Dragon Slayer Consulting

Marc Staimer

W H I T E P A P E R

Shattering the Storage System Price/Performance Curve

Shattering the Storage System Price/Performance Curve

Dramatically Revising Storage System Conventional Wisdom

Marc Staimer, President & CDS of Dragon Slayer Consulting
marcstaimer@mac.com 503-579-3763

Introduction

Except for those living off the grid, in a coma for the last 20 years, or stranded on a deserted island, most storage pros are acutely aware of the ongoing extraordinary growth in data. IDC recently estimated the amount of information in the digital universe would grow 4400% between 2009 and 2020 in their report "The Digital Universe Decade – Are You Ready?" IDC says this information growth correlates into a 3000% increase in storage capacity. Hold on for a moment. The numbers don't add up. The projected information growth is approximately 47% greater than capacity growth. Curious. Much of this disparity can be attributed to the explosion of storage efficiency and data reduction technologies aimed at slowing or putting the brakes on capacity growth.

But nowhere in this IDC report, nor any other report for that matter, is there discussion or prognostication on how that headlong effort into curbing the costs of runaway storage capacity growth is causing storage performance to be squeezed beyond all recognition. It's becoming IO bound. Put simply, the IO performance is not keeping pace with capacity, in effect causing price/performance to get completely out of sync. As the cost per TB of capacity declines, the cost per IOPS of performance increases.

The Operational Meaning of the Storage System Price/Performance Problem

Storage system performance falling behind the capacity curve causes applications to experience frustratingly slower response times as well as underutilizing high-end server and networking equipment. What's worse is the application performance slowdown is variable and exceedingly difficult to predict. Troubleshooting becomes an exercise in exasperation. The Admin's Blackberry lights up like a Christmas tree with increasing frequency and urgency.

The Multiple Causes of the Storage System Performance Choke Point

There are numerous root causes to the storage system performance choke point, and all of them are the direct result of storage efficiency solutions that aim at slowing the capacity growth cost curve. They include:

- Increased storage capacity
- RAID and file system overhead
- Primary storage deduplication
- Server and desktop Virtualization

❍ Increased Storage Capacity

Hard disk drive (HDD) vendors have been increasing HDD capacity quickly within the limits of current technology as they struggle to keep up with capacity demands. As HDD capacity increases, the number of heads per GB actually decreases and invariably so does performance. The devil is in the details.

HDDs are electro-mechanical devices subject to the laws of quantum physics. They are spinning disks or platters moving at a very rapid rate ranging from 5,400 RPM (revolutions per minute) to 15,000 RPM. The HDD has a very sensitive head at the very end of a moving arm positioned above the spinning platters sliding across to read or write data. The head can only move so fast, burdened with seek time (time it takes to move the arm and head) plus rotational latency (time it takes to rotate the disk to the proper

point). Spinning disks faster can reduce latency and speed up reads and writes. Unfortunately, it appears that 15,000RPM is as fast as the industry can go without causing unacceptable errors, corruption, and power consumption. Fortunately, as HDD capacity expands, its data density becomes more tightly packed, which in turn speeds up reads and writes. In reality, the higher capacity and constant RPM are only valuable if the reads and writes are 100% sequential. Otherwise performance degrades rapidly as a direct result of the latency from head seeks. Even with sequential reads and writes, HDD performance is still not keeping up with the capacity growth. The upshot is that the higher capacity HDD is not the answer and will continue to cause declining performance per GB.

HDD vendors are wonderfully inventive and have continually tweaked their HDD controllers in an attempt to overcome that growing performance gap. The controller essentially converts the HDD from a random access device (limited by the fact it cannot access more than one part of the platter at a time) to a highly serialized one, which is noticeably faster. It does this by leveraging command queuing. Unfortunately, this technique has had only limited success in slowing the widening gap and cannot eliminate it as it continues to widen. Ultimately, the increase in HDD capacity is contributing to the storage system IO choke point.

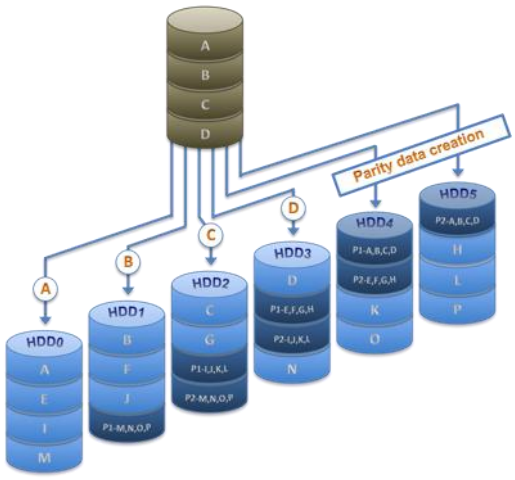
HDDs have other issues as well. Remember, HDDs are electro-mechanical devices that move at a very high speed, so they require a lot of power. The high speed generates enormous amounts of heat requiring a lot of cooling. Hot drives fail more often. That high speed also makes the HDDs quite sensitive to vibration or shock (analogous to a CD or DVD player being bumped during playing) causing read and write errors escalating HDD failure rates. Many of the reported failures in reality are not a failure. Typically 60 to 75% of the time those failed HDDs when tested by the manufacturer come back with a NTF (no trouble found) resulting in unnecessary HDD rebuilds that drag down storage system performance during the rebuild. Vibration and internal resonance are also performance thieves as they cause higher HDD latencies.

The lowest common denominator in standard storage systems today is the hard disk drive (HDD). The HDD has the highest probability of failure or lowest mean time between failures (MTBF). It is well documented that the HDD component is the Achilles heel of storage systems.

❍ RAID and File System Overhead

Conventional wisdom is that RAID mitigates the widening HDD performance gap and minimizes the impact of HDD errors and failures. RAID accelerates both throughput and IO. But those larger HDD capacities are exposing RAIDs performance limitations. A RAID group's performance is only as good as the performance of all of the HDDs within the group. As individual HDD performance per GB declines, so does the overall RAID group's performance per GB decline.

As previously discussed, HDDs fail. When a HDD fails, increased capacities has made rebuild times significantly longer. A 2TB HDD in a RAID 5 group now takes approximately 50 to 60 hours to rebuild. That's only if the rebuild is a high priority. High prioritization reduces storage system performance by as much as 50% or even more. Many organizations cannot tolerate the reduced storage system performance thereby making the rebuild priority to a background task to reduce that performance penalty to more tolerable levels. However, RAID HDD background rebuilds take up to 7 times longer. Longer rebuilds significantly increase the risk of a second drive failure or non-recoverable read error subsequently leading to lost data. This has led to increasing popularity of double parity RAID 6. This double parity does prevent data loss in the event of a second HDD failure or non-recoverable read error during a rebuild. However, the



additional parity overhead once again reduces performance and increases the time it takes to complete a rebuild. And should a 3rd HDD in the RAID group fail or suffer a non-recoverable read error, once again data is lost leading some to advocate triple parity RAID.

Conventional wisdom is wrong; RAID alone does not appear to be the answer to the price performance problem.

❍ Primary Storage Deduplication

Deduplication is a fabulous technology to slow the storage system capacity growth curve. Deduplication eliminates duplicate files, blocks, or blocklets. There are a lot of duplications in backup sets, snapshots, and replications, making it clear why dedupe provides excellent results for secondary data storage. Dedupe ratios typically range from 10:1 to as much as 500:1 depending on the type of data and timeframe. This is why most data backup software, VTLs, and backup target storage appliances currently have deduplication built-in today.

The success with secondary storage has driven considerable market interest in deduping primary storage. Lamentably, file, block, blocklet, or object deduplications have demonstrated much poorer results versus the results of deduped secondary storage. It starts with the fact that there are just a lot less instances of duplicate data. The best deduplication results of primary storage data comes from structured data (database data) and virtualized server images. This is still typically an order of magnitude lower than secondary data. There are performance issues as well. This comes from the inline architecture that is so prevalent with this type of data reduction. Smaller block sizes (ranging from 4K to 12K bytes) tend to provide better deduplication ratios. Each block write requires a hash calculation and look-up of that hash in a hash table before the write is either performed or discarded, resulting in increased IO and CPU utilization on the storage system, additional latency, reduced IOPS, and reduced throughput during writes. Read performance is also hurt by undeduping or rehydration latency. High bandwidth, caching, fast storage CPUs cannot offset the fundamental requirement for greater IO from the storage media baseline building block.

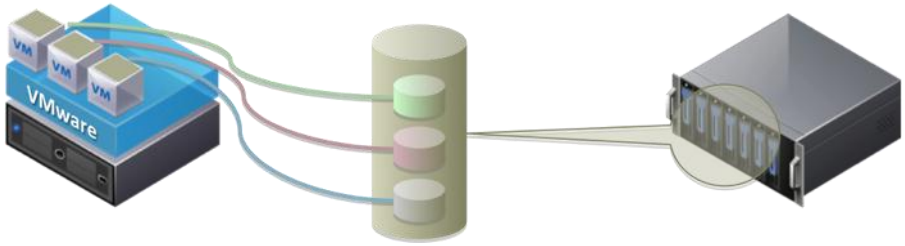
One workaround that some IT organizations attempt is to use of post-processing deduplication, which dedupes data as a batch job during periods of inactivity, rather than in real-time the way inline deduplication works. Just like with nightly backups and nightly RAID integrity verification, nightly post-processing is facing shorter and shorter windows. Disappointingly, post-processing dedupe does little better than inline when it comes to deduping primary data.

Neither inline nor post-processing dedupe do very well with unstructured data. Most unstructured data today is compressed already by the application that created it. Already compressed files do not dedupe or compress..

Therefore, while deduplication does reduce the storage system capacity growth curve somewhat, without a storage system architected to optimize both capacity and performance from the ground up, IT administrators will have do make difficult tradeoffs.

❍ Server and Desktop Virtualization

Server virtualization has exploded into the market over the last several years. The operational value proposition comes from much greater application availability as a direct result of drastically reduced scheduled and unscheduled downtime. The economic value proposition is just as compelling with the consolidation of servers, switches, cables, power, cooling, networks, etc. Server virtualization is here to stay. But since there is no such thing as a free lunch, there are several serious storage issues.



The first is too much LUN over subscription. Server virtualization virtualizes the LUNs assigned to them from their storage

systems. But the storage systems have no way of identifying the different VMs that are addressing the same LUN. If one mission-critical VM application is attempting to read or write to a LUN while another less critical VM application is currently reading or writing to that LUN, it may have to wait. Depending on the number of operations ahead of that mission critical application, it could be waiting quite awhile. The problem is exacerbated by high capacity SATA drives that have severely limited buffers or queues. If the queues fill up before an application's read/write IO's hit the system, that application can experience a storage time-out. Once again, the storage admin's phone lights up like a Christmas tree and the calls are not ones of congratulations.

The second comes from server virtualization's effect on storage IO performance. Multiple VMs coming from one physical server appear as random IO even if all of the VMs applications are sequential IO based. This is because the storage system once again cannot differentiate between the different VMs.

Virtual desktop interface (VDI) is becoming more popular with server virtualization administrators. Yet VDI creates an enormous random IOPS problem every morning when everyone logs on at approximately the same time. Most storage systems' performance slows to a crawl, causing exceptionally long boot times for the VDI users and migraine headaches for the administrators.

❍ The Negative Impacts of Declining Storage Performance

Each approach to reign in relentless storage capacity growth sacrifices performance IO and throughput to achieve their goals. This is especially onerous for random read/write IO. Random IOs per second (IOPS) is always a lot lower for storage systems than sequential IOPS. This is why most storage systems list their max IOPS performance as sequential. The rise of server virtualization, unprecedented unstructured data growth, , deduplication, and thin provisioning has caused random IOPS to become increasingly essential. Each of those technologies creates highly randomized IO. In the end, the random IOPS performance requirements are an ugly problem that's only getting worse.

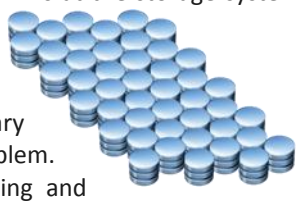
This is typically the point where storage administrators have that OMG moment, where they discover that everything they're doing in an attempt to get their storage growth under control is killing their application performance.

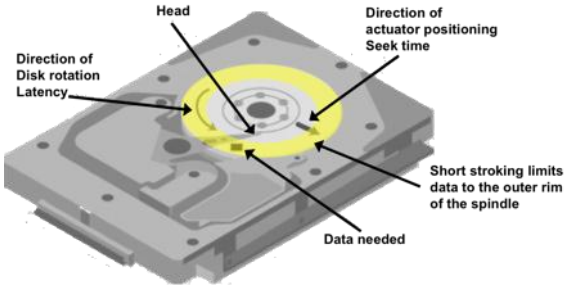
Real World Work-Arounds

Its human nature upon discovering a difficult problem is to make some attempt to figure out a solution. And there are a number of common workarounds that storage admins like to try in solving the performance problem. They include: HDD sprawl and/or HDD short-stroking; Single level per cell (one bit per cell a.k.a. SLC) solid-state flash drives (SSD) as cache; and SLC SSDs utilized as Tier 0 storage with automated tiering software. Each has its issues.

❍ HDD Sprawl and/or HDD Short-stroking

The most common workaround is HDD sprawl or the act of throwing more HDDs at the storage system. HDD sprawl does add performance incrementally; however, the law of diminishing marginal returns comes into play. Each additional HDD adds incrementally less performance until adding that next drive provides no additional performance or even a net subtraction. HDD sprawl is an illusionary solution because it does not really solve the random IOPS shortfall problem. What's worse is that it significantly increases OpEx such as power, cooling and datacenter space for what can only be described as marginal gains.





HDD short-stroking is another favorite workaround that does somewhat better at improving random IOPS than HDD sprawl while still coming up a bit…short. HDD short-stroking attempts to solve poor random IOPS by reducing HDD seek time latency that occurs each time the head moves to write or read data. Random IOPS performance is chiefly dependent upon the HDD or RAID random

seek time. HDDs and RAID random IOPS are generally not good. This can be especially onerous when the HDD head has to access random information distributed across the entire platter.

HDD short-stroking minimizes read/write head repositioning latency by software limiting the number of tracks used per HDD. Usually the limit is a fixed percentage ranging from 10 to 33% and makes only the outer sectors of the HDD available. The outer sectors provide the best performance and the head has far less movement in accessing the data. Independent testing has demonstrated HDD short-stroking adding between 40 to 50% random IOPS per drive and RAID group, which is significant.

Because there is no such thing as a free lunch and there are always tradeoffs, HDD short-stroking has drawbacks. Foremost, it means the loss of significant usable capacity. Throwing away 67 to 90% of usable capacity is an awful lot of wasted capacity. But that reduction of capacity does not reduce the amount of power and cooling required for those short-stroked HDDs. In fact, the wasted capacity leads to more HDDs to meet those pesky data growth requirements. More HDDs leads to more racks, floor space, cables, storage systems, ports, switches, tons of manually intensive tasks, plus more power, cooling and ultimately more CapEx and OpEx.

This workaround improves random IOPS at a very high direct and indirect cost and is ultimately limited by the number of drives supported by the storage system. So while HDD short stroking appears to raise costs marginally while increasing random IOPS significantly, it actually ends up raising costs significantly.

❍ SLC SSD Flash Drives as Cache

The advantage of SLC Flash SSDs is performance. SLC Flash SSDs are equally ideal for random IOPS as they are for sequential while providing equivalent or better reliability as HDDs. There is virtually zero seek time or latency. Random IOPS are multiple orders of magnitude faster than HDDs and measured in tens of thousands of random IOPS vs. hundreds for HDDs. This would appear to be the answer then to the random IOPS issue. Regrettably, it is not.

SLC Flash SSDs provide lower capacity than equivalent HDD drives as well as carrying a breathtakingly higher price tag. Current market prices place SLC Flash SSDs at between 10 to 20 times more expensive per TB than 15K RPM HDDs.

This is why many vendors utilize as few SLC Flash SSDs as possible and try to leverage them in the form of a cache. The SLC Flash SSD acts as the initial landing place (cache) for written data and the first place where applications seek their data. Data is written from the cache onto disk during idle storage system cycles and on a first in first out (FIFO) basis as the cache fills up. SLC Flash SSD cache significantly improves random IOPS performance by taking advantage of the SLC Flash random IOPS.

Of course there are limitations to this approach beyond the high cost. Limiting the amount of cache is designed to reduce the sticker shock, but has the effect of reducing read cache hits. Without a read cache hit, IO is once again redirected to the underlying HDDs, adding additional latency to the already poor random IOPS performance of hard drives. Most storage system architects recognize that to get cache hits of 50% or higher requires the cache to be at least 10% of the total storage. This means for every 100TB of usable storage there should be at least 10TB of cache, otherwise cache hit probability declines exponentially. As cache hits decline, so does the value of that cache. Few storage vendors recommend that much cache because of the cost; in fact, if SLC Flash SSDs are 10x the cost of 15K RPM drives, that prior example of adding 10TB of cache to a 100TB system will double the purchase price of the system without any gain in usable storage capacity.

When that cache is utilized as write-back cache, it only speeds up the write performance random IOPS, not the read performance random IOPS. When part of the cache is locked down for specific database applications (for indexes, metadata, and/or hot files), it reduces the amount available for the rest of the applications. Managing cache lock-down is a manual intensive process and quite time consuming as well. Admins tend to add more SLC Flash SSDs in an attempt to reduce their complexity which also raises costs and price/random IOPS.

SLC Flash SSDs as cache is a complex solution that adds significant cost and far too much complexity.

❍ SLC Flash SSD Drives as Tier 0 Storage with Automated Tiering Software

The concept behind this work-around is to make SLC Flash SSDs as the primary Storage Tier or Tier 0. This Storage Tier is for the most recent data making the assumption that most recent is also the hottest data and most likely to be accessed. It is also the target for data from applications requiring a high random IOPS performance. As Tier 0 data ages out and access is far less frequent, the concept is to move the data to a lower performing lower cost Storage Tier. All other data is also relegated to other lower performance Storage Tiers (Tier 1 with 15,000-RPM HDDs, Tier 2 with 10,000-RPM HDDs, Tier 3 with 7,200-RPM HDDs).

Cost savings comes primarily by leveraging the SLC Flash SSD's high IO performance to eliminate or reduce the requirements for lower capacity high RPM HDDs. Utilizing fewer HDDs with higher capacity (albeit lower performance) fulfills the capacity needs. The combination of SLC Flash SSDs drives with fewer high capacity HDDs reduces racks, floor space, with the key savings coming from reduced power and cooling.

The problem with this work-around is that it requires either manually intensive data migration to move data between tiers, or expensive automated Storage Tiering software that automatically migrates data between Storage Tiers based on policy. Automated storage tiering software tends to move data only in a downward direction. It is not designed to work with real-time dynamic workloads or IO storms. Neither the granularity nor flexibility is there at this time. This means the tiering software must be spot-on accurate nearly 100% of the time. Realistically, this is highly unlikely. This requires the most common storage workaround; buy more storage in each tier, which defeats a large portion of the economic value proposition.

Then there is the issue of ongoing automated data movement which puts data is in motion quite a bit. This frequent motion causes a whole new set of storage administrator issues.

- Performance can be seriously compromised if data being requested is in a state of movement.
- Power consumption is increased dramatically because automated tiering keeps HDDs spinning in an "active" state even if an application's data IO pattern is ideal for a storage tier adjustment.
- Automated tiering tends to create uncertainty and uneasiness about not knowing exactly where the data is at a given time.

More importantly though is that it is the mission critical or primary applications that require the best performance, and in the end, such applications require tier-0 or tier-1 storage, minimizing the ability of automated tiering to provide much in the way of benefit. Only data that has aged out of active use will get tiered. To be fair, even with these shortcomings, automated storage tiering software has made this work around somewhat more palatable. Be weary of vendor lock-in, however. Due to support issues, tiering solutions from storage vendors only support those vendors' products, locking in customers and giving the vendor the upper hand. In addition to the issues it causes, tiering solutions themselves are not inexpensive, and the licensing cost eats up so much of the savings derived from the mix of SLC Flash SSDs and high capacity high density HDDs that it makes this only a partial solution at best.

There has to be a better way and fortunately, there is.

The Better Solution to Storage O Price/Performance Problems – Nimbus Sustainable Storage

Nimbus Data Systems is completely changing the storage price/performance paradigm by essentially turning it on its ear. Nimbus is delivering storage systems (aptly called Sustainable Storage) that are 100% flash- based with zero spinning HDDs.

The Nimbus S-Class and E-Class are Enterprise Storage Systems using flash exclusively costing approximately the same at mid-tier storage systems using SAS and no SSDs. Plus, that equivalent cost impressively provides more than 10 times the IOPS and throughput performance. The Nimbus delivers up to 800,000 4K IOPS per shelf. It scales today from as small as 2.5TB up to 500TB in a single system with expectations of even higher scalability down the road with its 4 exabytes file system. Based on current pricing, that's $.03 per IOPS, which is 95% less expensive (or even more than that) than standard HDD-based mid tier systems. More impressive are the power numbers of 1,600 IOPS per watt and as low as 5W per TB of stored data, 80% less than 15K disk.

The question becomes how does Nimbus do this? The answer is two-fold: vertical integration and comprehensive, in-house developed storage software. Rather than rebadging someone else's SSD, Nimbus has designed its own modules called "flash blades" based on EMLC flash NAND silicon. The flash blades are 100, 200, 400GB, and 800GB in usable capacity, and Nimbus system packs 24 blades in one 2U rack mount shelf. 12 Westmere/Nehalem cores with up to 4 hardware offload engines provide ample processing power.

Of course, MLC Flash chips typically do not have the wear characteristics of SLC Flash chips. One of the things that allow SLC chips to be classified as enterprise grade is their ability to supply 100,000 erase cycles before failure. MLC Flash chips are considered consumer grade because they only deliver 1/33 or 3,000 erase cycles.

Once again, Nimbus turns that on its head. By utilizing extended life MLC Flash chips (a.k.a. Enterprise MLC or EMLC) that produce 30,000 erase cycles or 10 times as much as standard MLC, Nimbus delivers enterprise-grade reliability with capacity density that is superior to HDDs. For those concerned about the wear life of these EMLC, some basic math reveals they need not worry. To exceed the erasure cycle life of a 10TB shelf would require the actual writing of more than 164TB every single day on that shelf for five consecutive years. That translates into 7TB every single hour over those five years. In other words, it is not going to happen.

Nimbus goes even further in boosting EMLC Flash endurance by over-provisioning with 28% extra capacity reserved as a backup. This over provisioning provides some headroom within the controllers for the EMLC wear-leveling algorithms and garbage collection so that system performance is consistent regardless of how much data is on the Nimbus system.

To be truly enterprise-class requires sophisticated storage software. All Nimbus systems ship with Nimbus' HALO OS included at no additional cost. It includes all Enterprise Storage software such as:

- Intuitive Web interface
  - The expertise is built into the software not the user
  - Plus a remote CLI
- Unified Storage protocols
  - iSCSI
  - NFS v2, v3, and v4
  - CIFS with Microsoft Active Directory
  - Fibre Channel
  - Infiniband with SRP (SCSI RDMA Protocol)
  - FCoE for converged enhanced Ethernet networking
  - 4 Exabyte file system
- Inline deduplication to curb the storage capacity growth curve
  - With up to 10:1 reduction
- Storage virtualization and simple provisioning
- Thin-provisioning
- Performance-optimized schedulable snapshots
- AES-512 encryption (FIPS-compliant)
- Mirroring
  - Synchronous for high availability
  - Asynchronous continuous replication for disaster recovery
- RAID dual-parity protection and rapid data rebuilds
  - RAID 6
  - Hot-swap modules with 20-40 minute rebuild time
- Email/SNMP notification and reporting

- Benchmarking tools and proactive monitoring
- Multipathing and load-balancing

❍ The Nimbus Answer

As discussed above, applications such as virtualization and primary storage deduplication demand storage with uncompromising IO capabilities. No amount of HDD spindles, cache in front of HDDs, or tiering between HDDs and SSDs can compete with the IO potential of a purely flash-based storage system. Nimbus is the first company to have made such a solution both economically viable, with the performance, capacity, reliability, and advanced functionality required for enterprise production deployment. This means that those difficult problems and challenges discussed earlier are effortlessly conquered. Meeting virtualization performance while combatting storage capacity growth with primary storage inline deduplication is doable because of the high-performance IO and CPUs within the storage system. It also means that even irksome RAID rebuilds are no longer a performance or data loss problem because Nimbus flash blades typically rebuild in less than 30 minutes compared to hours or days with traditional HDD arrays. And it accomplishes all of this with less power, fewer media failures, less cooling, and less datacenter space. Fundamentally, that means better storage performance at a fraction of the operating cost.

Conclusion

The Nimbus Enterprise Flash Memory Storage is a true paradigm shift in storage systems that shatter the IO Price Performance Curve. There are numerous exasperating problems from the explosive storage capacity growth. Problems that include declining application performance, increased infrastructure complexity, plus greater power and cooling consumption. Many of the solutions implemented to slow the explosive storage capacity growth curve cause application performance to decline. And the workarounds create other problems and complexity. It is similar to squeezing a balloon. As pressure is applied on a part of the balloon, it bulges out in another part.

The answer is to fundamentally rethink the root cause of the problems and solve them rather than just treating the symptoms. The Nimbus is the first of its kind that does just that.

For more detailed information, please contact:
info@nimbusdata.com or go to www.nimbusdata.com

About the author: Marc Staimer is the founder, senior analyst, and CDS of Dragon Slayer Consulting in Beaverton, OR. The consulting practice of 13 years has focused in the areas of strategic planning, product development, and market development. With over 31 years of marketing, sales and business experience in infrastructure, storage, server, software, and virtualization, he's considered one of the industry's leading experts. Marc can be reached at *marcstaimer@mac.com.*