factors. Each of the ways SSD's are used in the Enterprise has its pros and cons. Since the applications that are supported determine which of these points are of importance to your environment we'll discuss on a high-level which applications could benefit from a certain type of SSD. With the above chapters explaining the limitations of flash cells it should be clear by now what the challenges are that SSD disks have to conquer. Let's have a look at what the current solutions have to offer and how they handle these limitations.

SSD's are currently used in mainly the following ways:

1. Flash-cached (performance acceleration of magnetic disks by extending cache)
2. Flash-tiered (performance acceleration of magnetic disks by using a tiered approach)
3. Hybrid file systems (performance acceleration of magnetic disks by using a tiered approach)
4. Flash-Only arrays (using SSD's in a purpose built all flash array)
5. Server side flash (connecting SSD directly to the PCI bus of the server)
6. Network flash (combination of flash-only arrays en server side flash)

3.1 FLASH CACHED

3.1.1 Technology overview

Most, if not all storage arrays will have some level of caching available within the system, sometimes just for read or just for write, ideally for both. As a rule this is RAM based, battery backed cache that is in size related to the size of the platform and generally only expandable on the larger systems.

Flash cached cards expand the internal RAM based cache with an extra layer of SSD.

With read caching, predictive algorithms are used to load data that is expected to be read or has been read several times in the last time period into cache to be able to serve it faster.

Write caching is generally used to buffer incoming writes and to optimize them before writing them to disk, for example by using full stripe writes, where enough data is cached to write across all disks in an underlying stripe set. In modern arrays the use of SSD technology allows vendors to expand these caches to beyond what is normally available in the system.

When caching writes the advantage of using SSDs are immediately obvious but it's only suitable for bursts. Sustained writes at a higher level than the underlying disks can handle will fill up the cache and at that point you're at the mercy of the performance of the underlying disks.

With regards to read caching an SSD cache needs to 'warm up' for a few days so the algorithm's in the array can determine which blocks are 'hot' and therefor need caching. Depending on the access pattern of the data this can increase performance from slight to significant.

3.1.2 Pros and Cons

Strong points of this technology are that it can decrease the amount of spindles required in your existing array while very little training is required for your support staff. In the case of a read cache card a disadvantage can be that if the card fails or a failover to another head occurs the cache will need a few days to warm up and reach the same efficiency levels as before.

3.1.3 Practical applications

A dataset consisting of VDI desktops which have a high de-dupe ratio will, in a de-dupe aware cache, perform very well. In a completely random access pattern to a volume with millions of small files performance may not see such a big improvement. In a scenario like that you could however still cache your file metadata to achieve a performance boost.

But if you can get a workable average out of your measurements that can be accommodated by your magnetic disks, extending the cache of your existing array for burst handling can be a

good idea. No requirements for training staff on a new platform, low impact implementation and relatively easy migration of data are all additional bonuses here.

Examples of Flash cached arrays would be NetApp with its Flash Cache cards, note that these cards only cache reads.

3.2 FLASH-TIERED

3.2.1 Technology overview

In flash-tiered arrays SSD is used as a Tier0 where a form of tiering is used to move data either down to underlying traditional disks or up from traditional disks to SSD. This can both be a dynamic or a scheduled process and it is important to note that depending on the implementation, blocks as large as 1GB and even entire volumes can be subject to relocation, regardless of the fact that the hot data might only be a few KB in size. Generally these are the more traditional arrays that are not necessarily optimized for the use of SSD's.

3.2.2 Pros and Cons

Wear-leveling can be a serious issue if the array is not optimized to write to Flash in a sequential way. Controllers may not be fully optimized for the higher throughput and IOPS rates that Flash can provide which is why quite a few startups have actually redesigned their array from the ground up to take full advantage of the capacity Flash drives can provide. Tiering algorithms can significantly increase the load on the system.

3.2.3 Practical applications

If you have a significant investment in an existing Enterprise platform, depending on your requirements, adding SSD as a tier 0 to your array could be a good idea. It would address the issue of having to manage multiple arrays and would allow for more than 100TB under a single pane of glass, which seems currently to be the limit for 'all flash arrays'. Examples of flash tiered arrays would be Hitachi's VSP and EMC's VNX platforms.

3.3 HYBRID FILE SYSTEMS

3.3.1 Technology overview

Hybrid file systems describe a technology that allows for DRAM, SSD and traditional spinning disks to be part of the same file system. Data written to the storage system is stored into battery protected DRAM or SSD after which an acknowledgement of the write is sent to the client.

The file system subsequently de-stages the data to the backend disks but is also capable of promoting data that is getting hot. It is in effect a level of tiering within the file system.

3.3.2 Pros and Cons

The level of intelligence required to apply this kind of tiering requires a file system with enough intelligence do the work transparently from the clients perspective.

3.3.3 Practical applications

File systems like WAFL and ZFS support this level of intelligence. NetApp has announced Flash Pools, the SUN/Oracle 7000 series uses ZFS in this way, as does Nexenta.

3.4 FLASH-ONLY ARRAYS

3.4.1 Technology overview

The architecture of Flash-only arrays resembles traditional storage systems in the sense that there is a central array with FC / iSCSI, etc connectivity. Flash-only arrays have some advantages over Flash-cached/tiered arrays, they are generally designed from the ground up to take full advantage of the speed that SSD drives can provide and have advanced wear-leveling and management capabilities.

3.4.2 Pros and Cons

The price/capacity ratio is a lot less favorable than that off traditional disks but through thin provisioning, (inline)de-dupe, compression, zeroing, etc, this form of SSD approaches the price per usable/effective GB which traditional Tier 1 arrays offer, only many times faster.

Only recently players in this segment have started to offer HA within the array as an option, without it, this can't be used to support business critical applications without organizing fail-over/redundancy at higher (application/OS) levels.

Network latency is still an issue here as the compute layer is divided from the storage layer by some sort of network, be it FC or Ethernet. That means for sub-millisecond response times a more localized solution is required.

Cost can also be an issue depending on what you need. If you don't need space but IOPS a system like this one will give you the best $/IOPS price, if however you do need a substantial amount of space as well, it might be cheaper to go with a few more spindles in a traditional array. The serious players in this segment are starting to sell their systems as competitive with traditional Tier1 storage on $/GB basis due to added features like compression, de-duplication, zeroing out, etc.

3.4.3 Practical applications

Customers requiring a consistently high rate of IOPS or throughput may benefit from a system in this category. It will require some extra training, a more complicated migration path and of course an implementation will need to be scheduled. A new VDI implementation would be a good example where this technology could do well.

Solutions like Whiptail use SSD drives with MLC cells but they consolidate IOs so data gets written with blocks at a time effectively eliminating the write amplification and making a single write from all the incoming random writes.

Other vendors like Violin Memory use PCI flash cards and have a dynamic net storage space depending on IO load. They allocate more of the gross space to allow garbage collection to create empty cells. The lower the IO load, the more net space you get.

Pure Storage is still in customer try-out but they have an excellent team of technicians at work and, not unimportant for a startup, some very strong VC backing.

Nimbus Data Systems, already a financially healthy company offers HA, unified access and a decent set of included software which includes replication, snapshots, etc.

Texas Memory Systems' RAMSAN product is the grandfather of modern day SSD arrays. These days not all their products are RAM based, they have MLC arrays as well.

3.5 SERVER SIDE FLASH

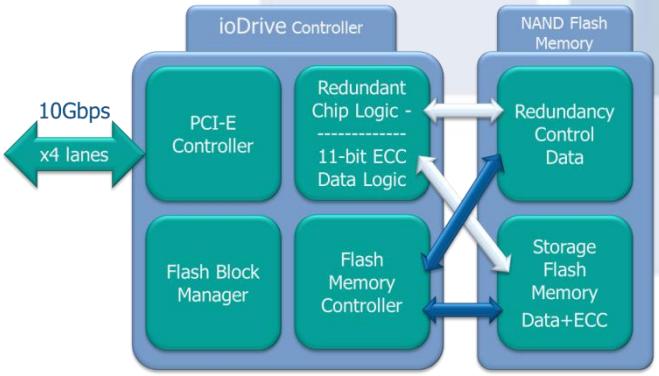
3.5.1 Technology overview

Server side Flash describes a solution where SSD's are connected to the local PCI bus of a server platform.

3.5.2 Pros and Cons

Local Flash directly on the PCI bus does not have network latencies to contend with so this is the fastest variant in the field with access times of microseconds as opposed to milliseconds. It does have some limitations; at least until flash cards can talk directly to each other over some kind of inter system PCI framework. Something some blade servers already provide! If however you need a synchronous multisite solution, network latency rears its ugly head again and you would have to go for a Hadoop-esque kind of solution to be able to use these cards.

There is a reason that from FusionIO's total turnover over 2011, 57% came from two customers, Apple and Facebook. The reason they don't see network latency as an issue is because they have solved the problem at the application level. Facebook's Hadoop systems that operate on a share nothing principle don't require shared storage functionality. Apple uses Oracles DataGuard which optimizes the network traffic between the nodes.



For those of us not using this technology this can be a serious limitation if you want to use for example, VMWare HA. DB systems that you want to mirror must be so at the application level, again ethernet latencies will increase latency considerably. However more and more Hypervisor and database manufacturers are adding this level of intelligence, MS SQL server 2012 is now being shipped with the 'Always On' functionality. At present the technology is primarily used for accelerating temporary datasets like Business Intelligence servers, non-persistent virtual desktops or temp space and log files of applications.

3.5.3 Practical applications

There are a few manufacturers that have decided to build a system where the servers hosting the PCI Flash cards can be interconnected through the use of a common enclosure. Nutanix and Kaminario use standard (blade) form factors to connect PCI busses together where each individual server has a FusionIO card with Nand based Flash and/or magnetic disk at the back of it. HP en HDS blade servers now support similar functionality where PCI busses can be interlinked, adding a FusionIO card to each server would create a system where compute and storage layers are connected via PCI.

Currently however server based Flash is intended for those datasets that have a temporary nature or that have application level redundancy combined with a demand for high IOPS and low latency.

3.6 NETWORK FLASH

3.6.1 Technology overview

Network Flash is a hybrid solution where server-side flash talks to a flash-tiered, cached or flash-only array. The key here is to notice that this technology involves network latency.

3.6.2 Pros and Cons

This approach creates a bottleneck within itself since the server-side Flash card will need to communicate with the Flash array over some kind of network.

Technologies like compression, inline de-duplication and perhaps a form of tiering that spans the internal card and the central array will alleviate this issue somewhat. With FC speeds going up to 16 and later 32Gbit/s and Ethernet going from 40 to 100Gbit/s we come a little closer to the 300+Gbits/s that PCI-E v3.0 x32 can accommodate but if you are looking at a solution at the time of writing this would be a limitation to consider.

3.6.3 Practical applications

EMC started in this Network Flash space with their Thunder and Lightning project. NetApp has also announced that it will start in this space. Currently this technology is not ready for production.

4. CONCLUSION

So what can we expect from NAND Flash Memory in the near future? Fact is that the cell degradation will keep haunting it. Density can't go down much further without this degradation causing problems. And Density is what flash needs to gain more market share.

Also, within the next couple of years Heat Assisted Magnetic Recording (HAMR) for spinning disks will become mature and we'll start to see hard disks of 100s of TB in size. IOPS might still be a problem for these disks but it will be a long time, if ever, before Flash Memory comes near that amount of GB/$.

It's more likely we'll see a mix of the two and have hybrid solutions that do *real* hot block auto tiering. Current tiering solutions where background processes move large blocks of data (100MBs in size at a time sometimes) to and from SSD in a form of cache a couple of times a day are only a viable in specific use cases. With files, a single byte change will move the whole file up to SSD. Rather pointless when it's a database file or a VDI disk image. Only inline auto tiering at block level will be able to do the job and no solutions exist today that offer that.

But there are other alternatives that may soon render Flash Memory obsolete altogether. Techniques like Phase-change Memory (PRAM) have working prototypes that are 50-100 times faster than Flash Memory. With current outlooks the data (bit-) switching times may end up only a few times slower than our current RAM[1] and make an excellent alternative to all storage solutions.

Other contestants are Ferroelectric RAM (FeRAM) and Magnetoresistive RAM (MRAM). These are comparable to DRAM in speed but are non-volatile so they don't require power to retain data. Currently they lack density and have much higher costs but in the near future one them may prove to be the one universal memory that runs and stores all data.

But until these technologies are ready for mass production and can be purchased at a reasonable price, RAM and NAND Flash are what we have to work with to close the IOPS gap created by ever growing traditional spinning disks stuck at 15000 RPM.

It is noteworthy to mention that the 5 to 7 years we expect it will take before HAMR and Persistent RAM based solutions will take over is a significantly shorter timespan than the 40+ years traditional magnetic disks have dominated as the preferred data storage media.

Is Flash therefor an intermediary solution and not worthy of investment? Not entirely. If you have a high IOPS requirement over the next 5-7 years you may very well not have a choice.

And let's not forget that although P-RAM solutions might be technically superior to NAND Flash that doesn't necessarily mean it will be adopted straight away. NAND Flash is being produced at an incredible rate and will continue to get cheaper, just as wear-leveling techniques will get better. Advances in these fields will very likely slow the adaptation of future technologies particularly since manufacturers have made some significant investments in NAND Flash production. Capital, unlike data, can be pretty slow to move.

Whatever you decide you might need, look before you leap. Measure the impact of your applications on your storage, don't stop at IOPS, block sizes and projected growth but also look at the level of availability, latency and intelligence you require at the storage level.

[1] RAM has a switch time of 2ns. That's 500.000.000 times per second. Flash Memory reads at 25µs which sounds fast but is over 10000 times slower. Writes to flash can even take up to 200-500µs when the block needs to be erased first.

5. MORE INFORMATION

5.1 ABOUT PQR

PQR is the professional ICT infrastructure specialist with a focus on availability of data, applications and work spaces with optimized user experience in a secure and manageable way. PQR provides its customers innovative ICT solutions that ensure the optimization of application availability and manageability, without processes getting complex. Simplicity in ICT, that's what PQR stands for.

PQR has traceable references and a wide range of expertise in the field, proven by many of our high partner statuses and certifications. PQR is Citrix Platinum Solution Advisor, HP GOLD Preferred Partner, Microsoft Gold Partner Virtualization, NetApp Star Partner, RES Platinum Partner, VMware Premier Partner en Gold Authorized Consultant Partner, Cisco Premier Certified Partner, CommVault CASP Value Added Reseller, Dell Enterprise Architecture Certified Partner, HDS Platinum Partner, HP Networking Master Partner, Juniper J-Partner, Veeam Gold ProPartner, Quest Software Platinum Partner and Wyse Premier Partner.

Customers of PQR can be found in all segments of society and are classified as medium to large enterprises to whom ICT provisioning is vital for running business. Sales is realized in both profit and non-profit organizations, a significant part is realized within the healthcare sector, education and local and national government.

PQR informs its clients as a Trusted Advisor about new technologies that keep ICT environments running even easier, creating secure optimal performance and information accessibility from any location or device. By using consolidation and virtualization techniques, PQR works towards an easy to use management environment. This not only applies to system administrators but also to users. PQR supports 'the new way of working' with its Dynamic Datacenter concept and cloud computing abilities. PQR implements private cloud infrastructures where availability of data, applications and workplaces in a secure and manageable way is key, and also designs and implements a variety of desktop virtualization solutions like server based computing, virtual desktop infrastructures (VDI), blade PC's and typical fat clients. In this way PQR is offering an ICT environment that increases productivity and entails significant cost decreases, not only in management but also in energy consumption.

PQR provides an ICT infrastructure that is stable, flexible and future proof. PQR has extensive experience in designing and implementing server & storage environments, including networking and security. Traditionally, massive storage environments have been PQR's specialty.

PQR's approach is based on four main pillars:

- Data & Systems Availability
- Application & Desktop Delivery
- Secure Access & Secure Networking
- Advanced IT Infrastructure & Management

The PQR approach is always transparent. To avoid common pitfalls of default configurations, only the best suitable solution will be selected, naturally in consultation with the client. During the whole process of designing up to implementation, PQR carries responsibility to deliver (part of) projects to its final result, as a rule against fixed prices and corresponding guarantees. PQR calls this Simplicity in ICT. PQR, founded in 1990, is headquartered in De Meern, The Netherlands, and counts over 100 employees. In fiscal year 20010/2011 posted sales of € 78.7 million and a net after tax profit of € 4.9 million have been recorded. www.PQR.com

5.2 CONTACT

- PQR; Tel: +31 (0)30 6629729
- E-mail: mailto:info@pqr.nl; www.pqr.com; http://www.virtuall.nl
- Twitter: http://www.twitter.com/pqrnl

5.3 About the authors

Herco van Brug

Herco van Brug was born in 1968 and studied mechanical engineering at the University of Twente in the Netherlands. Immediately after graduation he started working at Rijnhaave, later Syntegra. When Syntegra was taken over by British Telecom his position shifted to that of technical specialist, focusing mainly on specialized solutions and migrations. At present he is a Solutions Architect at PQR, with his primary focus being business continuity solutions in the datacenter.

He is the author of the VDI & Storage: Deep Impact whitepaper and co-author of the Data & System Availability diagram and is certified for Microsoft, RedHat, Citrix and VMware, while as a VMware Authorized Consultant, he undertakes VMware branded Professional Services assignments. He has been a speaker at several national conferences and published a number of articles, all related to virtualization.

Marcel Kleine

Marcel Kleine was born in 1975 and started his career as a WINTEL system administrator in the late nineties. Later he has branched out via VMWare to storage. At PQR as a Sr. storage consultant he is responsible for design and implementation of NetApp and HDS storage systems for both mid-range and enterprise customers. This involves both the implementation aspect and the link to virtualization platforms, performance troubleshooting and sizing systems based on the applications that are dependent on it. He holds certifications on Microsoft, NetApp and Hitachi technologies.

5.4 Community effort

A **BIG** thanks to everyone for their effort and support in reviewing this whitepaper.

A-Team!

Only through the effort and persistence of the team we achieved the goals, a big thanks to all!

Team Member	Job description	Email	Twitter
Ruben Spruijt	CTO	rsp@pqr.nl	@rspruijt
Marcel Kleine	Sr. Storage Consultant	mkl@pqr.nl	@marcelkleine
Herco van Brug	Solution Architect	hbr@pqr.nl	@brugh

PQR B.V.
Rijnzathe 7
3454 PV De Meern
The Netherlands

Tel: +31 (0)30 6629729
Fax: +31 (0)30 6665905
E-mail: info@pqr.nl
www.PQR.com