

INTRODUCTION TO THE EMC XTREMIO STORAGE ARRAY (Ver. 3.0)

A Detailed Review

Abstract

This white paper introduces the EMC XtremIO Storage Array. It provides detailed descriptions of the system architecture, theory of operation, and its features. It also explains how the XtremIO's unique features (such as Inline Data Reduction techniques [including inline deduplication and data compression], scalable performance, data protection, etc.) provide solutions to data storage problems that cannot be matched by any other system.

December 2014

Copyright © 2014 EMC Corporation. All Rights Reserved.

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

The information in this publication is provided "as is". EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

For the most up-to-date listing of EMC product names, see EMC Corporation Trademarks on EMC.com.

VMware is registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other trademarks used herein are the property of their respective owners.

Part Number H11752.6 (Rev. 07)

Table of Contents

Executive Summary	4
Introduction	5
System Overview	6
X-Brick	7
Scale-Out Architecture	9
10TB Starter X-Brick (5TB)	10
System Architecture	11
Theory of Operation	13
Mapping Table	13
How the Write I/O Flow Works	14
How the Read I/O Flow Works	19
System Features	20
Thin Provisioning	21
Inline Data Reduction	21
Inline Data Deduplication	21
Inline Data Compression	23
Total Data Reduction	24
XtremIO Data Protection (XDP)	25
How XDP Works	26
Data at Rest Encryption	28
Snapshots	30
Scalable Performance	35
Even Data Distribution	38
High Availability	39
Non-Disruptive Upgrade	40
VMware VAAI Integration	41
XtremIO Management Server (XMS)	45
System GUI	46
Command Line Interface	48
RESTful API	48
LDAP/LDAPS	48
Ease of Management	49
Integration with other EMC Products	50
Powerpath	50
VPLEX	50
RecoverPoint	51
Solutions Brief	52
Openstack Integration	54
Conclusion	55

Executive Summary

Flash storage is an attractive method for boosting I/O performance in the data center. But it has always come at a price, both in high costs and loss of capabilities like scalability, high availability, and enterprise features.

XtremIO's 100% flash-based scale-out enterprise storage array delivers not only high levels of performance and scalability, but also brings new levels of ease-of-use to SAN storage, while offering advanced features that have never been possible before.

XtremIO's ground-up all-flash array design was created from the start for maximum performance and consistent low latency response times, and with enterprise grade high availability features, real-time Inline Data Reduction that dramatically lowers costs, and advanced functions such as thin provisioning, tight integration to VMware, snapshots, volume clones, and superb data protection.

This is achieved with a competitive cost of ownership. The product architecture addresses all the requirements for flash-based storage, including achieving longevity of the flash media, lowering the effective cost of flash capacity, delivering performance and scalability, providing operational efficiency, and delivering advanced storage array functionality.

This white paper provides a broad introduction to the XtremIO Storage Array, with detailed descriptions of the system architecture, theory of operation, and its various features.

Introduction

XtremIO is an all-flash storage array that has been designed from the ground-up to unlock flash's full performance potential and deliver array-based capabilities that leverage the unique characteristics of SSDs, based on flash media.

XtremIO uses industry standard components and proprietary intelligent software to deliver unparalleled levels of performance. Achievable performance ranges from hundreds of thousands to millions of IOPS, and consistent low latency of under one millisecond.*

The system is also designed to provide minimal planning, with a user-friendly interface that makes provisioning and managing the array very easy.

XtremIO leverages flash to deliver value across the following main dimensions:

- **Performance** – Regardless of how busy the system is, and regardless of storage capacity utilization, latency and throughput remain consistently predictable and constant. Latency within the array for an I/O request is typically far less than one millisecond.*
- **Scalability** – The XtremIO storage system is based on a scale-out architecture. The system begins with a single building block, called an X-Brick. When additional performance and capacity are required, the system scales out by adding X-Bricks. Performance scales linearly, ensuring that two X-Bricks supply twice the IOPS, four X-Bricks supply four times the IOPS and six X-Bricks supply six times the IOPS of the single X-Brick configuration. Latency remains consistently low as the system scales out.
- **Efficiency** – The core engine implements content-based Inline Data Reduction. The XtremIO Storage Array automatically reduces (deduplicates and compresses) data on the fly, as it enters the system. This reduces the amount of data written to flash, improving longevity of the media and driving down cost. XtremIO arrays allocate capacity to volumes on-demand in granular data blocks. Volumes are always thin-provisioned without any loss of performance, over-provisioning of capacity, or fragmentation. Once content-based inline deduplication is implemented, the remaining data is compressed even further, reducing the amount of writes to the flash media. The data compression is carried out inline on the deduplicated (unique) data blocks.

Benefits gained from avoiding a large percentage of writes include:

- Better performance due to reduced data
- Increased overall endurance of the flash array's SSDs
- Less required physical capacity to store the data, increasing the storage array's efficiency and dramatically reducing the \$/GB cost of storage

* As measured for small block sizes. Large block I/O by nature incurs higher latency on any storage system.

- **Data Protection** – XtremIO leverages a proprietary flash-optimized data protection algorithm (XtremIO Data Protection or XDP), which provides performance that is superior to any existing RAID algorithm. Optimizations in XDP also result in fewer writes to flash media for data protection purposes.
- **Functionality** – XtremIO supports high performance and space-efficient snapshots, Inline Data Reduction (including inline deduplication and data compression), thin provisioning, and full VMware VAAI integration, as well as support for Fibre Channel and iSCSI protocols.

System Overview

The XtremIO Storage Array is an all-flash system, based on a scale-out architecture. The system uses building blocks, called X-Bricks, which can be clustered together to grow performance and capacity as required, as shown in [Figure 2](#).

The system operation is controlled via a stand-alone dedicated Linux-based server, called the XtremIO Management Server (XMS). Each XtremIO cluster requires its own XMS host, which can be either a physical or a virtual server. The array continues operating if it is disconnected from the XMS, but cannot be configured or monitored.

XtremIO's array architecture is specifically designed to deliver the full performance potential of flash, while linearly scaling all resources such as CPU, RAM, SSDs, and host ports in a balanced manner. This allows the array to achieve any desired performance level, while maintaining consistency of performance that is critical to predictable application behavior.

The XtremIO Storage System provides a very high level of performance that is consistent over time, system conditions and access patterns. It is designed for true random I/O.

The system's performance level is not affected by its capacity utilization level, number of volumes, or aging effects. Moreover, performance is not based on a "shared cache" architecture and therefore it is not affected by the dataset size or data access pattern.

Due to its content-aware storage architecture, XtremIO provides:

- Even distribution of data blocks, inherently leading to maximum performance and minimal flash wear
- Even distribution of metadata
- No data or metadata hotspots
- Easy setup and no tuning
- Advanced storage functionality, including Inline Data Reduction (deduplication and data compression), thin provisioning, advanced data protection (XDP), snapshots, and more

X-Brick

Figure 1 shows an X-Brick.

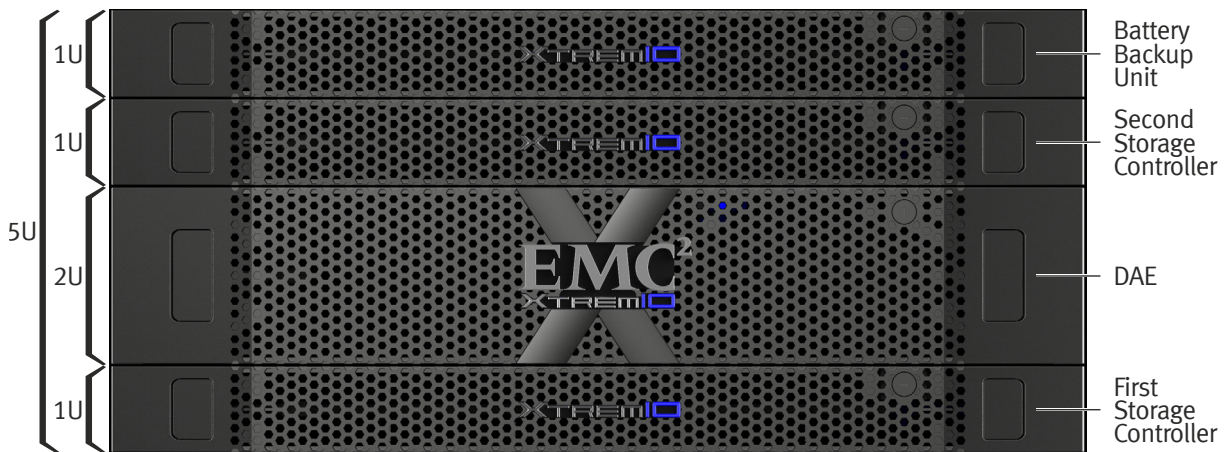


Figure 1. X-Brick

An X-Brick is the basic building block of an XtremIO array.

Each X-Brick is comprised of:

- One 2U Disk Array Enclosure (DAE), containing:
 - 25 eMLC SSDs (standard X-Brick) or 13 eMLC SSDs (10TB Starter X-Brick [5TB])
 - Two redundant power supply units (PSUs)
 - Two redundant SAS interconnect modules
- One Battery Backup Unit
- Two 1U Storage Controllers (redundant storage processors)

Each Storage Controller includes:

- Two redundant power supply units (PSUs)
- Two 8Gb/s Fibre Channel (FC) ports
- Two 10GbE iSCSI ports
- Two 40Gb/s InfiniBand ports
- One 1Gb/s management/IPMI port

Table 1 shows the system specifications per X-Brick.

Table 1. System Specifications (Per X-Brick)

Feature	Specification (per X-Brick)
Physical	<ul style="list-style-type: none"> • 5U • 13 x eMLC Flash SSDs (10TB Starter X-Brick [5TB]) • 25 x eMLC Flash SSDs (Regular X-Brick)
High Availability	<ul style="list-style-type: none"> • Redundant • Hot swap components • No single point of failure (SPOF)
Host Access	Symmetrical Active/Active – Any volume can be accessed in parallel from any target port on any controller with equivalent performance. There is no need for ALUA.
Host Ports	<ul style="list-style-type: none"> • 4 x 8Gb/s FC • 4 x 10Gb/s Ethernet iSCSI
Usable Capacity*	<ul style="list-style-type: none"> • For a 10TB Starter X-Brick (5TB) type: <ul style="list-style-type: none"> - 3.26TiB (13 SSDs, with no data reduction) - 7.22TiB (25 SSDs, with no data reduction) • For a 10TiB X-Brick type: <ul style="list-style-type: none"> 7.58TiB (with no data reduction) • For a 20TB X-Brick type: <ul style="list-style-type: none"> 15.16TiB (with no data reduction)
Latency	Less than one millisecond [†]

* Usable capacity is the amount of unique, non-compressible data that can be written into the array. Effective capacity will typically be much larger due to XtremIO's Inline Data Reduction. The final numbers might be slightly different.

[†] Sub-millisecond latency applies to typical block sizes. Latency for small blocks or large blocks may be higher.

Scale-Out Architecture

An XtremIO storage system can include a single X-Brick or a cluster of multiple X-Bricks, as shown in Figure 2 and Table 2.*

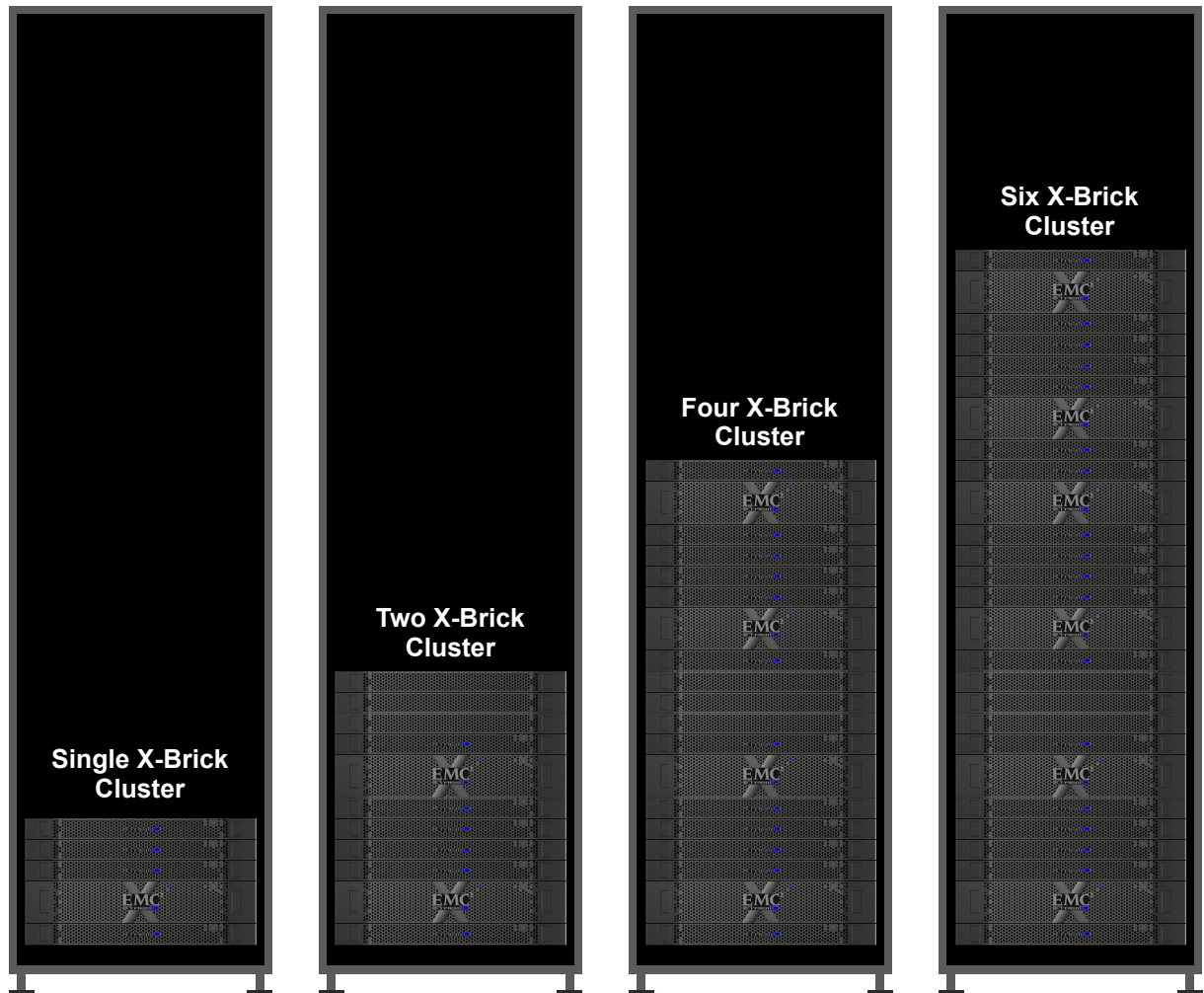


Figure 2. System Configurations as Single and Multiple X-Brick Clusters

With clusters of two or more X-Bricks, XtremIO uses a redundant 40Gb/s QDR InfiniBand network for back-end connectivity between the Storage Controllers, ensuring a highly available, ultra-low latency network. The InfiniBand network is a fully managed component of the XtremIO array and administrators of XtremIO systems do not need to have specialized skills in InfiniBand technology.

* System version 3.0 supports up to six X-Bricks in a cluster (available from Q4'14). This will continue to increase in subsequent releases of the XtremIO Operating System.

A single X-Brick cluster consists of:

- One X-Brick
- One additional Battery Backup Unit

A multiple X-Brick cluster consists of:

- Two or four X-Bricks
- Two InfiniBand Switches

Table 2. System Configurations as Single and Multiple X-Brick Clusters

	10TB Starter X-Brick (5TB)	One X-Brick Cluster	Two X-Brick Cluster	Four X-Brick Cluster	Six X-Brick Cluster
No. of X-Bricks	1	1	2	4	6
No. of InfiniBand Switches	0	0	2	2	2
No. of Additional Battery Backup Units	1	1	0	0	0

10TB Starter X-Brick (5TB)

The XtremIO's 10TB Starter X-Brick (5TB) is identical to a standard X-Brick cluster, but it is equipped with only 13 eMLC Flash SSDs instead of 25. The 10TB Starter X-Brick (5TB) can be expanded to a regular X-Brick by adding 12 SSDs.

System Architecture

XtremIO works like any other block-based storage array and integrates with existing SANs, with a choice of 8Gb/s Fibre Channel and 10Gb/s Ethernet iSCSI (SFP+) connectivity to the hosts.

However, unlike other block arrays, XtremIO is a purpose-built flash storage system, designed to deliver the ultimate in performance, ease-of-use and advanced data management services. Each Storage Controller within the XtremIO array runs a specially customized lightweight Linux distribution as the base platform. The XtremIO Operating System (XIOS), runs on top of Linux and handles all activities within a Storage Controller, as shown in [Figure 3](#). XIOS is optimized for handling high I/O rates and manages the system's functional modules, the RDMA over InfiniBand operations, monitoring and memory pools.

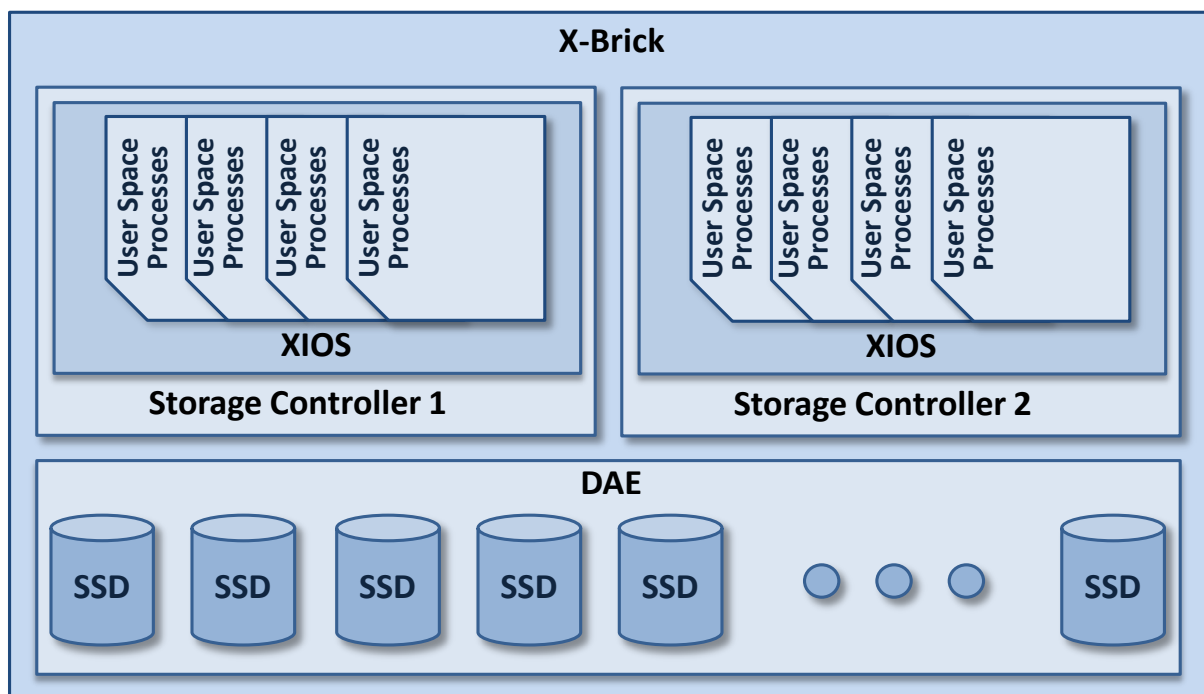


Figure 3. X-Brick Block Diagram

XIOS has a proprietary process-scheduling-and-handling algorithm, which is designed to meet the specific requirements of the content-aware, low latency, and high performance storage subsystem.

XIOS provides:

- Low-latency scheduling – to enable efficient context switching of sub-processes, optimize scheduling and minimize wait time
- Linear CPU scalability – to enable full exploitation of any CPU resources, including multi-core CPUs
- Limited CPU inter-core sync – to optimize the inter-sub-process communication and data transfer
- No CPU inter-socket sync – to minimize synchronization tasks and dependencies between the sub-processes that run on different sockets
- Cache-line awareness – to optimize latency and data access

The Storage Controllers on each X-Brick own the disk array enclosure (DAE) that is attached to them via redundant SAS interconnects. The Storage Controllers are also connected to a redundant and highly available InfiniBand fabric. Regardless of which Storage Controller receives an I/O request from a host, multiple Storage Controllers on multiple X-Bricks cooperate to process the request. The data layout in the XtremIO system ensures that all components inherently share the load and participate evenly in I/O operations.

Theory of Operation

The XtremIO Storage Array automatically reduces (deduplicates and compresses) data as it enters the system, processing it in data blocks. Deduplication is global (over the entire system), is always on, and is performed in real-time (never as a post-processing operation). After the deduplication the data is compressed inline, before it is written to the SSDs.

XtremIO uses a global memory cache, which is aware of the deduplicated data, and content-based distribution that inherently spreads the data evenly across the entire array. All volumes are accessible across all X-Bricks and across all storage array host ports.

The system uses a highly available back-end InfiniBand network (supplied by EMC) that provides high speeds with ultra-low latency and Remote Direct Memory Access (RDMA) between all storage controllers in the cluster. By leveraging RDMA, the XtremIO system is in essence a single, shared memory space spanning all storage controllers.

The effective logical capacity of a single X-Brick varies depending upon the data set being stored.

- For highly duplicated information, which is typical of many virtualized cloned environments, such as Virtual Desktop Integration (VDI), the effective usable capacity is much higher than the available physical flash capacity. Deduplication ratios in the range of 5:1 to 10:1 are routinely achieved in such environments.
- For compressible data, which is typical in many databases and in application data, compression ratios are in 2:1 to 3:1 range.
- Systems benefitting from both data compression and data deduplication, such as Virtual Server Infrastructures (VSI), commonly achieve a 6:1 ratio.

Mapping Table

Each Storage Controller maintains a table that manages the location of each data block on SSD, as shown in [Table 3](#) (on page 14).

The table has two parts:

- The first part of the table maps the host LBA to its content fingerprint.
- The second part of the table maps the content fingerprint to its location on SSD.

















Using the second part of the table provides XtremIO with the unique capability to distribute the data evenly across the array and place each block in the most suitable location on SSD. It also enables the system to skip a non-responding drive or to select where to write new blocks when the array is almost full and there are no empty stripes to write to.

How the Write I/O Flow Works

In a typical write operation, the incoming data stream reaches any one of the Active-Active Storage Controllers and is broken into data blocks. For every data block, the array fingerprints the data with a unique identifier.

The array maintains a table with this fingerprint (as shown in Table 3) to determine if incoming writes already exist within the array. The fingerprint is also used to determine the storage location of the data. The LBA to content fingerprint mapping is recorded in the metadata, within the Storage Controller's memory.

Table 3. Mapping Table Example

	LBA Offset		Fingerprint		SSD Offset / Physical Location	
 →	Address 0	→	20147A8	→	40	→ 
 →	Address 1	→	AB45CB7	→	8	→ 
 →	Address 2	→	F3AFBA3	→	88	→ 
 →	Address 3	→	963FE7B	→	24	→ 
 →	Address 4	→	0325F7A	→	64	→ 
 →	Address 5	→	134F871	→	128	→ 
 →	Address 6	→	CA38C90	→	516	→ 
 →	Address 7	→	963FE7B	-	Deduplicated	- 

Note:

In Table 3, the colors of the data blocks correspond to their contents. Unique contents are represented by different colors while duplicate contents are represented by the same color (red).

The system checks if the fingerprint and the corresponding data block have already been stored previously.

If the fingerprint is new, the system:

- Compresses the data.
- Chooses a location on the array where the block will go (based on the fingerprint, and not the LBA).
- Creates the "fingerprint to physical location" mapping.
- Increments the reference count for the fingerprint by one.
- Performs the write.

In case of a "duplicate" write, the system records the new LBA to fingerprint mapping, and increments the reference count on this specific fingerprint. Since the data already exists in the array, it is neither necessary to change the fingerprint to physical location mapping nor to write anything to SSD. All metadata changes occur within the memory. Therefore, the deduplicated write is carried out faster than the first unique block write. This is one of the unique advantages of XtremIO's Inline Data Reduction, where deduplication actually improves the write performance.

The actual write of the data block to SSD is carried out asynchronously. At the time of the application write, the system places the data block into the in-memory write buffer (which is protected by replicating to different Storage Controllers via RDMA), and immediately returns an acknowledgement to the host. When enough blocks are collected in the buffer, the system writes them to the XDP (XtremIO Data Protection) stripe(s) on SSDs. This process is carried out in the most efficient manner, and is explained in detail in the XtremIO Data Protection White Paper.

When a Write I/O is issued to the array:

1. The system analyzes the incoming data and segments it into data blocks, as shown in Figure 4.



Figure 4. Data Broken into Fixed Blocks

2. For every data block, the array allocates a unique fingerprint to the data, as shown in Figure 5.

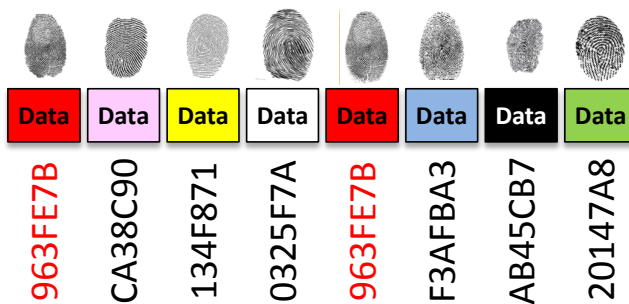


Figure 5. Fingerprint Allocated to Each Block

The array maintains a table with this fingerprint to determine if subsequent writes already exist within the array, as shown in Table 3 (on page 14).

- If a data block does not exist in the system, the processing Storage Controller journals its intention to write the block to other Storage Controllers, using the fingerprint to determine the location of the data.
- If a data block already exists in the system, it is not written, as shown in Figure 6.

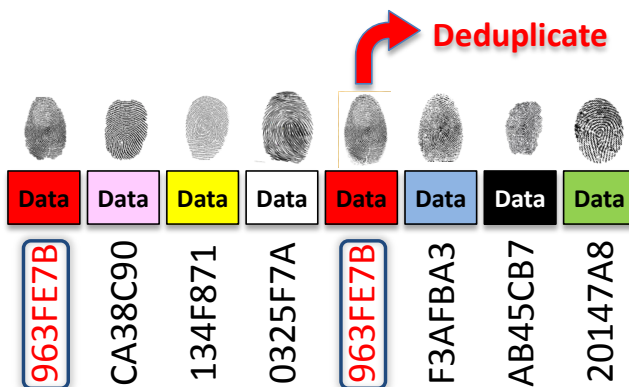


Figure 6. Deduplicating the Existing/Repeated Block

3. The array increases the reference count for each data block.
4. Using a consistent distributed mapping, each block is routed to the Storage Controller that is responsible for the relevant fingerprint address space.

The consistent distributed mapping is based on the content fingerprint. The mathematical process that calculates the fingerprints results in a uniform distribution of fingerprint values and the fingerprint mapping is evenly spread among all Storage Controllers in the cluster, as shown in Figure 7.

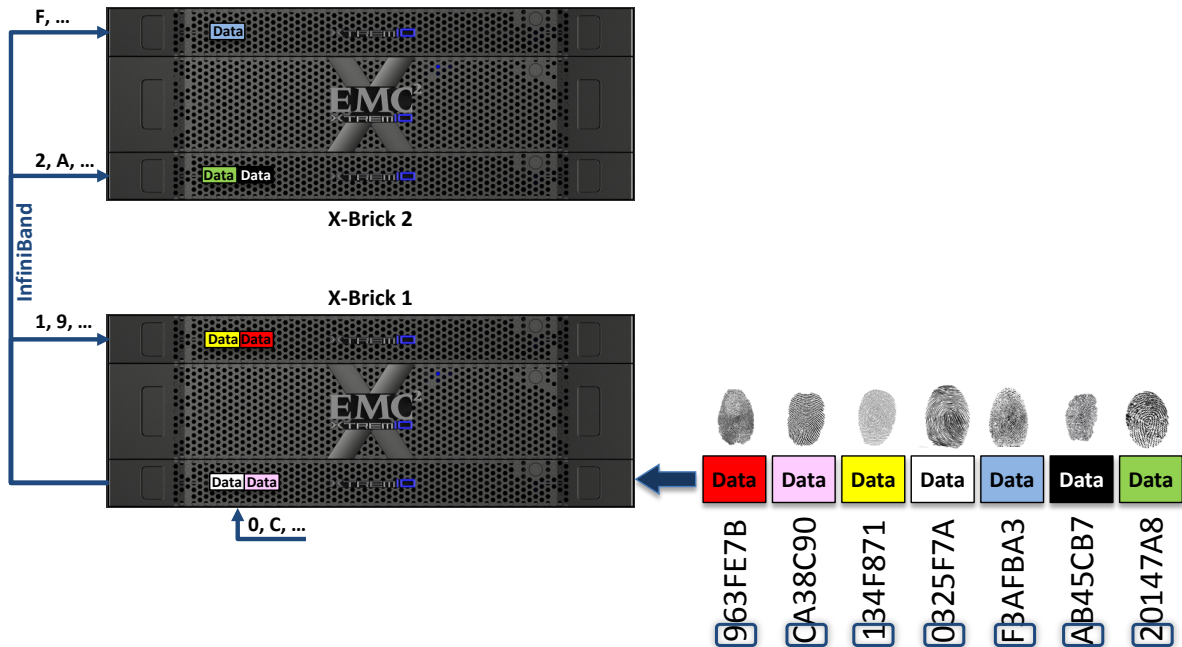


Figure 7. Data Spread Across the Cluster

Note:

Data transfer across the cluster is carried out over the low latency, high-speed InfiniBand network using RDMA, as shown in Figure 7.

5. The system sends back an acknowledgement to the host.

- Due to the even distribution of the fingerprint function, each Storage Controller in the cluster receives an even share of the data blocks. When additional blocks arrive they populate the stripes, as shown in [Figure 8](#).



X-Brick 2

X-Brick 1



Figure 8. Additional Blocks Populated Full Stripes

- The system compresses the data blocks to further reduce the size of each data block.
- Once a Storage Controller has enough data blocks to fill the emptiest stripe in the array (or a full stripe if available), it transfers them from the cache to SSD, as shown in [Figure 9](#).



X-Brick 2

X-Brick 1



Figure 9. Stripes Committed to SSDs

How the Read I/O Flow Works

In a data block read operation, the system performs a look-up of the logical address in the LBA to fingerprint mapping. Once the fingerprint is found, it looks up the fingerprint to physical mapping, and retrieves the data block from the specific physical location. Because the data is evenly written across the cluster and SSDs, the read load is also evenly shared.

XtremIO has a memory-based read cache in each Storage Controller.

- In traditional arrays, the read cache is organized by logical addresses. Blocks at addresses that are more likely to be read are placed in the read cache.
- In the XtremIO array, the read cache is organized by content fingerprint. Blocks whose contents (represented by their fingerprint IDs) are more likely to be read are placed in the cache.

This makes the XtremIO's read cache deduplication aware, which means that its relatively small read cache appears much larger than traditional caches of the same size.

If the requested block size is larger than the data block size, XtremIO performs parallel data block reads across the cluster and assembles them into bigger blocks, before returning them to the application.

A compressed data block is decompressed before it is delivered.

When a Read I/O is issued to the array:

1. The system analyzes the incoming request to determine the LBA for each data block and creates a buffer to hold the data.
2. The following processes occur in parallel:
 - For each data block, the array finds the stored fingerprint. The fingerprint determines the location of the data block on an X-Brick. For larger I/Os (e.g. 256K), multiple X-Bricks are involved in retrieving each data block.
 - The system transmits the requested Read data to the processing Storage Controller, via RDMA, over InfiniBand.
3. The system sends the fully populated data buffer back to the host.

System Features

The XtremIO Storage Array offers a wide range of features that are always available and do not require special licenses.

System features include:

- Data services features – applied in sequence (as listed below) for all incoming writes:
 - Thin Provisioning
 - Inline Data Reduction:
 - Inline Data Deduplication
 - Inline Data Compression
 - XtremIO Data Protection (XDP)
 - Data at Rest Encryption
 - Snapshots
- System-wide features:
 - Scalable Performance
 - Even Data Distribution
 - High Availability
- Other features:
 - Non-Disruptive Upgrade
 - VMware VAAI Integration

Thin Provisioning

XtremIO storage is natively thin provisioned, using a small internal block size. This provides fine-grained resolution for the thin provisioned space.

All volumes in the system are thin provisioned, meaning that the system consumes capacity only when it is actually needed. XtremIO determines where to place the unique data blocks physically inside the cluster after it calculates their fingerprint IDs. Therefore, it never pre-allocates or thick-provisions storage space before writing.

As a result of XtremIO's content-aware architecture, blocks can be stored at any location in the system (and only metadata is used to refer to their locations) and the data is written only when unique blocks are received.

Therefore, unlike thin provisioning with many disk-oriented architectures, with XtremIO there is no space creeping and no garbage collection. Furthermore, the issue of volume fragmentation over time is not applicable to XtremIO (as the blocks are scattered all over the random-access array) and no defragmentation utilities are needed.

XtremIO's inherent thin provisioning also enables consistent performance and data management across the entire life cycle of the volumes, regardless of the system capacity utilization or the write patterns to the system.

Inline Data Reduction

XtremIO's unique Inline Data Reduction is achieved by utilizing the following techniques:

- Inline Data Deduplication
- Inline Data Compression

Inline Data Deduplication

Inline data deduplication is the removal of redundancies from data **before** it is written to the flash media.

XtremIO automatically and globally deduplicates data as it enters the system. Deduplication is performed in real-time and not as a post-processing operation. With XtremIO, there are no resource-consuming background processes and no additional reads/writes (which are associated with post-processing). Therefore, it does not negatively affect performance of the storage array, does not waste the available resources that are allocated for the host I/O, and does not consume flash wear cycles.

With XtremIO, data blocks are stored according to their content, and not according to their user level address within the volumes. This ensures perfect load balancing across all devices in the system in terms of capacity and performance. Each time a data block is modified, it can be placed on any set of SSDs in the system, or not written at all if the block's content is already known to the system.

The system inherently spreads the data across the array, using all SSDs evenly and providing perfect wear leveling. Even if the same logical block address (LBA) is repeatedly written by a host computer, each write is directed to a different location within the XtremIO array. If the host writes the same data over and over again, it will be deduplicated, resulting in no additional writes to the flash.

XtremIO uses a content-aware, globally deduplicated cache for highly efficient data deduplication. The system's unique content-aware storage architecture enables achieving a substantially larger cache size with a small DRAM allocation. Therefore, XtremIO is the ideal solution for difficult data access patterns, such as the "boot storms" that are common in virtual desktop (VDI) environments.

The system also uses the content fingerprints, not only for Inline Data Deduplication, but also for uniform distribution of data blocks across the array. This provides inherent load balancing for performance and enhances flash wear level efficiency, since the data never needs to be rewritten or rebalanced.

Performing this process inline, and globally across the array, translates into fewer writes to the SSDs. This increases SSD endurance and eliminates performance degradation that is associated with post-processing deduplication.

XtremIO's Inline Data Deduplication and its intelligent data storage process ensure:

- Balanced usage of the system resources, maximizing the system performance
- Minimum amount of flash operations, maximizing the flash longevity
- Equal data distribution, resulting in evenly balanced flash wear across the system
- No system level garbage collection (as opposed to post-processing data reduction)
- Smart usage of SSD capacity, minimizing storage costs

Inline Data Compression

Inline Data Compression is the compression of the already deduplicated data **before** it is written to the flash media.

XtremIO automatically compresses data after all duplications have been removed. This ensures that the compression is performed only for unique data blocks. Data compression is performed in real-time and not as a post-processing operation.

The nature of the data set determines the overall compressibility rate. The compressed data block is then stored on the array.

Compression reduces the total amount of physical data that needs to be written on the SSD. This reduction minimizes the Write Amplification (WA) of the SSDs, thereby improving the endurance of the flash array.

XtremIO's Inline Data Compression provides the following benefits:

- Data compression is always inline and is never performed as a post-processing activity. Therefore, the data is always written only once.
- Compression is supported for a diverse variety of data sets (e.g. database data, VDI, VSI environments, etc.).
- Data compression complements data deduplication in many cases. For example, in a VDI environment deduplication dramatically reduces the required capacity for cloned desktops. Consequently, compression reduces the specific user data. As a result, an increased number of VDI desktops can be managed by a single X-Brick.
- Compression saves storage capacity by storing data blocks in the most efficient manner.
- When combined with XtremIO's powerful snapshot capabilities, XtremIO can easily support petabytes of functional application data.

Total Data Reduction

XtremIO's data deduplication and data compression complement each other. Data deduplication reduces physical data, by eliminating redundant data blocks. Data compression further reduces the data footprint, by eliminating data redundancy within the binary level of each data block.

Figure 10 demonstrates the benefits of both the data deduplication and data compression processes combined, resulting in total data reduction.

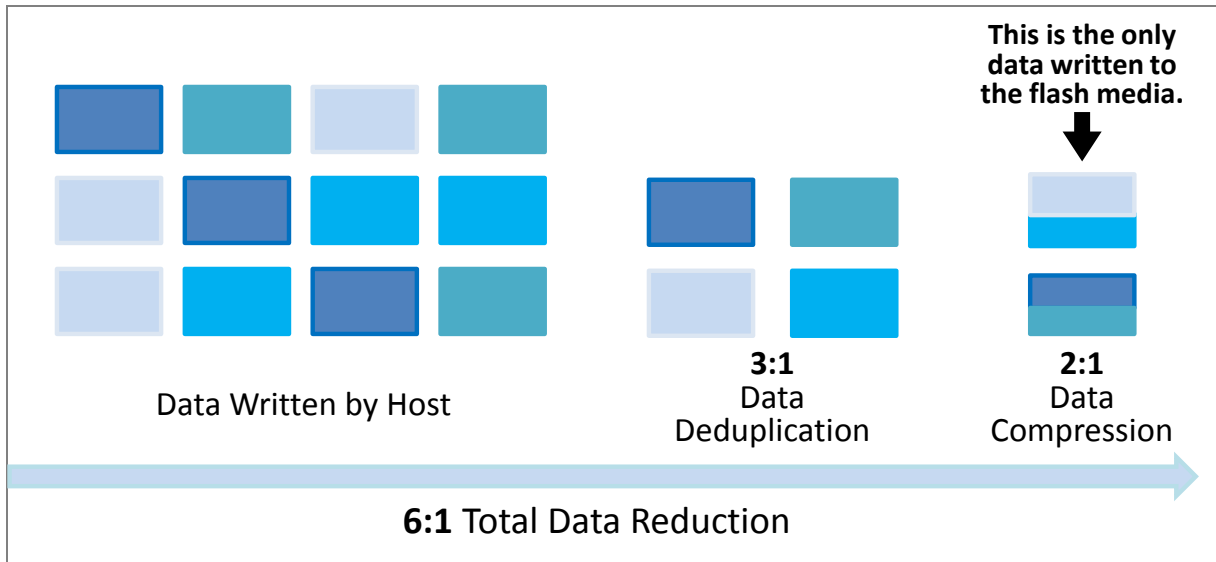


Figure 10. Data Deduplication and Compression Combined

In the above example, the twelve data blocks written by the host are first deduplicated to four data blocks, demonstrating a 3:1 data deduplication ratio. Following the data compression process, the four data blocks are then each compressed, by a ratio of 2:1, resulting in a total data reduction ratio of 6:1.

XtremIO Data Protection (XDP)

The XtremIO storage system provides "self-healing" double-parity data protection with a very high efficiency.*

The system requires very little capacity overhead for data protection and metadata space. It does not require dedicated spare drives for rebuilds. Instead, it leverages the "hot space" concept, where any free space available in the array can be utilized for failed drive reconstructions. The system always reserves sufficient distributed capacity for performing a single rebuild.

XtremIO maintains its performance, even at high capacity utilization, with minimal capacity overhead. The system does not require mirroring schemes (and their associated 100% capacity overhead).

XtremIO requires far less reserved capacity for data protection, metadata storage, snapshots, spare drives and performance, leaving much more space for user data. This lowers the cost per usable GB.

The XtremIO storage system provides:

- N+1 data protection
- Incredibly low data protection capacity overhead of 8%
- Performance superior to any RAID algorithm (RAID 1, the RAID algorithm that is most efficient for writes, requires over 60% more writes than XtremIO Data Protection.)
- Flash endurance superior to any RAID algorithm, due to smaller amount of writes and even distribution of data
- Automatic rebuild in case of drive failure and faster rebuild times than traditional RAID algorithms
- Superior robustness with adaptive algorithms that fully protect incoming data, even when failed drives exist in the system
- Administrative ease through fail-in-place support

* The current software version supports a single drive rebuild at a time. Dual concurrent rebuilds will be added in the next point release.

Table 4. Comparison of XtremIO Data Protection against RAID Schemes

Algorithm	Performance	Data Protection	Capacity Overhead	Reads per Stripe Update	Traditional Algorithm Read Disadvantage	Writes per Stripe Update	Traditional Algorithm Write Disadvantage
RAID 1	High	1 failure	50%	0	–	2 (64%)	1.6x
RAID 5	Medium	1 failure	25% (3+1)	2 (64%)	1.6x	2 (64%)	1.6x
RAID 6	Low	2 failures	20% (8+2)	3 (146%)	2.4x	3 (146%)	2.4x
XtremIO XDP	60% better than RAID 1	1 failure per X-Brick	Ultra Low 8% (23+2)	1.22	–	1.22	–

How XDP Works

XtremIO Data Protection (XDP) is designed to take advantages of flash media specific properties and XtremIO's content addressable storage architecture.

Benefiting from the fact that it can control where data is stored without any penalty, XDP achieves high protection levels and low storage overhead, but with better performance than RAID 1. As an additional benefit, XtremIO Data Protection also significantly enhances the endurance of the underlying flash media, compared to any previous RAID algorithm, which is an important consideration for an enterprise flash array.

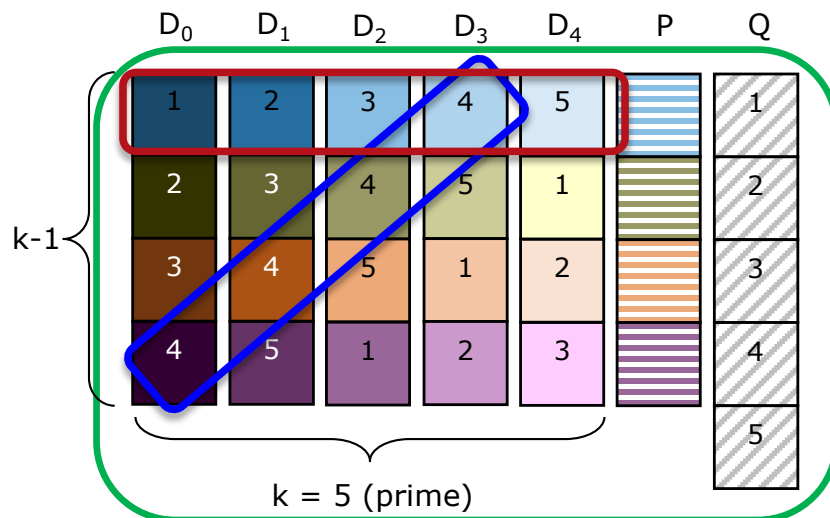


Figure 11. Row and Diagonal Parity

XDP uses a variation of N+2 row and diagonal parity, as shown in [Figure 11](#), which provides protection from two simultaneous SSD errors. With arrays of 25 SSDs this results in 8% of capacity overhead.

Traditional arrays update logical block addresses (LBAs) in the same physical location on the disk (causing the high I/O overhead of a stripe update). XtremIO always places the data in the emptiest stripe. Writing data to the emptiest stripe effectively amortizes the overhead of read and write I/O operations for every stripe update and is only feasible in XtremIO's all-flash, content-based architecture. This process ensures that XtremIO performs consistently as the array fills and is in service for extended periods of time when overwrites and partial stripe updates become the norm.

XtremIO also provides a superior rebuild process. When a traditional RAID 6 array faces a single disk failure, it uses RAID 5 methods to rebuild it by reading each stripe and computing the missing cell from the other cells in the stripe. In contrast, XtremIO uses both the P and Q parity to rebuild the missing information and uses an elaborated algorithm that reads only the needed information for the next cell rebuild.

Table 5. Comparison of XDP Reads for Rebuilding a Failed Disk with those of Different RAID Schemes

Algorithm	Reads to Rebuild a Failed Disk Stripe of Width K	Traditional Algorithm Disadvantage
XtremIO XDP	$3K/4$	–
RAID 1	1	None
RAID 5	K	33%
RAID 6	K	33%

Note:

For more detailed information on XDP, refer to the XtremIO Data Protection White Paper.

Data at Rest Encryption

Data at Rest Encryption (DARE) provides a solution to securing critical data even when the media is removed from the array. XtremIO arrays utilize a high performance inline encryption technique to ensure that all data stored on the array is unusable if the SSD media is removed. This prevents unauthorized access in the event of theft or loss during transport, and makes it possible to return/replace failed components containing sensitive data.

DARE is a mandatory requirement that has been established in several industries, such as health care (where patient records must be kept closely-guarded), banking (where financial data safety is extremely important), and in many government institutions.

At the heart of XtremIO's DARE solution lies the use of Self-Encrypting Drive (SED) technology. An SED has dedicated hardware, which is used to encrypt and decrypt data as it is written to or read from the SSD. Offloading the encryption task to the SSD enables XtremIO to maintain the same software architecture whenever encryption is enabled or disabled on the array. All of XtremIO's features and services, including Inline Data Reduction, XtremIO Data Protection (XDP), thin provisioning and snapshots are available on an encrypted cluster (as well as on non-encrypted clusters).

A unique Data Encryption Key (DEK) is created during the drive manufacturing process. The key does not leave the drive at any time. It is possible to erase the DEK or change it, but this causes the data on the drive to become unreadable and no option is provided to retrieve the DEK. In order to ensure that only authorized hosts can access the data on the SED, the DEK is protected by an Authentication Key (AK). Without this key the DEK is encrypted and cannot be used to encrypt or decrypt data.

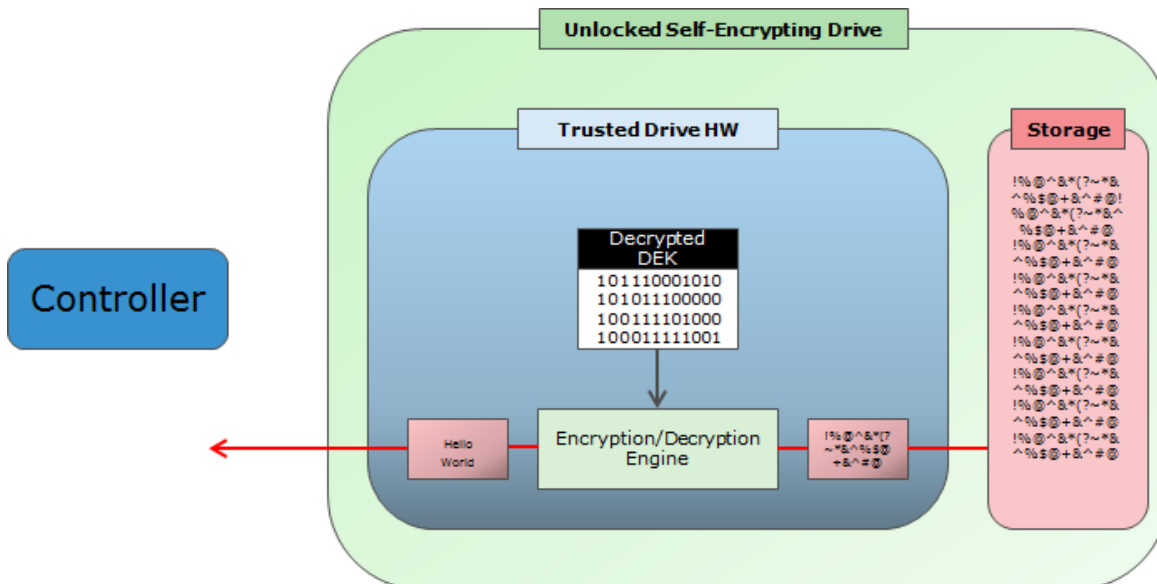


Figure 12. Unlocked SED

SEDs are shipped out of the factory in an unlocked state, meaning that any host can access the drive data. In unlocked drives the data is always encrypted, but the DEK is always decrypted and no authentication is required.

Locking the drive is made possible by changing the default drive's AK to a new, private AK and changing the SED settings so that it remains locked after a boot or power fail (such as when an SSD is taken out of the array). When an SSD is taken out of the array, it is turned off and will require the AK upon booting up. Without the correct AK the data on the SSD is unreadable and safe.

To access the data the hosts must provide the correct AK, a term that is sometimes referred to as "acquiring" or "taking ownership of " the drive, which unlocks the DEK and enables data access.

Drive acquisition is achieved only upon boot, and the SED remains unlocked for as long as the array is up. Since data passes through the encryption or decryption hardware in all cases, there is no performance impact when locking an SED.

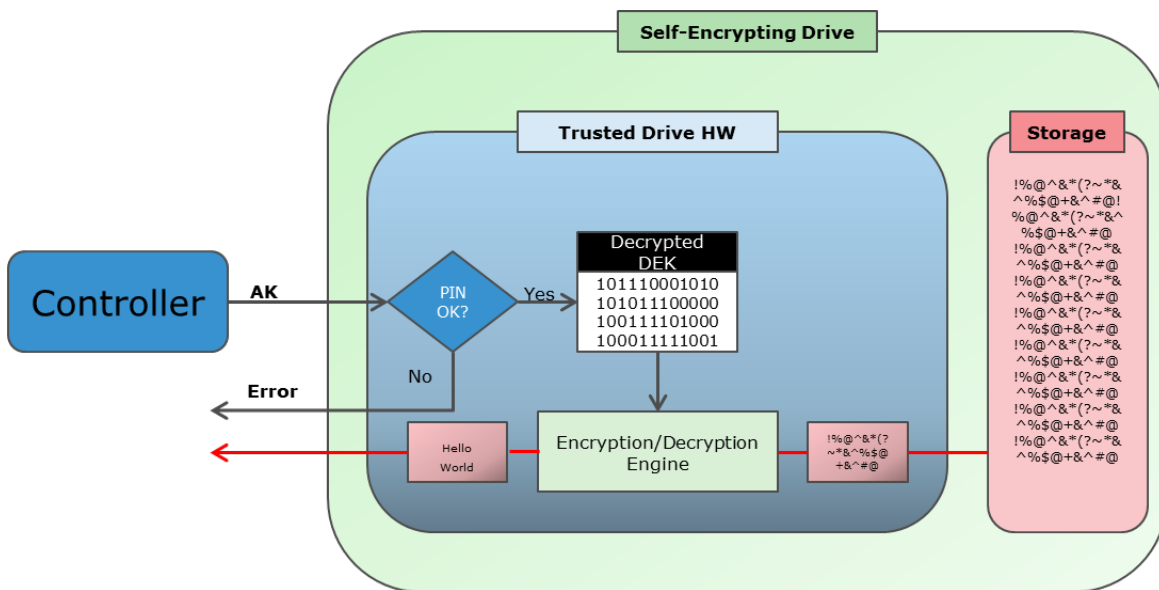


Figure 13. SED Operation Mode

The XtremIO All-Flash Array encrypts data on the following SSDs:

- Data SSDs – where all User Data is stored
- Storage Controller SSDs – which may contain User Data journal dumps

Snapshots

Snapshots are created by capturing the state of data in volumes at a particular point in time and allowing users to access that data when needed, even when the source volume has changed. XtremIO snapshots are inherently writeable, but may be mounted read-only to maintain immutability. Snapshots can be taken from either the source or any snapshot of the source volume.

Snapshots can be used in a number of use cases, including:

- Logical corruption protection

XtremIO allows creating frequent snapshots (based on the desired RPO intervals) and using them to recover from any logical data corruption. The snapshots can be kept in the system as long as they are needed. If a logical data corruption occurs, snapshots of an earlier application state (prior to logical data corruption) can be used to recover the application to a known good point in time.

- Backup

It is possible to create snapshots to be presented to a backup server/agent. This can be used in order to offload the backup process from the production server.

- Development and Test

The system enables the user to create snapshots of the production data, create multiple (space-efficient and high-performance) copies of the production system and present them for development and testing purposes.

- Clones

With XtremIO, it is possible achieve clone-like capabilities, by using persistent writable snapshots. They can be used in order to present a clone of the production volume to multiple servers. Performance of the clone will be identical to that of the production volume.

- Offline processing

Snapshots can be used as a means to offload the processing of data from the production server. For example, if it is needed to run a heavy process on the data (which can affect the production server's performance), it is possible use snapshots to create a recent copy of the production data and mount it on a different server. The process can then be run (on the other server), without consuming the production server's resources.

XtremIO's snapshot technology is implemented by leveraging the content-aware capabilities of the system (Inline Data Reduction), optimized for SSD media, with a unique metadata tree structure that directs I/O to the right timestamp of the data. This allows efficient snapshotting that can sustain high performance, while maximizing the media endurance, both in terms of the ability to create multiple snapshots and the amount of I/O that a snapshot can support.

When creating a snapshot, the system generates a pointer to the ancestor metadata (of the actual data in the system). Therefore, creating a snapshot is a very quick operation that does not have any impact on the system and does not consume any capacity. Snapshot capacity consumption occurs only if a change requires writing a new unique block.

When a snapshot is created, its metadata is identical to that of the ancestor volume. When a new block is written to the ancestor, the system updates the metadata of the ancestor volume to reflect the new write (and stores the block in the system, using the standard write flow process). As long as this block is shared between the snapshots and the ancestor volume, it will not be deleted from the system following a write. This applies both to a write in a new location on the volume (a write on an unused LBA) and to a rewrite on an already written location.

The system manages the snapshot's and ancestor's metadata, using a tree structure. The snapshot and the ancestor volumes are represented as leaves in this structure, as shown in Figure 14.

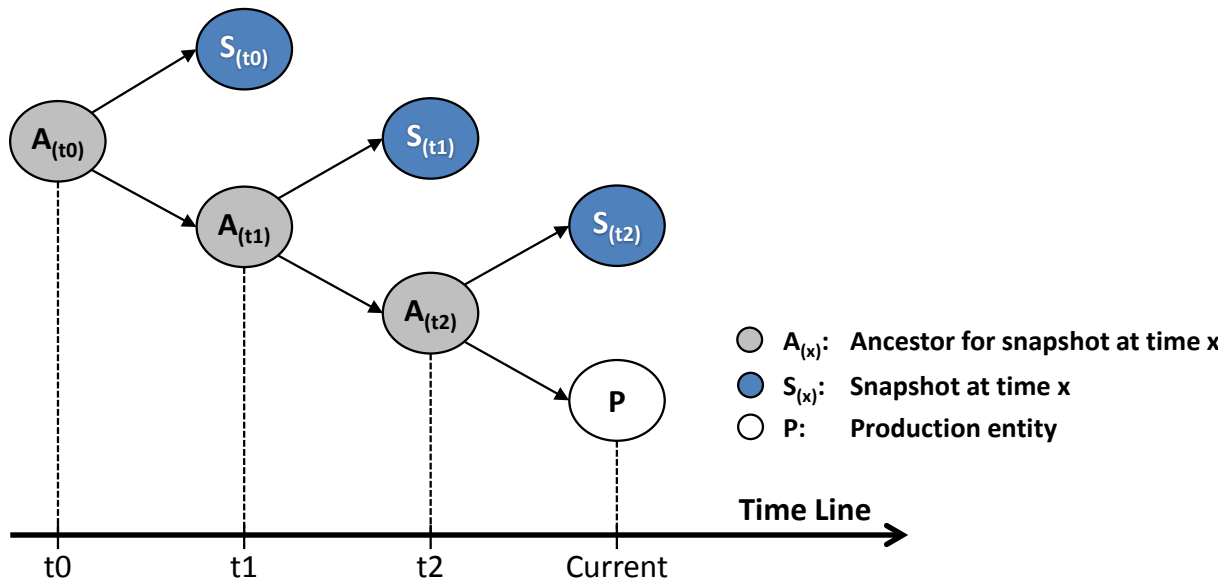


Figure 14. Metadata Tree Structure

The metadata is shared between all snapshot blocks that have not been changed (from the snapshot's original ancestor). The snapshot maintains unique metadata only for an LBA whose data block is different from that of its ancestor. This provides economical metadata management.

When a new snapshot is created, the system always creates two leaves (two descendant entities) from the snapshotted entity. One of the leaves represents the snapshot, and the other one becomes the source entity. The snapshotted entity will no longer be used directly, but will be kept in the system for metadata management purposes only.

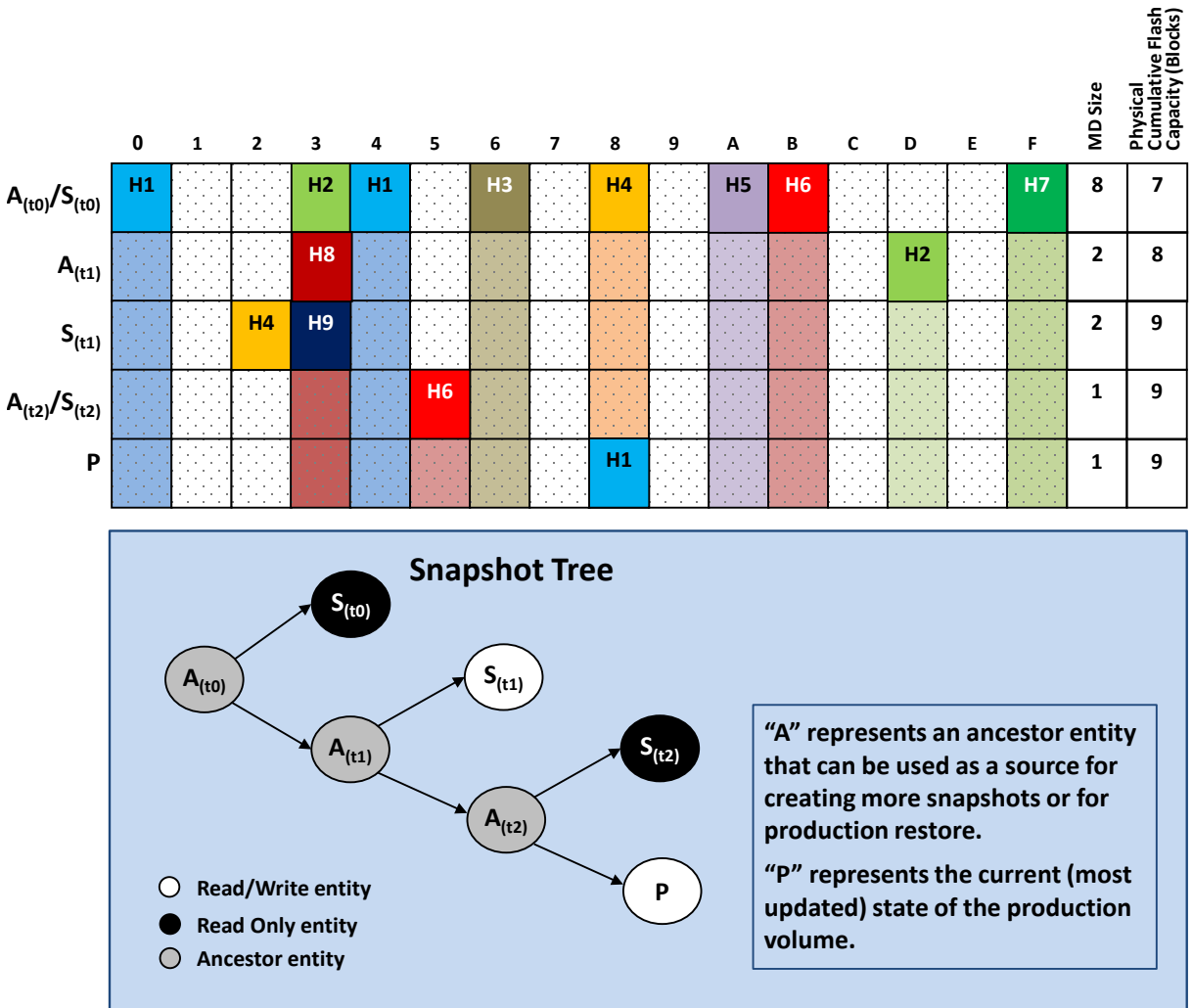


Figure 15. Creating Snapshots

Figure 15 illustrates a 16-block volume in the XtremIO system. The first row (marked as $A_{(t_0)}/S_{(t_0)}$) shows the volume at the time the first snapshot was taken (t_0). At t_0 , the ancestor ($A_{(t_0)}$) and the snapshot ($S_{(t_0)}$) have the same data and metadata, because $S_{(t_0)}$ is the read-only snapshot of $A_{(t_0)}$ (containing the same data as its ancestor).

Note:

Out of the 16 blocks, only 8 blocks are used. Blocks 0 and 4 consume only one block of physical capacity as a result of deduplication. The blanked dotted blocks represent the blocks that are thinly provisioned and do not consume any physical capacity.

In Figure 15, before creating the snapshot at $S_{(t_1)}$, two new blocks are written to P:

- H8 overwrites H2.
- H2 is written to block D. But it does not take up more physical capacity because it is the same as H2, stored in block 3 in $A_{(t_0)}$.

$S_{(t_1)}$ is a read/write snapshot. It contains two additional blocks (2 and 3) that are different from its ancestor.

Unlike traditional snapshots (which need reserved spaces for changed blocks and an entire copy of the metadata for each snap), XtremIO does not need any reserved space for snaps and never has metadata bloat.

At any time, an XtremIO snapshot consumes only the unique metadata, which is used only for the blocks that are not shared with the snapshot's ancestor entities. This allows the system to efficiently maintain large numbers of snapshots, using a very small storage overhead which is dynamic and proportional to the amount of changes in the entities.

For example, at time t_2 , blocks 0, 3, 4, 6, 8, A, B, D and F are shared with the ancestor's entities. Only block 5 is unique for this snapshot. Therefore, XtremIO consumes only one metadata unit. The rest of the blocks are shared with the ancestors and use the ancestor data structure in order to compile the correct volume data and structure.

The system supports the creation of snapshot on a set of volumes. All of the snapshots from the volumes in the set are cross-consistent and contain the exact same-point-in-time for all volumes. This can be created manually, by selecting a set of volumes for snapshotting, or by placing volumes in a consistency group container and creating a snapshot of the consistency group.

During the snapshot creation there is no impact on the system performance or overall system latency (performance is maintained). This is regardless of the number of snapshots in the system or the size of the snapshot tree.

Snapshot deletions are lightweight and proportional only to the amount of changed blocks between the entities. The system uses its content-aware capabilities to handle snapshot deletions. Each data block has a counter that indicates the number of instances of that block in the system. When a block is deleted, the counter value is decreased by one. Any block whose counter value is zero (meaning that there is no logical block address [LBA] across all volumes or snapshots in the system that refers to this block) is overwritten by XDP when new unique data enters the system.

Deleting a child with no descendants requires no additional processing by the system.

Deleting a snapshot in the middle of the tree triggers an asynchronous process. This process merges the metadata of the deleted entity's children with that of their grandparents. This ensures that the tree structure is not fragmented.

With XtremIO, every block that needs to be deleted is immediately marked as freed. Therefore, there is no garbage collection and the system does not have to perform a scanning process to locate and delete the orphan blocks. Furthermore, with XtremIO, snapshot deletions have no impact on system performance and SSD media endurance.

The snapshot implementation is entirely metadata driven and leverages the array's Inline Data Reduction to ensure that data is never copied within the array. Thus many snapshots can be maintained.

XtremIO's snapshots:

- Require no reserved snapshot space.
- Allow for the creation of immutable copies and/or writable clones of the source volume.
- Are created instantaneously.
- Have negligible performance impact on the source volume and the snapshot itself.

Note:

For more detailed information on snapshots, refer to the XtremIO Snapshots White Paper.

Scalable Performance

XtremIO is designed so as to scale out in order to meet future performance and capacity needs, not only for new applications, but also for those already deployed. XtremIO's architecture allows performance and capacity to be increased by adding building blocks (X-Bricks), while maintaining a single point of management and balance of resources across the system.

Scale out is an intrinsic part of the XtremIO's architecture and can be performed without a forklift upgrade of the existing hardware or any need for prolonged data transfers.

When additional performance or capacity is required, the XtremIO Storage System can be scaled-out by adding additional X-Bricks. Multiple X-Bricks are joined together over a redundant, high-availability, ultra-low latency InfiniBand network.

When the system expands, resources remain balanced, and data in the array is distributed across all X-Bricks to maintain consistent performance and equivalent flash wear levels.

System expansion is carried out without any need for configuration or manual movement of volumes.* XtremIO uses a consistent fingerprinting algorithm that minimizes remappings. A new X-Brick is added to the internal load balancing scheme and only the relevant existing data is transferred to the new DAE.

Storage capacity and performance scale linearly, such that two X-Bricks supply twice the IOPS, four X-Bricks supply four times the IOPS and six X-Bricks supply six times the IOPS of the single X-Brick configuration. However, the latency remains consistently low (less than 1ms) as the system scales out, as shown in [Figure 16](#).

* The current software version does not support dynamic scale out.

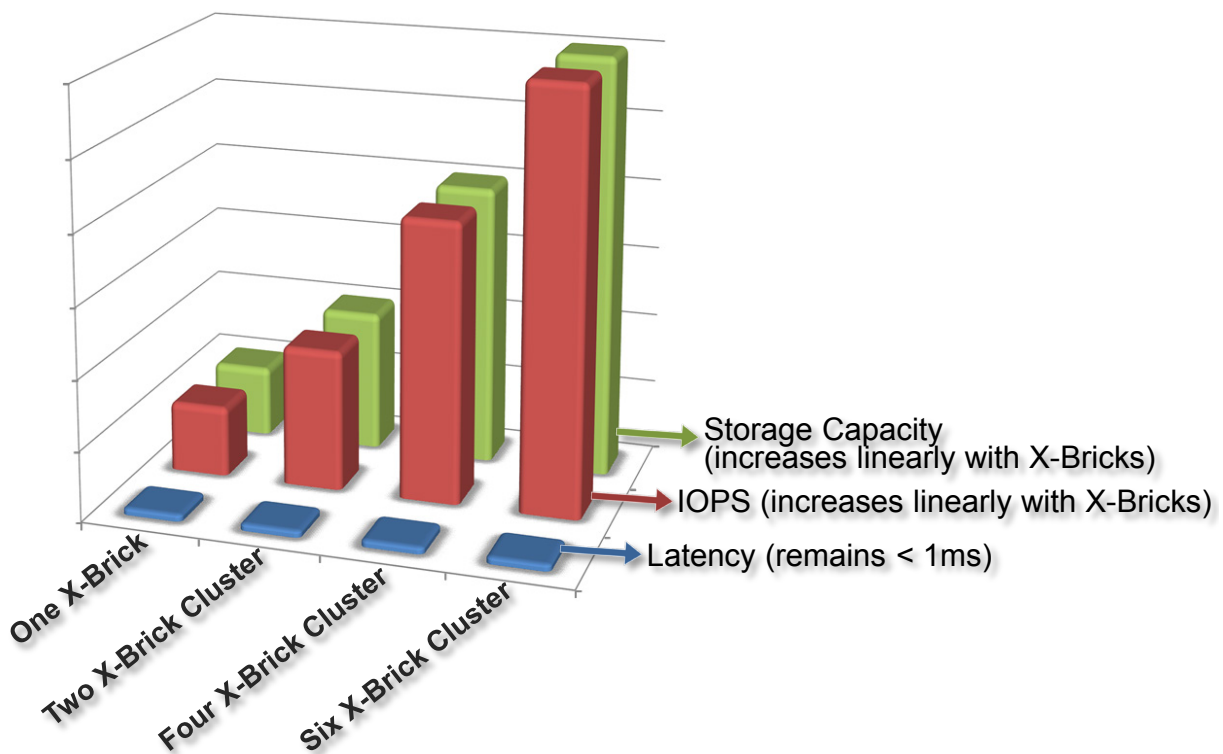


Figure 16. Linear Performance Scalability with Consistent Low Latency

Since XtremIO is specially developed for scalability, its software does not have an inherent limit to the size of the cluster.* The system architecture also deals with latency in the most effective way. The software design is modular. Every Storage Controller runs a combination of different modules and shares the total load. These distributed software modules (on different Storage Controllers) handle each individual I/O operation, which traverses the cluster. XtremIO handles each I/O request by two software modules (2 hops), no matter if it is a single X-Brick system or a multiple X-brick cluster. Therefore, the latency remains consistent at all times, regardless of the size of the cluster.

Note:

The sub-millisecond latency is validated by actual test results, and is determined according to the worst-case scenario.†

* The maximum cluster size is based on currently tested and supported configurations.

† Sub-millisecond latency applies to typical block sizes. Latency for small blocks or large blocks may be higher.

InfiniBand plays an important role in XtremIO's architecture. XtremIO uses two types of communication over the InfiniBand backplane: Remote Procedure Calls (RPC) for control messages and Remote Direct Memory Access (RDMA) for moving data blocks.

InfiniBand has not only one of the highest bandwidths available in any interconnect technology (40Gb/s for one QDR connection), but also has the lowest latency. The round-trip time for an RDMA transfer of a data block between two XtremIO Storage Controllers is about 7 microseconds, making it almost negligible compared to XtremIO's 500-microsecond latency allowance for each I/O. This enables the software to select any necessary Storage Controller and SSD resources, no matter if they are local or remote (over InfiniBand) to the Storage Controller that receives the I/O.

All XtremIO's enterprise features (including Inline Data Reduction, snapshots, XDP, HA, etc.) have been developed as part of the scale-out architecture. All data and metadata are evenly distributed across the entire cluster. I/Os are admitted to the array via all the host ports, utilizing SAN zones and multi-pathing. Therefore, since all the workload is evenly shared among the controllers and SSDs, it is virtually impossible for any performance bottlenecks to occur anywhere in the system.

With XtremIO:

- Processors, RAM, SSDs and connectivity ports scale together, providing scalable performance with perfect balance.
- The internal communication is carried out via a high-availability QDR (40Gb/s) InfiniBand internal fabric.
- The cluster is N-way active, enabling any volume to be reached from any host port on any storage controller on any X-Brick with equivalent performance.
- RDMA zero-copy data access makes I/Os to local or remote SSDs equivalent, regardless of the cluster size.
- Data is balanced across all X-Bricks as the system expands.
- There is a higher level of redundancy, and the cluster is more resilient to hardware and software failures. In an N-way Active scale-out cluster, if one Storage Controller fails, the system loses only 1/Nth of the total performance.
- The system is easy to upgrade and, unlike traditional dual-controller systems, XtremIO's scale-out model allows customers to start small, and grow both storage capacity and performance as the workload increases.

Even Data Distribution

To external applications, XtremIO appears and behaves like a standard block storage array. However, due to its unique architecture, it takes a fundamentally different approach to internal data organization. Instead of using logical addresses, XtremIO uses the block contents to decide where to place data blocks.

XtremIO uses data blocks internally. In a write operation, any data chunks that are larger than the native block size are broken down into standard blocks when they first enter the array. The system calculates a unique fingerprint for each of the incoming data blocks, using a special mathematical algorithm.

This unique ID is used for two primary purposes:

- To determine where the data block is placed within the array
- Inline Data Reduction (see page 21)

Because of the way the fingerprinting algorithm works, the ID numbers appear completely random and are evenly distributed over the possible range of fingerprint values. This results in an even distribution of data blocks across the entire cluster and all SSDs within the array. In other words, with XtremIO it is neither necessary to check the space utilization levels on different SSDs, nor to actively manage equal data writes to every SSD. XtremIO inherently provides even distribution of data by placing the blocks based on their unique IDs (see [Figure 7](#) on page 17).

XtremIO maintains the following metadata:

- Logical address (LBA) to fingerprint ID mapping
- Fingerprint ID to physical location mapping
- Reference count on each fingerprint ID

The system keeps all metadata in the Storage Controllers' memory and protects them by mirroring the change journals among different Storage Controllers, via RDMA. It also saves them periodically to SSD.

Keeping all metadata in the memory enables XtremIO to provide the following unique benefits:

- No SSD lookups
By avoiding SSD lookups, more I/Os are available to hosts' operations.
- Instant Snapshots
Snapshot operations are instantaneous, as the process of taking a snap is carried out entirely in the array's memory (see page 30).
- Instant VM cloning
Inline Data Reduction and VAAI, combined with in-memory metadata, enable XtremIO to clone a VM by memory operations only.
- Steady performance
Physical locations of data, large volumes and wide LBA ranges have no effect on the system performance.

High Availability

Preventing data loss and maintaining service in case of multiple failures, is one of the core features in the architecture of XtremIO's All Flash Storage Array.

From the hardware perspective, no component is a single point of failure. Each Storage Controller, DAE and InfiniBand Switch in the system is equipped with dual power supplies. The system also has dual Battery Backup Units and dual network and data ports (in each of the Storage Controllers). The two InfiniBand Switches are cross connected and create a dual data fabric. Both the power input and the different data paths are constantly monitored and any failure triggers a recovery attempt or failover.

The software architecture is built in a similar way. Every piece of information that is not committed to the SSD is kept in multiple locations, called Journals. Each software module has its own Journal, which is not kept on the same Storage Controller, and can be used to restore data in case of unexpected failure. Journals are regarded as highly important and are always kept on Storage Controllers with battery backed up power supplies. In case of a problem with the Battery Backup Unit, the Journal fails over to another Storage Controller. In case of global power failure, the Battery Backup Units ensure that all Journals are written to vault drives in the Storage Controllers and the system is turned off.

In addition, due to its scale-out design and the XDP data protection algorithm, each X-Brick is preconfigured as a single redundancy group. This eliminates the need to select, configure and tune redundancy groups.

XtremIO's Active-Active architecture is designed to ensure maximum performance and consistent latency. The system includes a self-healing mechanism that attempts to recover from any failure and resume full functionality. An attempt to restart a failed component is performed once before a failover action. Storage Controller failover is carried out as the last resort. Based on the nature of the failure, the system attempts to failover the relevant software component, while maintaining the operation of other components, thus minimizing the performance impact. The whole Storage Controller fails over only if recovery attempts are not successful or if the system must act in the best interest of protecting against data loss.

When a component that was temporarily unavailable recovers, a failback is initiated. This process is carried out at the software component or Storage Controller level. An anti-bounce mechanism prevents the system from failing back to an unstable component or to a component that is under maintenance.

Built on commodity hardware, XtremIO does not rely solely on hardware-based error detection and includes a proprietary algorithm that ensures detection, correction and marking of corrupted areas. Any data corruption scenario that is not automatically handled by the SSD hardware is addressed by the XDP mechanism on the array or the multiple copies that are held in the Journals. The content fingerprint is used as a secure and reliable data integrity mechanism during read operations to avoid silent data corruption errors. If there is a mismatch in the expected fingerprint, the array will recover the data, either by reading it again or by reconstructing it from the XDP redundancy group.

Non-Disruptive Upgrade

During Non-Disruptive Upgrades (NDU) of the XtremIO Operating System, the system performs the upgrade procedure on a live cluster, updates all Storage Controllers in the cluster, and restarts the application in a process that takes less than 10 seconds. Since the underlying Linux Kernel is active throughout the upgrade process, the hosts do not detect any path disconnection during the application restart period.

In the rare case of a Linux kernel or firmware upgrade, it is possible to upgrade the XtremIO All Flash Array without any service interruption and without any risk of data loss. The NDU procedure is launched from the XtremIO Management Server and is able to upgrade the XtremIO software and the underlying operating system and firmware.

During Linux/firmware NDU, the system automatically fails over a component and upgrades its software. After completing the upgrade and verifying the component's health, the system fails back to it and the process repeats itself on other components. During the upgrade process the system is fully accessible, no data is lost, and the performance impact is kept to minimum.

VMware VAAI Integration

VAAI (vSphere Storage APIs for Array Integration) was introduced as an improvement to host-based VM cloning. Without VAAI, in order to clone a full VM, the host has to read each data block and write it to the new address where the cloned VM resides, as shown in Figure 17. This is a costly operation that loads the host, the array and the storage area network (SAN).

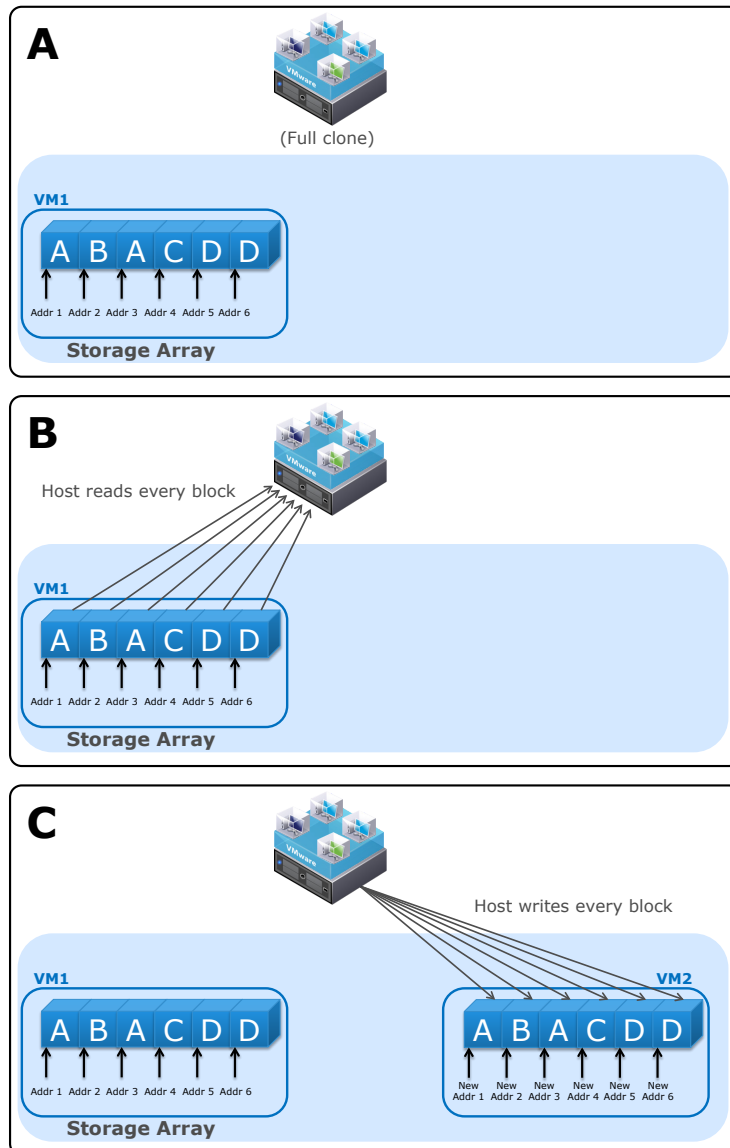


Figure 17. Full Copy without VAAI

With VAAI, the workload of cloning a VM is offloaded to the storage array. The host only needs to issue an X-copy command, and the array copies the data blocks to the new VM address, as shown in Figure 18. This process saves the resources of the host and the network. However, it still consumes the resources of the storage array.

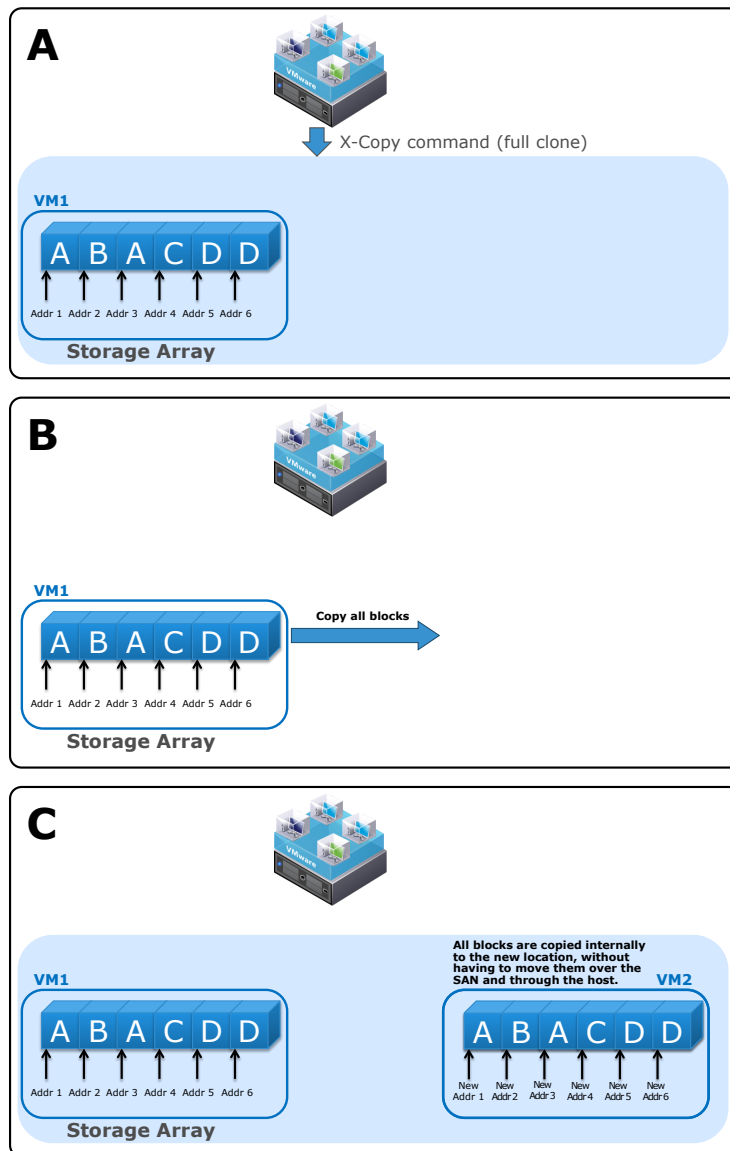


Figure 18. Full Copy with VAAI

XtremIO is fully VAAI compliant, allowing the array to communicate directly with vSphere and provide accelerated storage vMotion, VM provisioning, and thin provisioning functionality.

In addition, XtremIO's VAAI integration improves the X-copy efficiency even further, by making the whole operation metadata driven. With XtremIO, due to Inline Data Reduction and in-memory metadata, no actual data blocks are copied during the X-copy command. The system only creates new pointers to the existing data, and the entire process is carried out in the Storage Controllers' memory, as shown in Figure 19. Therefore, it does not consume the resources of the storage array and has no impact on the system performance.

For example, a VM image can be cloned instantaneously (even multiple times) with XtremIO.

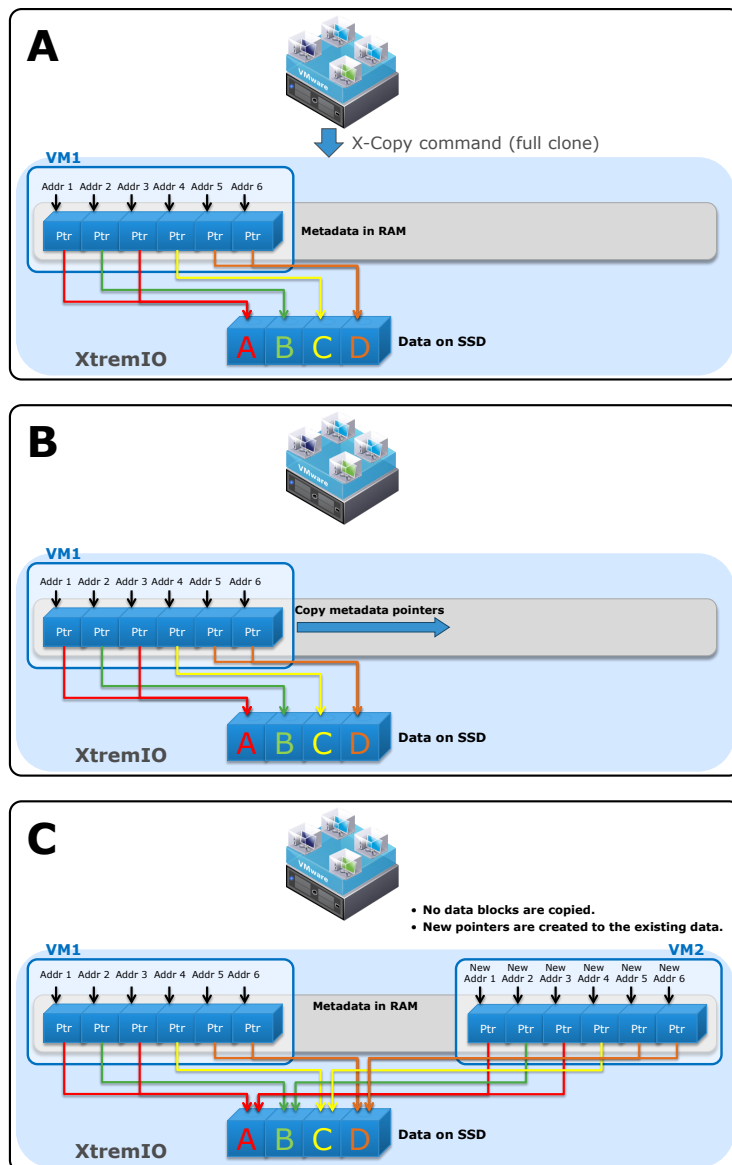


Figure 19. Full Copy with XtremIO

This is only possible with XtremIO's in-memory metadata and inline data reduction. Other flash products that implement VAAI but do not have inline deduplication still need to write the X-COPY to flash and deduplicate it later. Arrays that do not have in memory metadata need to carry out lookups on SSD to perform the X-COPY, which negatively impacts I/O to existing active VMs. Only with XtremIO is this process completed quickly, with no SSD writes, and with no impact to I/O on existing VMs.

The XtremIO features for VAAI support include:

- Zero Blocks/Write Same
Used for zeroing-out disk regions (VMware term: HardwareAcceleratedInit).
This feature provides accelerated volume formatting.
- Clone Blocks/Full Copy/XCOPY
Used for copying or migrating data within the same physical array (VMware term: HardwareAcceleratedMove).
On XtremIO, this allows VM cloning to take place almost instantaneously, without affecting user I/O on active VMs.
- Record based locking/Atomic Test & Set (ATS)
Used during creation and locking of files on a VMFS volume, for example, during powering-down/powering-up of VMs (VMware term: HardwareAcceleratedLocking).
This allows larger volumes and ESX clusters without contention.
- Block Delete/UNMAP/TRIM
Allows for unused space to be reclaimed, using the SCSI UNMAP feature (VMware term: BlockDelete; vSphere 5.x only).

XtremIO Management Server (XMS)

The XMS enables controlling and managing the system, including:

- Forming, initializing and formatting new systems
- Monitoring system health and events
- Monitoring system performance
- Maintaining a performance statistics history database
- Providing GUI and CLI services to clients
- Implementing volume management and data protection groups operation logic
- Maintaining (stopping, starting and restarting) the system

The XMS is preinstalled with the CLI and GUI. It can be installed on a dedicated physical server in the data center, or as a virtual machine on VMware.

The XMS must access all management ports on the X-Brick Storage Controllers, and must be accessible by any GUI/CLI client host machine. Since all communications use standard TCP/IP connections, the XMS can be located anywhere that satisfies the above connectivity requirements.

Since the XMS is not in the data path, it can be disconnected from the XtremIO cluster without affecting the I/O. An XMS failure only affects monitoring and configuration activities, such as creating and deleting volumes. However, when using a virtual XMS topology, it is possible to take advantage of VMware vSphere HA features to easily overcome such failures.

System GUI

Figure 20 illustrates the relationship between the system GUI and other network components.

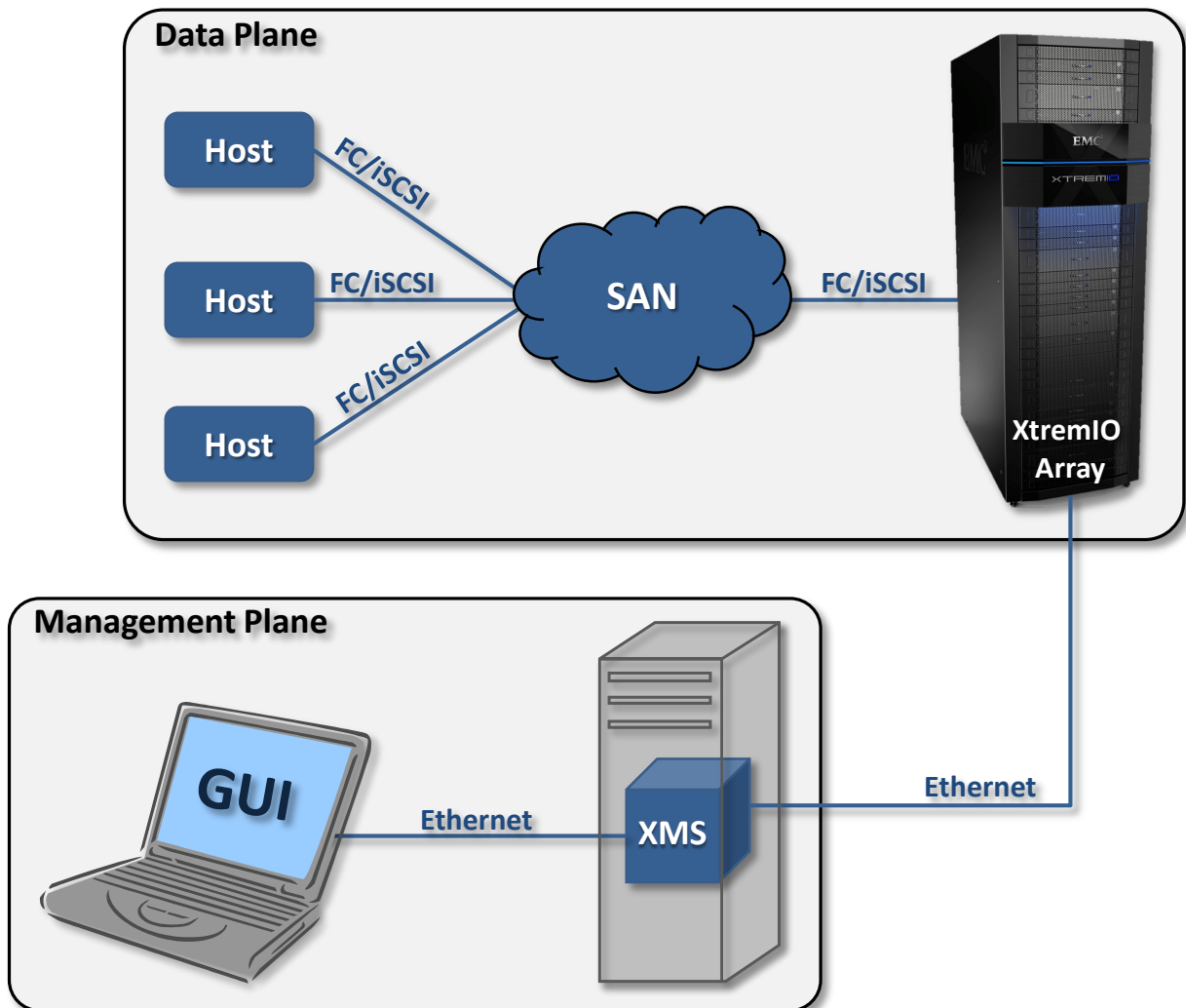


Figure 20. Relationship between GUI and other Network Components

The system GUI is implemented using a Java client. The GUI client software communicates with the XMS, using standard TCP/IP protocols, and can be used in any location that allows the client to access the XMS.

The GUI provides easy-to-use tools for performing most of the system operations (certain management operations must be performed using the CLI). Additionally, operations on multiple components, such as creating multiple volumes, can only be performed using the GUI.

Figure 21 shows how the GUI can be used to map Volumes to Initiator Groups in a few simple steps.

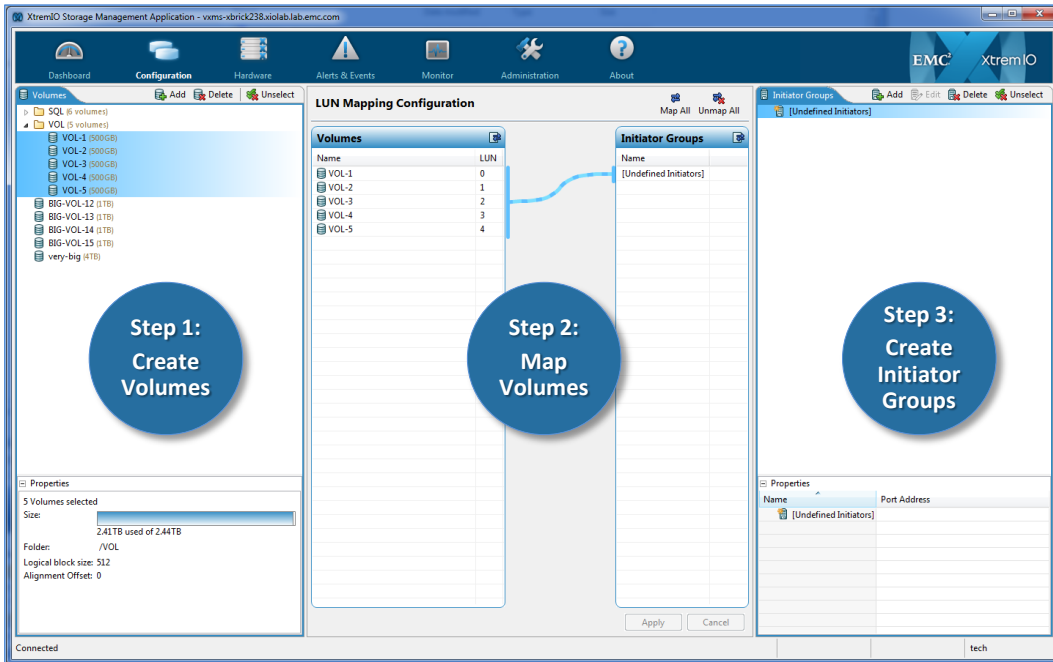


Figure 21. Mapping Volumes to Initiator Groups Using the GUI

Figure 22 shows the GUI's Dashboard, enabling the user to monitor the system's storage, performance, alerts, and hardware status.

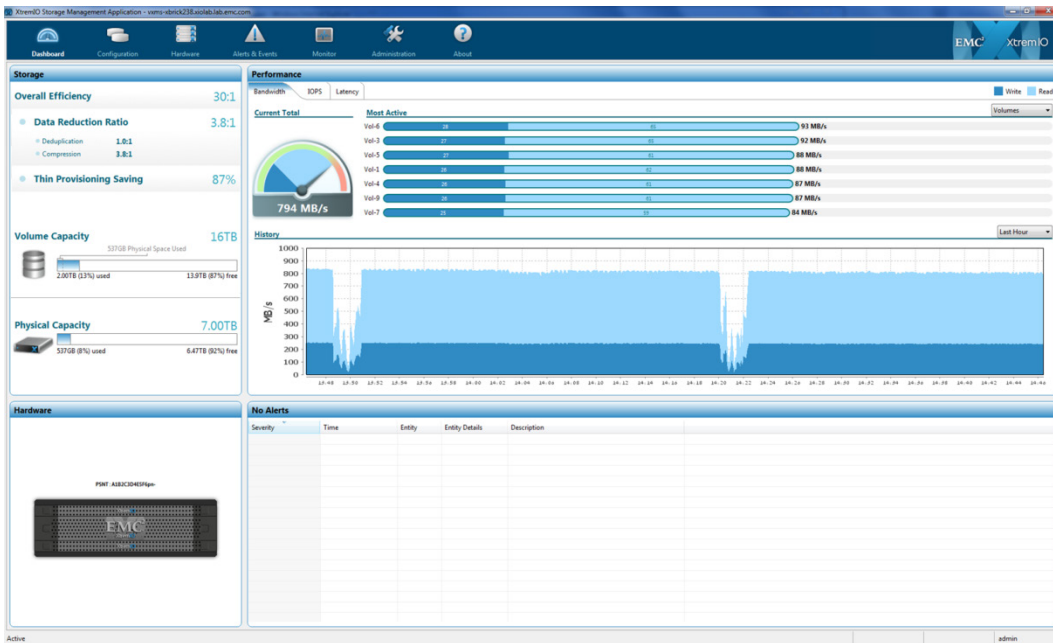


Figure 22. Monitoring the System Using the GUI

Command Line Interface

The system's Command Line Interface (CLI) allows administrators and other system users to perform supported management operations. It is preinstalled on the XMS and can be accessed using the standard SSH protocol.

A CLI client package, which communicates with the XMS server via standard TCP/IP connections and can be installed on a Linux CentOS host with access to the XMS, is also available.

RESTful API

The XtremIO's RESTful API allows HTTP-based interface for automation, orchestration, query and provisioning of the system. With the API, third party applications can be used to control and fully administer the array. Therefore, it allows flexible management solutions to be developed for the XtremIO array.

LDAP/LDAPS

The XtremIO Storage Array supports LDAP users' authentication both for CLI and GUI users. Once configured for LDAP authentication, the XMS redirects users' authentication to the configured LDAP or Active Directory (AD) servers and allows access to authenticated users only. Users' XMS permissions are defined, based on a mapping between the users' LDAP/AD groups and XMS roles.

The XMS Server LDAP Configuration feature allows using single or multiple servers to authenticate the external users' for their login to the XMS server.

The LDAP operation is performed once when logging with external user credentials to an XMS server. The XMS server operates as an LDAP client and connects to an LDAP service, running on an external server. The LDAP Search is performed, using the pre-configured LDAP Configuration profile and the external user login credentials.

If the authentication is successful, the external user logs in to the XMS server and accesses the full or limited XMS server functionality (according to the XMS Role that was assigned to the LDAP user's group).

The XtremIO Storage Array also supports LDAPS for secure authentication.

Ease of Management

XtremIO is very simple to configure and manage and there is no need for tuning or extensive planning.

With XtremIO the user does not need to choose between different RAID options in order to optimize the system. Once the system is initialized, the XDP (see page 25) is already configured as a single redundancy group. All the user data is spread across all the X-Bricks. Also, there is no tiering and performance tuning. All I/Os are treated the same. All volumes, when created, are mapped to all ports (FC and iSCSI) and there is no storage tiering in the array. This eliminates the need for manual performance tuning and optimization settings, and makes the system easy to manage, configure and use.

XtremIO provides:

- Minimum planning
 - No RAID configuration
 - Minimal sizing effort for cloning/snapshots
- No tiering
 - Single tier, all-flash array
- No performance tuning
 - Independent of I/O access pattern, cache hit rates, tiering decisions, etc.

Integration with other EMC Products

XtremIO is well integrated with other EMC products. Integration points will continue to expand in subsequent XtremIO releases to offer additional value for EMC customers.

Powerpath

EMC Powerpath is a host-based software that provides automated data path management and load balancing capabilities for heterogeneous servers, network and storage, deployed in physical and virtual environments. It enables users to meet service levels with high application availability and performance. Powerpath automates path failover and recovery for high availability in case of error or failure, and optimizes performance by load balancing I/Os across multiple paths. XtremIO is supported under Powerpath both directly and by virtualizing the XtremIO system using VPLEX.

VPLEX

The EMC VPLEX family is the next-generation solution for data mobility and access within, across and between data centers. The platform enables local and distributed federation.

- Local federation provides transparent cooperation of physical elements within a site.
- Distributed federation extends access between two locations across distance.

VPLEX removes physical barriers and enables users to access a cache-coherent, consistent copy of data at different geographical locations, and to geographically stretch virtual or physical host clusters. This enables transparent load sharing between multiple sites while providing the flexibility of relocating workloads between sites in anticipation of planned events. Furthermore, in case of an unplanned event that could cause disruption at one of the data centers, failed services can be restated at the surviving site.

VPLEX supports two configurations, local and metro. In the case of VPLEX Metro with the optional VPLEX Witness and Cross-Connected configuration, applications continue to operate in the surviving site with no interruption or downtime. Storage resources virtualized by VPLEX cooperate through the stack, with the ability to dynamically move applications and data across geographies and service providers.

XtremIO can be used as a high performing pool within a VPLEX Local or Metro cluster. When used in conjunction with VPLEX, XtremIO benefits from all of the VPLEX data services, including host operating system support, data mobility, data protection, replication, and workload relocation.

RecoverPoint

The EMC RecoverPoint family provides cost-effective, local continuous data protection (CDP), continuous remote replication (CRR) and continuous local and remote replication (CLR) solutions that allow for any-point-in-time data recovery.

RecoverPoint/EX supports local and remote replication for EMC Symmetrix® VMAX™ 10K, Symmetrix VMAX 20K, Symmetrix VMAX 40K, VPLEX™, XtremIO (when virtualized by VPLEX, with native RecoverPoint support planned for a subsequent release), VNX Series and Clariion CX3 or CX4 array.

The product enables the customer to centralize and simplify their data protection management and provide local and continuous data protection and/or remote replication:

- Block level remote or local replication
- Dynamic synchronous, synchronous or asynchronous remote replication
- Policy-based replication to enable optimizing storage and network resources, while obtaining desired RPO and RTO
- Application-aware integration
- Support for geo-cluster on windows environment (using RecoverPoint/CE)

RecoverPoint for VMs is a fully virtualized hypervisor-based replication solution and is built on a fully virtualized EMC RecoverPoint engine.

RecoverPoint for VMs:

- Optimizes RPO/RTO for VMware environment at lower TCO.
- Streamlines OR & DR and Increases business agility.
- Equips IT or service providers for cloud-ready data protection to deliver disaster as a service for private, public and Hybrid clouds.

Solutions Brief

PowerPath, VPLEX, RecoverPoint and XtremIO can be integrated* together to offer a strong, robust, and high performing block storage solution.

- Powerpath – Is installed on hosts to provide path failover, load balancing and performance optimization VPLEX engines (or directly to the XtremIO array if VPLEX is not used).
- VPLEX Metro – Allows sharing storage services across distributed virtual volumes and enables simultaneous read and write access across metro sites and across array boundaries.
- VPLEX Local – Used at the target site, virtualizes both EMC and non-EMC storage devices, leading to better asset utilization.
- RecoverPoint/EX – Any device encapsulated by VPLEX (including XtremIO) can use the RecoverPoint services for asynchronous, synchronous or dynamic synchronous data replication.

For example:

An organization has three data centers at New Jersey, New York City, and Iowa, as shown in Figure 23.

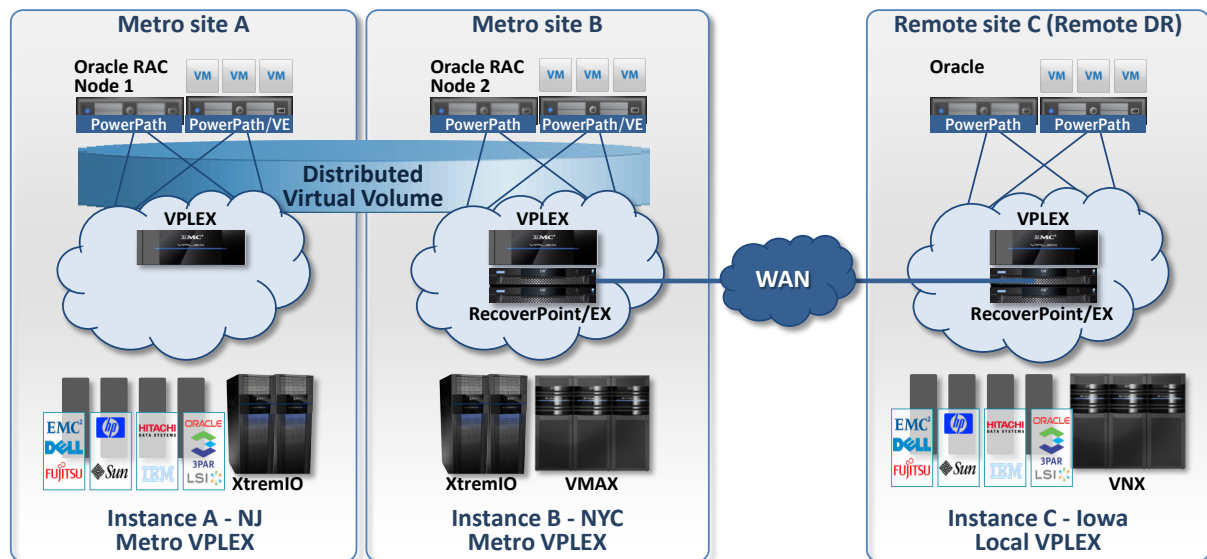


Figure 23. Integrated Solution, Using XtremIO, PowerPath, VPLEX and RecoverPoint

* RPQ approval is required. Please contact your EMC representative.

Oracle RAC and VMware HA nodes are dispersed between the NJ and NYC sites and data is moved frequently between all sites.

The organization has adopted multi-vendor strategy for their storage infrastructure:

- XtremIO storage is used for the organization's VDI and other high performing applications.
- VPLEX Metro is used to achieve data mobility and access across both of the NJ and NYC sites. VPLEX metro provides the organization with Access-Anywhere capabilities, where virtual distributed volumes can be accessed in read/write at both sites.
- Disaster recovery solution is implemented by using RecoverPoint for asynchronous continuous remote replication between the metro site and the Iowa site.
- VPLEX metro is used at the Iowa site to improve assets and resource utilization, while enabling replication from EMC to non-EMC storage.

EMC solutions (such as those in the above example) offer unique and superior values, including:

- High availability and performance optimization of multiple paths in a high performing storage environment
- High performance content-aware all flash storage that supports hundreds of thousands of IOPS with low latency and high throughput
- Geographically dispersed clusters with zero RPO
- Automated recovery with near-zero RTO
- High availability within and across VPLEX Metro data centers
- Increased performance as workloads can be shared between sites
- Continuous remote replication (or CDP or CLR) of XtremIO systems

Openstack Integration

OpenStack is the open platform for managing private and public clouds. It allows storage resources to be located anywhere in the cloud and available for use upon demand. Cinder is the block storage service for OpenStack.

The XtremIO Cinder driver enables OpenStack clouds to access the XtremIO storage. The XtremIO Cinder management driver directs the creation and deletion of volumes on the XtremIO array and attaches/detaches volumes to/from instances/VMs created by OpenStack. The driver automates the creation of initiator mappings to volumes. These mappings allow the running of OpenStack instances to access the XtremIO storage. This is all performed on demand, based on the OpenStack cloud requirements.

The OpenStack XtremIO Cinder driver utilizes the XtremIO RESTful API to communicate OpenStack's management requests to the XtremIO array.

The OpenStack cloud can access XtremIO using either iSCSI or Fibre Channel protocols.

Conclusion

XtremIO has developed an advanced revolutionary architecture, which is optimized for all-SSD enterprise storage subsystems. XtremIO offers a rich set of features that leverage and optimize the SSD media capabilities and have been especially designed to provide unparalleled solutions for enterprise customers' needs and requirements.

XtremIO's features include truly-scalable solutions (buy additional capacity and performance when needed), high performance with hundreds of thousands of IOPS, constant sub-millisecond low latency, content-aware Inline Data Reduction, high availability, thin provisioning, snapshots, and VAAI support.

XtremIO also offers a unique patent-protected scheme that leverages the SSD media characteristics to provide an efficient and powerful data protection mechanism which can protect the data against two simultaneous and multiple consecutive failures.

In addition, XtremIO incorporates a comprehensive, intuitive and user-friendly interface which includes both GUI and command line modes, and is designed for ease-of-use while enabling efficient system management.

XtremIO provides the perfect solution for all-SSD enterprise SAN storage while offering a superior total cost of ownership solution for its customers.