

IBM XIV: Storage Architecture Evolution for High Performance

November 2008



For a number of years, vendors have been chasing the elusive holy grail of delivering breakthrough high-performance primary storage with traditional magnetic, rotating disks. The problem is that enterprises have long outgrown the ability of rotational disks to meet their performance demands, and each year, performance demands continue to increase exponentially. In turn, delivering high performance storage based on magnetic, rotating disk is more challenging than ever.

Historically, vendor attempts at improving storage solutions have often come by way of piling more and faster hardware components on top of aging architectures. Such improvements are incremental at best, and fast fade away in the face of constantly growing demands for performance.

This has not gone unnoticed in parts of the market where performance demands have been extreme. As is often the case, extreme fringes of the market – such as High Performance Computing, digital content editing, and others – have motivated select sets of vendors to innovate. Today, the performance needs of these fringe markets are rapidly becoming mainstream and associated storage solutions have turned to support the enterprise as well.

At Taneja Group, we've seen a number of vendors attack this challenge with new and innovative approaches to how data is stored on disk, and we have before classified such storage systems as Next Generation Block Storage. These solutions reach deep into stored data to optimize granular chunks of data (ranging in size from blocks to sets of blocks) in order to tune performance, increase utilization, and perform any number of advanced storage features more efficiently. The best of these solutions use their Next Generation Block Storage approach as a foundation on which they build performance and scalability.

In January 2008, IBM acquired one of these performant solutions – XIV – undoubtedly to serve as a key component of their *Information Infrastructure* approach to the data center. The XIV storage design team took the traditional approach to storage performance engineering, and turned the model on its head to deliver a truly innovative approach to high-performance, scalable storage. The IBM XIV Storage System drives performance from the bottom up, starting with disk modules that are aggregated together as distributed, intelligent, IO orchestrating, high performance building blocks designed to leverage every IO from every disk for maximum performance. In this Technology Brief, we'll take a look at challenges that drove the XIV design team to innovate, and the resulting XIV Storage System architecture.

Redefining Performance

The average storage manager is no doubt aware that storage performance is a precious resource today. For years, disk capacities, as well as processor, server, and application performance levels have each been growing at multiplicative to exponential annual rates. Meanwhile, disk performance has improved at best incrementally on annual levels, and today falls far short of delivering the performance the enterprise requires.

Today, IO stacks are also consolidating all around us – performance demands are exploding as more organizations perform more complex, HPC-like analytics than ever before; server virtualization is consolidating spread out machines onto high power virtual hosts that hammer fewer storage arrays with more IO than ever before; and new data types, such as rich media, are springing up in nearly every business around.

In turn, storage vendors have been challenged with the task of squeezing higher performance out of traditional storage architectures – many of which were designed with capacity aggregation rather than performance in mind. Most vendors have turned to cache as their salvation – creating complex buses for cache memory, clustering multiple controllers for bigger caches, or just piling increasing amounts of cache into existing dual controller configurations. Without a doubt, caching can offset the lagging performance of disk drives, but this approach to just

unintelligently storing and caching more blocks of data from ever-larger collections of disks is falling short of meeting today's demands.

The Market has Innovated

Researchers and innovators have not overlooked the shortcomings that existing architectures force upon storage. They have in fact turned to look deeper into the very nature of how blocks are stored and accessed, and how this process can be improved throughout the chain of components that make up networkable storage arrays.

In Taneja Group's view, these innovators are tackling a set of systemic shortcomings in traditional storage architectures:

1. Traditional block storage has been unintelligent about where data is stored, and fails to optimize the performance of every disk request.
2. Designing intelligent block management into the limited controllers of existing storage arrays can easily create bottlenecks as they try to manage the enormous numbers of blocks inside of large storage systems today.
3. Similarly, throwing huge amounts of specialized hardware and memory into a limited number of head-end controllers increases cost, while complicating internal component design, and still failing to deliver scalability beyond the finite limits of the controller hardware.
4. This same head-end controller architecture has created a complex tangle of pipes or connectivity inside of

traditional storage arrays, and these pipes constrain scalability and introduce hot spots, with little opportunity to redistribute, re-route, multi-link, or load balance traffic inside of a system. Increasing the performance of such systems often requires a vendor to completely re-engineer the entire internal fabric of an array.

5. Finally, traditional data protection architectures pay little attention to sustaining performance during commonplace component failures. Degraded performance levels due to failures, and the high overhead associated with parity-based recovery, encourage vendors to neglect the operational performance of their systems in order to buffer worst case performance.

While a handful of vendors have tackled these sticky issues with various different innovations – running the gamut from intelligent storage block management to scalable performance – only a few have taken on the challenge of comprehensively addressing traditional array shortcomings to comprehensively deliver new storage solutions that redefine enterprise class performance, availability, scalability, and capacity management. One solution that claims to have done just this is IBM's XIV Storage System.

The IBM XIV Storage System

The XIV storage array was brought to life by a team of innovators influenced by years of traditional high performance enterprise storage engineering. That team turned to

look at extreme requirements in the market, and how high performance storage could better be delivered. The XIV team found that many market niches with high performance needs were encountering a unique set of challenges. Not only were their storage performance demands growing off the charts, but they were facing more unpredictable needs than ever, and ending up with grossly wasted capacity on scattered, multiplying, hard to manage storage systems. These organizations needed a fundamental shift in how they were doing storage – a shift that would allow them to get more performance from fewer spindles with better capacity utilization, while providing less care and feeding than ever before.

In response, this team stood the traditional storage array model upside down, and focused first on small modular building blocks – what would traditionally be a disk shelf – that they infused with data block management intelligence and sophisticated caching algorithms. By doing so, that team turned a small collection of disk spindles into a high performance storage system building block. Then that team turned to connecting those blocks together with right-sized pipes, and collecting these intelligent building blocks into a powerful, massive XIV storage array.

XIV Architecture in Depth

To better understand XIV, we'll take a look at each building block of this modular, scalable array. We'll start with the smallest components – disks – and work our way up to the overarching layers of management

intelligence that orchestrate system-wide activities like XIV's intrinsic, always-on thin provisioning, innovative snapshots, and more.

Optimizing Disks

The XIV design team started their performance-centric problem solving by looking for a better way to aggregate raw disks themselves. The problem, in their view, is that the centralized controllers in traditional disk arrays cannot apply performance optimization and caching to the huge number of disks arrayed behind them. XIV distributes controller intelligence throughout an array, and associates a smaller number of disks with each controller. Specifically, an XIV array is made up of a number of Data Modules performing traditional controller functions on x86 hardware. Each Data Module, contains 8GB of ECC cache, and is connected to the XIV array with 8 1Gb Ethernet interfaces. It uses proprietary, highly tuned XIV algorithms to pseudo-randomly place 1MB data stripes across each of 12 internal SATA drives split across 2 PCIe SAS controllers. Doing so, fully distributes performance demands across each of the 12 spindles, regardless of the mix of workloads, hosts, or variation in workloads. Rather than maintaining unused spare drives, XIV also maintains spare capacity on each drive to host the data from any single or double drive failure. Any configuration brings high availability with it through dual power supplies in every component, redundant switches, and three fully redundant Uninterruptible Power Supplies (UPSs) in every rack of XIV components.

Not all caches are the same

The XIV design team created this grid-like architecture and harnessed the power of x86 hardware at a disk-shelf level to spread storage block management intelligence and caching intelligence throughout the XIV array, multiplying its effectiveness. This architecture combines disks as well as controller intelligence in every Data Module to create nearly fully independent building blocks that are loosely joined together into a grid-like XIV array that could start at any size, and easily grow to deliver nearly any imaginable configuration. Data Modules are responsible for optimizing data block placement, protecting against disk failures, performing caching, and carrying out a number of other storage controller functions. But perhaps most important of these functions is XIV's caching.

By turning the resources of a x86 processor to the task of caching data from disk spindles, over local dual PCIe buses, IBM is able to deliver extreme levels of performance. From a single rack of XIV, using 15 12 disk Data Modules, or 180 SATA drives, IBM can deliver 100,000 IOPS from cache, 20,000 IOPs from disk, and 2.4GBps and 1.4GBps of sustained sequential read and write bandwidth respectively. The secret behind delivering this level of performance is XIV's optimization of every IO in combination with intelligent caching algorithms that rapidly parse and refine cached data. Those algorithms, in IBM's labs, have demonstrated a 65% cache hit ratio with fully randomized workloads, compared to previous best-in-class cache efficiencies of 50%. With tremendous potential linear scalability not yet on the

TECHNOLOGY BRIEF

market but so far demonstrated in that same lab, XIV arrays stand poised to aggregate economy building blocks into huge pools of disk and globally shared, high performance cache.

The reason XIV caches so efficiently, is an XIV Data Module optimizes every IO request that comes to the system, by making the most of every disk access to cache data, and then applying XIVs unique IP to optimize which data is retained or purged from cache. If underlying disk performance can handle it, an XIV Data Module will take a small IO and turn it into a large read request to the disk – optimally as large as a full 1MB stripe – in order to retrieve more data and populate more cache. Once in cache, XIV then parses and optimizes this data in 4KB blocks, allowing it to perform granular and powerful optimization. For every potential IO that is served from cache instead of disk – leaving potential additional disk performance unused – XIV may execute another IO to disk to read additional data into cache. In turn, XIV is automatically and constantly performance optimized in the background, with no manual tuning required, regardless of application variations.

Moreover, since caching and disk access all happens over an isolated, local PCIe bus, XIV has enormous IO bandwidth to disk, and can perform cache and disk optimizations without impacting any other component in the storage system. XIV has effectively replaced the complex arbitrated loop or switched connections between controllers and disks in traditional arrays with a distributed, high bandwidth, PCIe

and SATA bus with exponentially greater bandwidth. In turn, the XIV design team has turned a single Data Module into a powerhouse of block data storage that can squeeze more sustainable performance from a single set of disk drives than previously thought possible.

Pipes and Plumbing

The XIV design team built Data Modules with the intent of connecting them together into larger arrays, and did so with an attentiveness to every component of connectivity within the system. Traditional storage arrays have taken a top heavy approach, where a big controller is fed by components with lower performance, connected by smaller pipes, that might have other components in them with yet less performance. As a whole, a well-designed system with this approach may perform and operate well. But when a single component is replaced with a higher performance one, the model falls apart as the connectivity and other components of the system may not support the aggregate performance.

The XIV design team set out to distribute intelligence closer to the disk within an XIV storage system, to overbuild performance in the smallest building blocks, and to make connectivity and front end controllers into disposable components that can easily be replaced or upgraded as processors get faster and/or interfaces change and mature – potentially including faster FC, 10Gb Ethernet, InfiniBand, and FCoE. XIV literally turned the architecture of existing arrays upside down.

Orchestrating an Array of Modules

The XIV design team took the position that the point of ingress for an array should do little else besides connect hosts and orchestrate and distribute data and storage management requests to intelligent subcomponents – the Data Modules – within an XIV array. In the XIV approach, this point of ingress is called an Interface Module. Interface Modules are actually Data Modules that have external Fibre Channel and iSCSI interfaces, and coordinate storage array tasks – including snapshots, volume configuration, and identification of which Data Module controls which blocks. An XIV system can scale in number of interfaces by aggregating together multiple Interface Modules (currently up to 4 Interface Modules serving up 24 FC and 6 iSCSI interfaces) that cluster together and synchronously track data blocks between them. Interface Modules and Data Modules are connected together by a web of switched gigabit Ethernet connectivity (currently 4 ports per data module and 8 ports per Interface Module) that assures every component has sufficient connectivity to fully saturate all of the host facing ports in an XIV system. Incoming hosts connections are load balanced across Interface Modules through manually assigned connections and/or their use of multi-path agents, but inside an XIV system, high performance load balancing is yet another feature.

Data Placement and Protection

Early on, the XIV design team evaluated protection architectures, and decided against parity-based approaches. XIV instead protects data by mirroring 1MB

data “chunks” on different spindles and Data Modules across the system. Chunk mirroring can apply more disk resources and performance to every read or write request, tolerate failures with almost zero performance impact, and protect data integrity.

Extreme Performance

Chunk mirroring in small 1MB chunks frees each half of a chunk mirror from locality restrictions – meaning a chunk is not tied to a single disk set or shelf as would be the case with many traditional parity-based storage systems. Once chunk mirrors are freed from location restrictions, they can be placed on other Data Modules within an XIV system. Placing each half on a separate Data Module allows the Interface Module to load balance read requests across multiple nodes and effectively double performance. Moreover, randomizing the distribution of every mirrored chunk across all Data Modules in an XIV system can load balance the entire array and avoid hotspots anywhere in the system. This approach to performance optimization – randomized chunk mirror distribution and n-way load balancing – has been recognized among experts in specialized fields (such as High Performance Computing and Video-on-Demand) as the best potential approach to constantly delivering maximum block storage performance in the face of unpredictable demands.

Extreme, Zero-Impact Failure Tolerance

Traditional parity protection technology today is looking less and less up to the task of supporting enterprise storage. As disk

T E C H N O L O G Y B R I E F

drives become denser, faster, and hotter the chances of a failure are increased. Today, more users than ever before are likely to face a failure during a typical storage system's operational lifespan. What's worse, a failure with traditional parity schemes grinds these systems to a halt on two fronts. First, operating with one less disk drive can make what used to be a single IO into multiple IOs because the storage system has to read or write multiple disks in a set and perform parity calculations to compensate for missing data. This can make sophisticated storage systems grind to a halt as these extra IOs cut deeply into the limited amount of IO each disk spindle has to begin with. Second, with a failed disk, most arrays immediately start attempting to rebuild to a hot spare disk. If rebuild takes place with any priority, this can again cut deeply into the array's limited performance, and make production IO grind to a halt. If rebuild is not prioritized, there is greatly increased risk that another disk could fail during what could be a multiple day rebuild.

The XIV design team architected the controlling logic in the XIV Interface Module to randomly distribute each chunk mirror across Data Modules, and make sure that no two copies of the same chunk exist within the same Data Module. By doing so, XIV has achieved extreme fault tolerance and mitigated the challenges associated with recovering from failures. Since every block of data is distributed XIV can tolerate a single disk, or a complete Data Module, failure. Moreover, during a disk failure, XIV redistributes and rebalances data across the remaining spindles in the involved Data

Module as well as all other Data Modules in the system. Since every disk in the system is involved, this happens with negligible impact on system performance. XIV can also tolerate a complete Data Module failure, and then recreates mirrored chunks from other Data Modules and redistributes those chunks across all other Data Modules. XIV again relies on proprietary algorithms that optimize this data redistribution with minimal change in data, and with minimal transfer of data by any single node. Whether a disk is involved, or a Data Module is involved, IBM claims that there is still minimal impact on system performance, aside from the loss of IO associated with a complete Data Module failure. This is because rebuild is a simple copy process, allowing reads and writes to continue free from the additional parity calculation that can drag traditional systems to a crawl, and the copy-rebuild process itself can copy, write, or redistribute data using all nodes in the system. Compared to hours, or potentially even days, required to rebuild traditional parity drive sets, IBM claims XIV can fully rebuild a 1TB drive loss in 30 minutes, with only negligible IO performance impact (3-5%).

Extreme Data Integrity

XIV also leverages these mirrored chunks for data scrubbing and integrity monitoring. XIV always performs simultaneous writes of each chunk mirror on separate Data Modules. Since XIV's mirrored chunk approach has negligible performance impact from drive loss, XIV monitors SMART data from disks and performs aggressive disk drive shutdown when anomalies are detected. But XIV also uses

T E C H N O L O G Y B R I E F

mirrored chunks for additional data scrubbing and verification. Since every 1MB chunk is check summed, during periods of low IO, XIV scrubs data by verifying it against checksums, optimizing its placement on disk, and replacing any invalid chunks with its mirrored half from another Data Module.

Thin Intelligence

Finally, orchestrated by the Interface Module, intelligent block management (in the form of chunks) is fundamental to the XIV storage approach. Since each Interface Module coordinates the distribution of each data block to underlying Data Modules, the Interface Module also reaches deeper into data blocks to understand the nature of the data being stored – specifically whether a data block really contains data or is in fact empty space. XIV then optimizes data storage by only storing real data. Moreover, during XIV's routine data scrubbing, any identified empty data blocks are removed, freeing up space from deleted block data. Even if volumes have been migrated from traditional storage, empty space will be identified and freed. This level of block management is built into XIV, and makes every XIV storage volume fundamentally thin, and keeps it that way.

Extreme Snapshots

XIV leverages a similar approach for snapshots as well. With a snapshot, XIV effectively locks the original data blocks as read-only and shared by the snapshot, and then begins to accrue changes as new blocks with what we call redirect-on-write snapshots (again, real data only, not empty space). Each snapshot is immediately fully

writable, at the same performance as any full XIV volume. Snapshots read at full performance because chunks and chunk mirrors are randomly and fully distributed across all nodes in an array, harnessing the full capability of the entire array for reads from shared data blocks. Snapshots write at full performance because only the changes in existing data are written to disk, and these writes are cached by XIV's globally distributed cache (currently 120GB) and randomly distributed and mirrored across all of the disks in an array. Finally, highly unique to XIV and demonstrative of the power of XIV's intelligent block management, these redirect-on-write snapshots are fully independent – any snapshot in a series can be deleted, without harming snapshots that were dependent upon it. XIV supports up to 16,000 of these zero impact, unlinked snapshots on an array.

XIV's highly efficient full volume copy process also has zero performance impact, as the duplication of data blocks for any given volume happens entirely over the local internal PCIe buses of an XIV Data Module. While volume copy may have its place, in our opinion, with XIV's built-in thin-provisioning, high availability and deep data protection built on chunk mirroring, snapshots become the preferred approach to creating local replicas, as an XIV snapshot offers all of the functionality of traditional clones.

Extreme Ease of Use

Finally, and unlikely to be easily outdone by other solutions on the market, XIV delivers storage technology simplification. By

T E C H N O L O G Y B R I E F

making all of this technology pervasive – including intelligent block management, thin provisioning, and more – XIV can hide it well beneath the surface where administrators never need touch it or manage it. In turn, administrators can focus on simplified always-thin provisioning and data protection. XIV's ability to optimize every storage volume, without tradeoffs, removes the need to analyze each decision about volume types, stripe widths, rebuild levels, performance impacts, thin utilization, or snapshot performance impact common to traditional arrays. The result is that administrator effort can scale just as effectively as XIV storage.

Taneja Group Opinion

In late 2007, Taneja Group defined an emerging category of block storage arrays that we labeled Next Generation Block Storage (*Next Generation FC Systems Market Profile, October 2007, Taneja Group*). XIV firmly fits into this category by intelligently managing and optimizing stored data blocks at the sub-volume level (in XIV's case, chunks). We firmly believe such arrays will largely displace traditional storage arrays that simply cannot intelligently and continually optimize performance, capacity and protection for increasingly consolidated, virtualized, and high performance enterprise systems.

Moreover, XIV has shifted the perspective with which traditional storage systems have been designed, and represents the intersection of deeply innovative architecture with next generation block storage intelligence. Instead of overlaying

insufficient architectures with increasingly sophisticated but complex caching and controllers, XIV inverted this top down approach, and tackled the fundamental performance issues that exist at the disk level. Other innovators are doing this today, but perhaps none with such a systematic, performance-centric orientation as XIV. XIV has innovated from the inside by methodically examining every component of the traditional storage system, and redesigning every linkage to apply the best possible performance optimization from the bottom up.

In our view, XIV is simply the next generation of previous enterprise storage architectures. The model remains similar – backend controllers manage disk, front end controllers manage connectivity, and data, caching, and overall management operations remain distributed widely enough across that architecture for end to end redundancy without impact to performance in the event of any failure. Similar to existing enterprise architectures, modules can be hot-plugged and replaced, mixed, or upgraded, without impact to other modules in the system. The change with XIV revolves around delivering all of these capabilities with industry standard components, with better economies and less proprietary lock-in than ever before. Based on what we've seen of its design and architecture, XIV remains unstoppable like the best of enterprise storage, but may be fundamentally more extensible, scalable, and adaptable as technologies evolve. Without a doubt, it brings a new set of economics to the table for enterprise-level storage.

T E C H N O L O G Y B R I E F

When viewed with an enterprise mindset it looks like XIV will rapidly have significant impact, especially when brought to the table as a key component of the IBM Information Infrastructure and backed up by IBM global support capabilities. But the enterprise is not the only sphere of impact for XIV. XIV is a highly available system that can tolerate multiple failures without much maintenance, and at the same time be administered with very little impact. That puts XIV in a good position for service providers or other large infrastructures that need many systems, and can't necessarily manage individual systems as often and as well as traditional storage arrays require. This, at the primary storage level, is in fact a key tenet of IBM's Information Infrastructure approach – all components of the infrastructure should be better enabled for easy, intelligent data management that automatically optimizes the infrastructure to support the deluge of

data and information that IT customers the world over are experiencing. We suspect XIV will go far in defining what ideal next generation block storage architectures look like for such customers.

Moreover, IBM's current presentation of XIV just skims the surface of the potential hidden in the intellectual property (IP) behind this platform. IBM has long been expert at acquiring key IP and integrating it into IBM solution sets in innovative ways. XIV's intelligent block management, distributed caching architecture, aggressive cache optimization algorithms, randomized distributed mirrored chunk algorithms, and other IP will likely influence IBM solutions for years to come. From web infrastructures to storage systems, we're confident the IBM XIV Storage System is only the beginning for this innovative technology.

***NOTICE:** The information and product recommendations made by the TANEJA GROUP are based upon public information and sources and may also include personal opinions both of the TANEJA GROUP and others, all of which we believe to be accurate and reliable. However, as market conditions change and not within our control, the information and recommendations are made without warranty of any kind. All product names used and mentioned herein are the trademarks of their respective owners. The TANEJA GROUP, Inc. assumes no responsibility or liability for any damages whatsoever (including incidental, consequential or otherwise), caused by your use of, or reliance upon, the information and recommendations presented herein, nor for any inadvertent errors which may appear in this document.*