PAVILION

WHITE PAPER

Storage Performance Density

Emerging Requirements for Modern Applications

Table of Contents

Introduction

IT organizations have been dealing with data's relentless and unabated expansion for many years. Storing, managing, protecting, analyzing, and rapidly reacting to data growth is an ever-evolving adaptive process. Dealing with data growth is a never-ending battle and it's only getting worse. IoT, compliance, security, online gaming, social media, high frequency trading, AdTech, eCommerce, are just some examples of workloads generating huge amounts of data requiring real-time processing. Modern applications were designed to process swelling amounts of data in real-time.

But as data continues to scale, performance has not kept up. Providing the required performance has led to complicated mechanisms adding significant costs and diminishing marginal performance returns. It has become increasingly clear that the issue is the lack of storage performance density. This is not to be confused with storage capacity density or drive capacity density.

This paper examines why these applications suffer from poor storage performance density, why the common storage scale-out server centric architecture of direct attached storage (DAS) a.k.a. software defined storage (SDS); shared all-flash-arrays (AFA); and even shared accelerated storage; fail to provide the necessary storage performance density, and how that deficiency causes a cascade of multiple urgent problems. It then details how the Pavilion Hyperparallel Flash Array (HFA) uniquely solves the modern application storage performance density problem without the forced compromises of alternative approaches.

Understanding Storage Performance Density

Storage vendors have been consistently rewarded by the market for storage capacity density. Storage capacity density is the amount of capacity capable of being crammed into a shelf or rack unit (RU). It's why drive vendors increase drive capacities for every form factor on a constant basis and ODMs are so clever in stuffing as many drives as possible into a chassis. IT's runaway data growth continually drives the more capacity in a smaller space paradigm. However, increasing capacity per RU does little for the performance side of the equation. In fact, the opposite is true. Performance hard disk drives are all but obsolete because every year they got bigger, but they never got faster. Performance per terabyte actually decreased every year. The same trend is now occurring with SSD's: a 1 TB SSD has the same performance as a 15 TB SSD. In other words, the 15 TB SSD is 15 times slower than the 1 TB SSD. Storage capacity density increase per RU erodes storage performance. Storage performance density per RU is now more important than ever.

Why Storage Performance Density Matters

Data center real estate is costly and precious. Each floor tile has power, cooling, conduit, and more running to it. There are hard limits on how much data center real estate, power, cooling, UPS, conduit, access, etc. is available. That real estate is measured and allocated in rack units (RUs). RU consumption also has a high operating cost (OpEx). Each RU consumed requires a greater percentage of fixed data center allocation and personnel costs besides the storage plus the supporting infrastructure equipment costs themselves.

Higher storage performance density means fewer application server nodes; reduced network switch ports and switches; reduced cables, transceivers, less RU consumption for applications. It also provides additional flexible elasticity for more applications and processing. Limited storage performance density generally results in reduced and inadequate performance, which has a markedly harmful effect on revenues, time-to-market, worker productivity and effectiveness, business reputation, growth, competitiveness, employee satisfaction, turnover, and cost. Poor storage performance density has outsized cost to the organization.

Technology Factors Affecting Storage Performance Density

There are 5 primary technology factors affecting storage performance density:

1. Drive technology type — Flash SSDs always have higher performance density than HDDs:
    - Flash SSDs are approximately 3 orders of magnitude (1000x) more performant that HDDs
2. Underlying SSD NAND technology
    - Bits per cell
    - 2D Planar or 3D

3. SSD protocol
    - SATA, SAS, or NVMe
4. Server to storage interconnect protocol:
    - SCSI or NVMe-oF
5. Storage architecture:
    - Drives per RU
    - Bandwidth per RU
    - Controllers per RU
    - Underlying SSD Technology

Not all SSDs are the same. There are tradeoffs between IOPS, throughput, and latency with capacity, wearlife, and cost. A common way SSD storage density capacity is increased is by adding more bits per cell in the NAND chips. This methodology usually decreases storage performance density per RU. Each additional bit adds capacity density at the cost of much higher latency, reduced IOPS, reduced throughput, and much shorter wear-life.

	IOPS	Throughput	Latency	Capacity	~Wear-life	~Cost	Notes
SLC 1 bit/cell	Highest	Highest	Lowest	Low	100k writes	$$$$	Rarely utilized anymore due to high cost and low capacity
MLC 2 bits/cell	IOPS High	High	Low	Medium	10k writes	$$$	Popular because of performance, capacity, cost balance made more by 3D layering
TLC 3 bits/cell	Medium	Medium	High	High	1k writes	$$	Rising popularity because of much greater capacities, lower costs, 3D layering, and use of over-provisioning to compensate for performance and wear-life
QLC 4 bits/cell	Low	Low	Highest	Highest	100 writes	$	Niche product primarily used for cold or immutable data (WORM)

Key SSD Characteristics Tradeoffs

Another way SSD capacity density is increasing is by way of 3D layering of the cells. The additional layers add capacity and in addition some latency even as total IOPS and throughput per drive goes up. 3D layering is being utilized with MLC, TLC, and QLC NAND, and is most popular in TLC and QLC NAND.

These technological advances result is much greater storage capacity density per drive, but with storage performance density nominally greater, flat, or actually decreasing per RU depending on the underlying NAND technology.

Measuring the performance density per TB — the drive density metric — demonstrates that it is continually decreasing for SSDs. Drive capacity is increasing faster than drive performance. That doesn't translate into a reduced storage performance density per RU. What it means is that the storage performance density per RU is unlikely to increase in any substantial way.

SSD Protocol

It doesn't matter how much storage drive performance there is if it's bottlenecked by the drive protocol. SATA bandwidth tops out at approximately 500MB/s and SAS at about 1.5GB/s. NVMe is in a different tier. It tops out at approximately 1.5GB/s per lane. Running on PCIe gen 3 and 4 lanes means it's about 6GB/s, on 8 lanes 12GB/s. To get the highest SSD performance density requires NVMe.

Server-to-Storage (Outside The Server) Interconnect Protocol

It doesn't matter how much storage drive performance there is if it's bottlenecked by the drive interconnect protocol outside the server. Again, SATA bandwidth tops out at approximately 500MB/s and SAS at about 3 times that much 1.5GB/s. NVMe is in a different tier. It tops out at approximately 1 GB/s per lane for PCIe gen 3. A 2 lane NVMe card tops out at approximately 2GB/s, 4 lanes at 4GB/s, 8 lanes at 8GB/s, 16 lanes at 16GB/s, and so on. Achieving the highest SSD performance density outside of the server requires NVMe and a lot of lanes.

Storage Architecture

The 3 crucial aspects of architecting the highest storage performance density are the:

1. Number of NVMe SSDs

2. Number of controllers

3. Use of a low latency NVMe-oF bandwidth end-to-end

Conventional wisdom says cram as many NVMe SSDs as possible into a dual controller (active-active) chassis with a lot of NVMe-oF ports. The problem with conventional wisdom is the dual controllers. They become the performance bottleneck, severely choking performance because each controller only has a few PCIe channels, which limits both the number of NVMe drives and NVMe-oF IO network ports. This means to scale storage performance density requires more controllers in order to increase PCIe channels (lanes).

Understanding Storage Performance Density

Defining Modern Applications

Modern applications fall into two major categories. The first are transactional applications demanding large numbers of IOPS or IOs per second. These applications characteristically are structured utilizing any of several SQL databases, NoSQL databases, clustered databases, distributed databases, or combinations thereof. The second are high throughput applications typically associated with high performance computing (HPC) a.k.a. as supercomputing and parallel file systems such as GPFS (Spectrum Scale), Lustre, and Panasas. These parallel file systems are used by modern applications to process large amounts of data in parallel.

Cognitive computing including artificial intelligence (AI) such as AI for IT operations (AIOps), machine learning (ML) such as Enterprise search, and autonomous systems (AS) such as autonomous data management; ecommerce; and high frequency trading, a.k.a. algorithmic trading are examples of the transactional modern application utilizing parallel computing. Cybersecurity; online gaming; EDA; fraud detection; behavioral anomalies; compliance; protein analysis; oil and gas; media and entertainment, real-time analytics; real-time IoT management, orchestration, and insights are examples of modern high throughput applications. Many modern applications such as Splunk fit both categories.

The common denominator for modern applications is the mandate for consistent extremely high-performance storage. The 3 most frequent ways modern applications meet that consistent extremely high-performance storage demand are:

1. Scale-out storage utilizing:

    - Direct attached storage (DAS) to each server node;
    - Software defined storage (SDS) to share the DAS in each server node in system.

2. All Flash Arrays utilizing:

    - SATA, SAS, or NVMe SSDs.

3. Rack Scale Designs utilizing:

    - NVMe SSDs
    - NVMe-oF/RDMA : Ethernet or InfiniBand fabrics;
    - Or NVMe-oF/FC: FC (Fibre Channel).

All 3 of these storage technologies have different storage performance densities. And all 3 have different major shortcomings that cause consequential problems.

Scale-out Storage

This type of storage is the primary default storage implementation for most modern applications. It's based on commodity off-the-shelf servers (COTS) a.k.a. white box servers. Each server node converges the modern application (such as a NoSQL database) with Software Defined Storage to share all server nodes' embedded SSDs and/or HDDs with the other server nodes in the system, plus software defined networking (SDN) leveraging the server node's networking ports. This server centric storage architecture definitively constrains storage performance density.

Poor Storage Performance Density

Nodes share their computational power between the modern application, SDN software, and/the SDS software stack. The SDS storage software services functions are CPU intensive, stealing cycles from the modern application reducing its performance. Storage performance density is weak and gets continually worse as it scales. That doesn't mean that performance is not good when localizing data in a specific modern application server node on its high performance NVMe SSDs. It is in fact excellent. The problem is that data and storage are siloed. Any other server node accessing that data or node will have nowhere near the same performance. Localization also complicates scalability, data management, and data protection.

Performance suffers because each server node and associated SDS adds latency. Reading or writing data to non-localized SSDs means traversing the system network, e.g. more latency and more network traffic. The modern application and data storage must be localized together in the same node to limit unwanted latency and longer response times. If the data can't be localized because the application's distribution across all the nodes allows access from any or all of them, latency becomes frustratingly variable, as do response times.

Yet the conventional wisdom is that this storage architecture is the best bet for modern applications because of a perceived lower cost. That conventional wisdom perception is grossly incorrect. The server centric storage approach frequently costs much more than expected. Costs including increased data center consumption from exceptionally poor storage performance density and wasteful capacity consumption from server node data protection strategies; additional supporting infrastructure such as switches, cables, transceivers, conduit, and UPS; higher operating expenses from more system maintenance, software licensing, power, cooling, personnel, and personnel training.

Several noteworthy modern application problems are caused by this architecture. The first is coarsely granular scaling. To add performance requires adding a server node with additional SDN, SDS, network ports, SSDs and their capacities. But when the system just needs additional compute, network bandwidth, or storage capacity, it is simply out of luck. The scaling of the

network or storage aspects are restricted by each server node. Compute, networking, and storage are generally not separately scalable. That coarse granularity makes operations such as adding a new node somewhat simpler. However, the costs are high in wasted bandwidth, higher latencies, more supporting infrastructure, less flexibility, and a lot more unnecessary CapEx and OpEx.

That disappointing storage performance density drives several other problems:

SKU Sprawl

SKU sprawl a.k.a. server node sprawl, exists in modern applications when procurement is done on an application basis. Each application decides which server type to use, memory, network ports, SSD type, and capacity. The result is each modern application ends up with its own unique server node SKU. Lots of SKUs cause too many combinations of supported hardware, becoming operationally overwhelming. Separate maintenance agreements, different vendors, unique sparing, inventory, management, training, operations, etc. CapEx is higher from reduced economies of scale. No one likes or wants SKU sprawl.

Inefficient Storage Utilization

Server centric architecture silos data because the storage is packaged or converged with the application on the server equipment and the storage is not shareable outside the system cluster image. The application silos are compounded by the natural human tendency to err on over- purchasing at procurement time. Unused or underutilized storage capacity in every modern application cluster is simply wasted.

Maintaining data access in a server centric architecture when a server node fails or has an outage, requires all data on each node be copied to a different server node. It's called multi- copy mirroring. The number of copies required is tied to the number of concurrent server node failures data access will be protected against. Two concurrent server node failures require the data be copied at least twice. That equates into 300% storage capacity consumed compared to a typical RAID 6 of 125% storage capacity consumed. No matter how it's sliced, this is poor capacity utilization.

Excessive Data Movement

Poor storage performance density leaves few cycles for data protection. The software defined storage stack in the cluster steals CPU cycles and memory from the modern application forcing the backups off-host. Daily backups have a large recovery point objective (RPO) of 24 hrs. RPO is the timeframe between backups and is the amount of data that can be lost. Off-host means secondary storage, which will not be as performant as the server centric storage and demands more shelf space, rack space, switches, cables, transceivers, conduit, power, cooling, management, operations, maintenance, sparing, and cost.

Data is also replicated off-host to enable dev-ops and test-dev. Again, secondary storage is the target for that replicated data. In short, data protection requirements cause variability in modern application performance via CPU contention while driving massive increases in network traffic through excessive data movement.

Complicated Performance Management and Tuning

One way some server centric storage attempts to deal with the latency problem is via DRAM caching. DRAM has latencies much lower than SSDs with significantly faster response times. But DRAM caching requires cache coherency across all nodes. Cache coherency is keeping the cache in sync across all server nodes. Cache coherency becomes exponentially more difficult as the number of nodes grows thereby constraining scalability. DRAM volatility requires battery backup or super capacitors and onboard NAND flash. That form of DRAM is called a NVDIMM or Non-volatile DRAM. NVDIMMs are considerably more expensive than standard DRAM DIMMs. DRAM caching is capacity constrained per node, generally ranging from 128GB up to 3TB. Those constraints make cache misses progressively more frequent as nodes are added, causing unacceptable variable response times.

Another drastic way to improve performance is by eliminating all storage software services such as RAID, data protection, snapshots, thin provisioning, etc. because they consume too many resources. Neither are good options.

Inability to Meet Compliance and Security Requirements

Non-compliance is no longer an option. It has become increasingly costly in reputation and revenue. New regulations such as the European Union's 'General Data Protection Regulation' (GDPR) and the '500 New York Cyber Rules and Regulations Part 23' (NYCRR) have big teeth for non-compliance.

4. GDPR applies to any business that collects EU residents' personal data.

    - Minimum fine: €10,000,000 or 2% WW revenues whichever is more.
    - Max fine: €20,000,000 or 4% WW revenues whichever is more.

5. NYCRR applies to any financial service organization with an office in NY.

    - Up to several $Billions in fines.

Data audit analysis additionally has a tendency to drastically reduce modern application performance. These applications slow to a crawl when the audit scans are occurring. CPU and memory become starved. This workaround is to snapshot the data, replicate it off-server to another system, mount it on another host and then do the scans. It's time consuming and costly CapEx and OpEx while slowing down compliance.

By attempting to achieve mandatory regulatory compliance through additional software and hardware products, security exposure escalates rapidly from the greater numbers of external attack points. More servers and systems mean more operating systems that must be kept patched at all times to correct discovered vulnerabilities. The same issue for drivers, application software, analytics software, even backup software and its agents. More attack points increase the probability of a breach or data loss.

To summarize, scale-out server centric architectures appear not to be the best choice for modern applications. They have poor storage performance density, SKU sprawl, poor storage utilization, excessive data movement, complicated and costly performance management and tuning, and an inability to meet compliance or security requirements, and a much higher total cost of ownership (TCO) than anticipated.

AFA — All Flash Array

The AFA was the industry's first attempt at increasing storage performance density. AFAs are generally designed for good storage performance, good storage capacity density, and full featured storage software services. They solve much of the server-centric storage architecture runaway SKUs problems, waste problems, and excessive data movement problems. But AFAs are not designed for storage performance density requirements of Modern Applications. They have ineffective storage utilization, silo data, difficulty with compliance/security, and high costs relative to the performance they provide.

AFAs cram a lot of high capacity 3D TLC and or MLC SSDs into its system for high storage capacity density. Many now utilize NVMe SSDs to improve storage performance. Some have now added NVMe-oF or NVMe/FC to further improve storage performance. Yet most of these AFAs are limited by dual controllers. Each x86 based controller has at most 40 PCIe gen 3 lanes with 16 ingress and 24 egress. Given each NVMe SSD requires 4 PCIe lanes to optimize performance, each x86-based controller is maxed by 6 NVMe drives. That makes the controller a massive performance bottleneck. Dual controllers are going to severely constrain the system network bandwidth, number of NVMe drives, and amount of performance that can be derived from each drive. Performance increases require more controllers.

But AFAs are controller limited. That results in separate systems for the vast majority of AFAs. Each AFA is its own silo isolating data from the other systems. Once again, more systems result in more security real estate and more avenues of attacks and breaches, making compliance and security riskier. Notwithstanding AFAs built-in RAID 5, 6, 50, 60, and virtual snapshots, replication is a hard requirement between all of the AFAs if data access is to be guaranteed if an AFA failure occurs. That replication consumes at least 2x the capacity. A few AFAs can combine these separate systems into a cluster that simplifies management, security, and compliance but does nothing to change the performance density problem.

Rack Scale Flash a.k.a. Shared Accelerated Storage

Rack Scale Flash is architected for extreme performance with some improvements to storage capacity density, but few if any storage software services. Storage software services such as RAID, snapshots, and thin provisioning are CPU and memory intensive and reduce performance. Rack Flash storage leaves them out to optimize for performance.

Like AFAs, shared accelerated storage addresses SKU sprawl by disaggregating storage from the server. But early Shared Accelerated Storage solutions come up appreciably short in the areas of storage performance density, storage utilization, excessive data movement, storage performance management, storage performance tuning, compliance, and security.

Storage performance density is marginally better than the latest AFAs. Early Shared Accelerated Storage vendors overcome the active-active controller performance constraints with proprietary architectures. There are some clever – albeit still non-compelling storage performance density – high-performance architectures in rack scale flash.

How Pavilion HFA Solves These Problems

Pavilion took a hard look at the modern application market requirements and realized that storage needed a fundamental rethink. Modern applications demand the extreme storage performance of server node locally embedded NVMe SSDs; however, IT organizations do not want to pay through the nose for that performance. They abhor the manual labor-intensive management of performance management and tuning. They love simplicity and hate complexity. They don't want to add disruptive server node agent software that has to be licensed, maintained, patched, and upgraded ongoing. They detest scheduling downtime to open up their servers to replace COTS storage adapters with expensive proprietary ones.

They require their server node hardware be COTS for cost purposes. They don't enjoy or appreciate wasting their storage or IT resources. They need to protect their data from both hardware failures or outages in addition to malware, maliciousness, human error, and software corruption without expensive software and hardware. They would like to create real-time virtual copies of their data for test-dev, dev-ops, and analytics.

Pavilion recognized these market requirements and architected a completely different type of NVMe-oF storage array. Pavilion's objective was to deliver the best of each of the modern application storage options: provide the extreme performance of embedded NVMe SSDs with localized data in a server centric architecture; the shareability and storage services of AFAs; the extreme latencies, throughput, IOPS, and NVMe end-to-end of rack flash, and do so without requiring proprietary modern application server node hardware or software. But what was delivered is truly groundbreaking in terms of storage performance density and storage capacity density.

The Pavilion HFA, the industry's first HFA, can deliver in just 4 RUs, up to 20 storage controllers in active-active pairs, up to 3 Tb/s of bandwidth with up to 40 (100Gbps) NVMe-oF Ethernet/InfiniBand standard NICs, up to 72 U.2 2.5" NVMe SSDs, over 1.1PB of raw capacity, up to 4 hot swap power supplies, dual redundant management modules, thin provisioning, distributed RAID 6 data protection, redirect on write (ROW) snapshots, at a latency of 40µs, with up to 20 million IOPS, and up to 90GBps throughput. All of that again in only 4 RU, not 80 RU, not multiple physical systems, but one system.

The Pavilion HFA's performance density is as much as 5 million IOPS/RU, and 22.5 GBps/RU with a matching storage raw capacity density over 250 TB/RU. Compare that with AFA alternatives. There is no comparison.

Pavilion solves the modern application storage performance density problem. It's high bandwidth, IOPS and throughput enables it to share all of its performance and capacity with all of the application's server nodes. The exceptionally high level of scalability eliminates complicated scaling problems of server centric storage. The extraordinary performance eliminates any need to tune or manage performance.

The performance is so close to that of embedded SSDs, the application server nodes no longer need embedded drives at all. They can boot off the Pavilion HFA. That alone creates a new paradigm empowering the modern application server nodes to be a much smaller form factor such as a 1 RU server, half RU micro server, or blade server at much lower total costs. It also improves modern application availability. When a server node fails IT has only to swap out the server node, boot it off of the Pavilion HFA, point it at its volumes and that's it. Simple. The server cost reduction in hardware, maintenance, RUs, power, cooling, and other components can pay for the Pavilion HFA. And SKU sprawl goes away completely.

The Pavilion HFA solves each of the other problems as well. Thin provisioning makes sure no capacity is orphaned. RAID 6 ensures the data is protected from up to two concurrent drive failures while consuming

no more that 12% more capacity. The ROW snapshots protect the data from malware, maliciousness, human error, and software corruption without affecting application performance or having to move the data off host. This makes the Pavilion HFA platform utilization highly effective and efficient. The ROW snapshots also allow real-time virtual copies of live data to be utilized for dev-ops, test-dev, and analytics without ever having to move the data off host.

When the RAID and ROW snapshots are combined with the massive reduction of hardware and software parts that reduces the number of security attack vectors, making compliance and security that much simpler. Pavilion doesn't make the analytics, intrusion detection, breach detection, or compliance software. It just makes them all much faster. In doing so, there is a much-reduced chance of being non-compliant and having to pay fines.

Conclusion

Storage performance density is evolving as the most important requirement for Modern Application infrastructure decisions. Modern applications need extreme storage performance that scales, effectively utilizes storage capacity, doesn't requires additional vendors to protect the data, reduces data movement, eliminates cumbersome performance tuning and management, and simplifies compliance and security.

> *For more information about the Pavilion HFA visit our website or contact us directly.*
> *Website: http://www.pavilion.io/*
> *Email: info@pavilion.io*
> *Phone: (669) 263-6900*