KIOXIA

Technical Brief

Introducing High-Performance SSD Data Rebuilds Using Award-Winning KIOXIA RAID Offload Technology

Frees Up System Resources / Reduces the Risk of Data Loss

Collecting, refining and leveraging data is crucial for most industries and governments today and key for businesses to gain a competitive edge by obtaining the best AI, machine learning and data analytics responses. However, these data-intensive applications give rise to data integrity, movement and recovery challenges. To help address some concerns, storage systems with read-intensive workloads are adopting high-capacity SSDs with low-endurance NAND, such as quad-level cell (QLC), to be more cost effective, but this can lead to other challenges:

1. *Lower endurances increase the probability of SSD failures, especially for write-intensive tasks*
2. *Data from data-intensive workloads spend longer time in data recovery due to high capacity, which in turn keeps the storage system in a degraded mode*

The focus of this technical brief is the KIOXIA three-prong strategy to ensure data integrity and how to develop efficient rebuild capabilities if SSD failures occur. The KIOXIA three-prong strategy addresses ways to:

1. *Avoid failure at the SSD level*
2. *Provide an efficient mechanism to detect errors early at the RAID stripe (or object) level*
3. *Efficiently rebuild data using KIOXIA RAID Offload technology if a failure occurs*

Of note, the ways to recover data and the handling of incoming I/O when an SSD is in a degraded mode are implementation specific to a given storage system and will not be addressed in this technical brief.

KIOXIA Policy to Mitigate SSD Failures

KIOXIA SSDs possess a variety of capabilities to avoid complete failures from happening at both the SSD level and object level (or early error detection on a RAID stripe). The table below describes these available capabilities to mitigate failures at the SSD level:

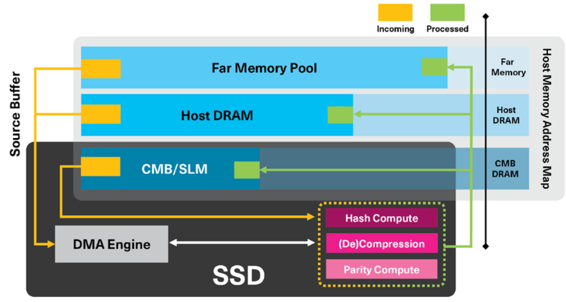
KIOXIA SSD Failure Mitigation Capabilities	Description
Wear Leveling	Internally developed "Always on" algorithm allows even media wearing by distributing write and erase cycles evenly across all storage blocks, preventing any single block from wearing out prematurely due to excessive use.
Flash Die Failure Protection	Enables the SSD to continue functioning with full reliability if a NAND die fails.
Customizable End of Life Behavior	SSD end of life in many cases is not a sudden, complete failure but rather a gradual degradation of the flash memory cells. The ability to customize this behavior can increase SSD lifespans.
Overprovisioning	A technique where a portion of an SSD is reserved for internal use by the drive's controller, rather than being available to the user. This reserved space is crucial for maintaining performance and extending SSD life.
Available Spare	In SMART (Self-Monitoring, Analysis, and Reporting Technology), the Available Spare attribute indicates the amount of spare capacity remaining on an SSD that can be used to replace failing or worn-out flash blocks. An Available Spare warning typically means that this spare capacity has fallen below a manufacturer-defined threshold, indicating potential reliability issues and could suggest that the drive may need to be replaced soon.
End-to-End Data Protection	Ensures data integrity throughout the entire data transfer process, from the host system to the SSD and back. It uses error detection and correction mechanisms at multiple points within the SSD to safeguard against data corruption during storage and transmission.

Using and observing these failure mitigation capabilities, system administrators can make better decisions when replacing SSDs whose health parameters are deteriorating.

Early Error Detection at the RAID Stripe or Object Level

For early error detection at the RAID stripe level or object level, KIOXIA provides access to power-efficient compute engines (i.e., XOR and hash engines) that already exist in each SSD through a standard interface. The image to the right presents the building blocks for early error detection at the RAID stripe level.

Using parity and hash engines within each SSD, the host can detect errors early and take steps to mitigate a complete failure. [KIOXIA Data Scrubbing Offload technology], and based on KIOXIA RAID Offload technology, is an error detection technique that can be used for ensuring data validity when applications need to access the data. It is a background task that periodically inspects stored data for errors. Data scrubbing reduces the likelihood that single correctable errors will accumulate, reducing the risk of uncorrectable errors.



Created by and the property of KIOXIA America, Inc.

The image below describes how the host uses these compute engines:

Created by and the property of KIOXIA America, Inc.

The Generic Rebuild Process

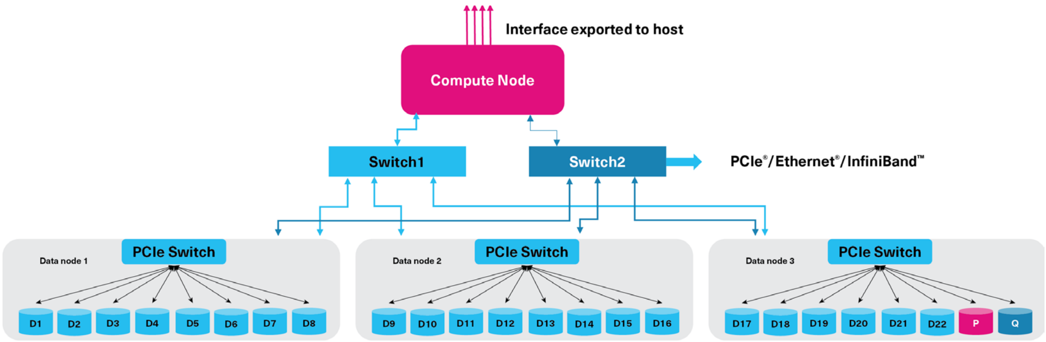
If the storage system supports SSD level fault tolerance, the following steps need to be performed:

1. *Identify failed SSDs*
2. *Replace failed SSDs with the same or higher capacity SSDs*
3. *Start the rebuild process*
4. *Verify rebuilt data on replaced SSDs*

KIOXIA RAID Offload technology can be used for efficient SSD rebuilds and data verification enabling a storage system in degraded mode to efficiently move into normal mode.

The Conventional Rebuild Process

In many aggregated or disaggregated storage systems, the conventional RAID rebuilding process uses good data that resides on other SSDs to rebuild the failed SSDs. A RAID stripe can span across many data nodes, as depicted in the image below:



Created by and the property of KIOXIA America, Inc.

In this configuration, the RAID stripe is spread across 24 drives (22 data drives/2 parity drives) that span across three data nodes. If two SSDs fail in this configuration, the data stored on the two failed drives can be recovered. If the drive size is assumed to be 4 terabytes[1] (TB), then the RAID group size would become 96 TB. Also, in one RAID stripe, the data segments are defined as D1 to D22, and the parity segments defined as P and Q. It is assumed that failed SSDs contained data segment D1 and the parity segment P. In the 96 TB configuration, 8 TB of data is required for rebuilds.

The conventional compute node method of SSD rebuilding requires the following steps:

1. *Replace failed SSDs that contain the D1 data segment and the P parity segment with new SSDs*
2. *Read good data segments D2 to D22 and the Q parity segment on the compute node*
3. *Compute these two equations in order (+ represents XOR operation):*
    a. $D1 = (Q + g2.D2^2 + g3.D3 + ... + g22.D22)/g1$
    b. $P = D1 + D2 + D3 + D4 + ... + D22$
4. *Write the D1 data segment and P parity segment in parallel to the new SSDs on the respective stripe*
5. *Repeat steps 2-4 for each RAID stripe generation*

The resources used for this conventional rebuilding method were determined to be:

System Resources to Rebuild 8 TB	Conventional
Data passes through CPU for parity computation	88 TB
Data passes through DRAM controller	192 TB
Data passes through network	96 TB
Data passes through PCIe interface to/from compute node	96 TB

Efficient Rebuild Process Using KIOXIA RAID Offload Technology

The new and advanced approach to SSD rebuilds are based on KIOXIA RAID Offload technology which earned "Best of Show" honors at the FMS 2024 symposium in the 'Most Innovative Technology,' SSD technology category. In simplest terms, this architecture offloads compute and DRAM bandwidth demands to SSDs delivering the following benefits:

- *Usage optimization of compute resources*
    - *Frees up host CPU, cache and memory resources to focus on primary applications*
- *Improvement in system-level performance*
    - *Initializes/rebuilds a RAID volume at the maximum sequential write performance of an SSD*
- *Efficient scaling*
    - *RAID parity compute throughput scales proportionally as the number of SSDs increases*
    - *Addresses the memory wall issue*
- *Easier integration with the existing infrastructure*
    - *Utilizes the existing mature RAID stack and user interface*
    - *Agnostic to RAID geometry*
- *TCO optimization*
    - *Increases power efficiency in server and storage systems*

Rebuilding One RAID Stripe

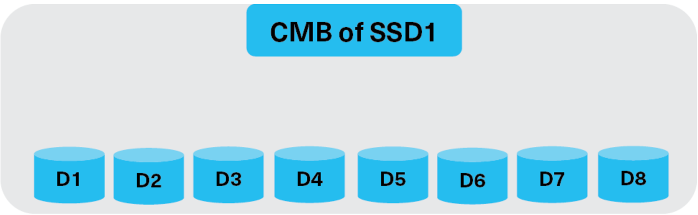
KIOXIA RAID Offload technology employs a 'compute near data' strategy when conducting a rebuild. There are six steps involved in rebuilding SSD data from one RAID stripe:

1. *Replace failed SSDs that contain the D1 data segment and the P parity segment with new SSDs*
2. *In parallel, read data from all good SSDs in a data node to the controller memory buffer (CMB) of one SSD and calculate interim data and a parity segment for that data node*
3. *Transfer the interim data segment and parity segment to the compute node*
4. *Calculate the final data and parity segment on compute node*
5. *Write the final data segment and parity segment to newly added SSDs in step 1*
6. *Repeat steps 2-5 for generating other stripes*

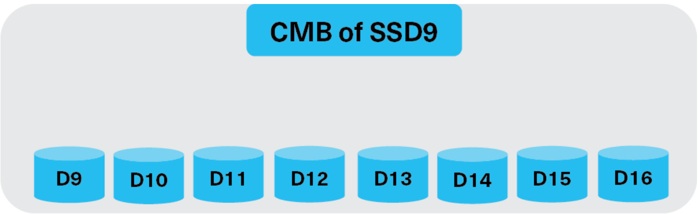
An efficient SSD data rebuilding process should be able to generate data and parity segments fast enough to saturate the write throughput of SSDs in step 5. To meet this goal, each SSD in the RAID setup can participate in rebuilding different stripes in parallel by following these five steps:

1. *Add two new SSDs in data node 1 and data node 3*
2. *Since each SSD has a CMB and parity computation, the host can now read the per node data segments of a RAID stripe into the CMB of one of the SSDs in that node. For example, using the conventional rebuilt image presented earlier, the host performs three steps in parallel for the three data nodes as follows:*

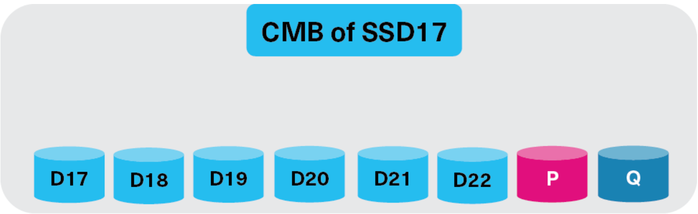
    a. *Read D2 to D8 in the CMB of SSD1 for data node 1:*



    b. *Read D9 to D16 in the CMB of SSD9 for data node 2:*



    c. *Read D17 to D22, and parity segment, in the CMB of SSD17 for data node 3:*



After the steps above occur in parallel for the three data nodes, the host, also in parallel, calculates the interim data and parity segment in each node as follows:

Data Node	SSD CMB	Calculations (+ represents XOR operation)
Data Node 1	SSD1	$Di1 = g2.D2^2 + \ldots + g8.D8$ $Pi1 = D2 + D3 + \ldots + D8$
Data Node 2	SSD9	$Di2 = g9.D9^2 + \ldots + g16.D16$ $Pi2 = D9 + D10 + \ldots + D16$
Data Node 3	SSD17	$Di3 = g17.D17^2 + \ldots + g22.D22 + Q$ $Pi3 = D17 + D18 + \ldots + D22$

3. *Transfer node specific interim data and parity segments (Di1, Di2, Di3, Pi1, Pi2, Pi3) to the compute node. To rebuild one data segment D1 and one parity segment P1 only requires six segment moves to the compute node using KIOXIA RAID Offload technology compared to 22 segment moves using the conventional rebuild method.*
4. *On the compute node, calculate the two equations in order (+ represents XOR operation):*
    a. *D1 = (Di1 + Di2 +Di3)/g1*
    b. *P = D1 + Pi1 + Pi2 + Pi3*
5. *Write the calculated D1 and P results to the newly added SSDs*

When compared with the conventional rebuild process, the table below shows the estimated system resource usage savings[3] when KIOXIA RAID Offload technology is used to rebuild one data SSD and one parity SSD:

System Resources to Rebuild 8 TB	Conventional Rebuild	KIOXIA Rebuild w/RAID Offload	% of Savings
Data passes through CPU for parity computation	88 TB	24 TB	73%
Data passes through DRAM controller	192 TB	64 TB	66%
Data passes through network	96 TB	32 TB	66%
*Data passes through PCIe® interface to/from compute node**	96 TB	32 TB	66%

**Does not account for internal PCIe read operations within data nodes.*

By deploying a 'compute near data' strategy, the rebuild offload process using KIOXIA RAID Offload technology demonstrates high efficiency of system resource usage.

Summary

Existing RAID solutions can use KIOXIA RAID Offload technology to rebuild SSD data, save time, better utilize system resources and deliver scale out performance.

The KIOXIA three-prong strategy for addressing SSD failures includes:

1. *KIOXIA SSD level [data loss mitigation](#)*
    a. *Avoids the rebuilding process, especially for high-capacity SSDs*
    b. *Recovers data even in the case of a die failure*
2. *KIOXIA Data Scrubbing Offload technology*
    a. *Detects failures early*
    b. *Ensures data validity when applications need to access the data*
    c. *Saves significant system resources*
3. *Rebuild using KIOXIA RAID Offload technology*
    a. *Saves significant system resources*
    b *Allows a storage system to move from degraded mode to normal mode in less time, reducing the risk of a drive failure in degraded mode, which could result in data loss*

Key Takeaway:
To rebuild one data segment D1 and one parity segment P1 requires six segment moves to the compute node using KIOXIA RAID Offload technology compared to 22 segment moves using the conventional rebuild method.

More information on KIOXIA enterprise SSDs and advanced technologies available here.

FOOTNOTES:

[1] Definition of capacity: Kioxia Corporation defines a megabyte (MB) as 1,000,000 bytes, a gigabyte (GB) as 1,000,000,000 bytes, a terabyte (TB) as 1,000,000,000,000 bytes and a petabyte (PB) as 1,000,000,000,000,000 bytes. A computer operating system, however, reports storage capacity using powers of 2 for the definition of 1Gbit = 2^{30} bits = 1,073,741,824 bits, 1GB = 2^{30} bytes = 1,073,741,824 bytes, 1TB = 2^{40} bytes = 1,099,511,627,776 bytes and 1PB = 2^{40} bytes = 1,125,899,906,842,624 bytes and therefore shows less storage capacity. Available storage capacity (including examples of various media files) will vary based on file size, formatting, settings, software and operating system, and/or pre-installed software applications, or media content. Actual formatted capacity may vary.

[2] The 'g' in the equation is a Galois coefficient that is computed by the host using a polynomial equation. The 'g.D' in the equation means each byte of data segment 'D' is multiplied by 'g.' In erasure coding, many RAID solutions use proprietary polynomials to generate Galois coefficients.

[3] The estimated system resource usage savings are based on data flow and algorithms used in general during a rebuild process and the estimates were shared with storage solution providers.

TRADEMARKS:

Ethernet is a registered trademark of Fuji Xerox Co., Ltd. InfiniBand is a trademark of the InfiniBand Trade Association. PCIe is a registered trademark of PCI-SIG. All other company names, product names and service names may be trademarks of third-party companies.

DISCLAIMERS: