KIOXIA

Frequently Asked Questions

Questions You Should Ask About KIOXIA RAID Offload

What is KIOXIA RAID Offload?

KIOXIA RAID Offload is a technology that shifts the RAID parity computation and required memory bandwidth to SSDs from host CPU and memory. It is a scalable solution where the overall performance of a RAID system scales proportionally to the number of SSDs. Kioxia Corporation is actively working with the NVMe™ community to standardize KIOXIA RAID Offload, aiming for broader adoption and availability.

Why is KIOXIA RAID Offload needed?

RAID[1] and erasure coding[2] (EC) are highly compute-intensive data redundancy solutions due to extensive parity calculations required during data writes and failure recovery. Typically, RAID/EC computations are performed sequentially on-stream of buffers that can lead to significant CPU cache thrashing. As PCIe® generations advance, NVMe SSD read/write performance typically doubles, exacerbating the problem and shifting performance bottlenecks from storage to RAID and EC. As these data redundancy solutions reside higher in the storage software stack - both in hardware and software - their performance remains constrained due to limited compute and memory resources.

To unburden system compute and memory resources, Kioxia Corporation developed RAID/EC parity computation offloading to SSDs. At the Future of Memory and Storage (FMS) 2024 symposium, a limited performance proof of concept (PoC) was introduced showcasing the technology's potential. The PoC demonstrated a 91% reduction in system DRAM bandwidth utilization using KIOXIA RAID Offload when compared to a conventional software RAID implementation performed on the CPU (*more details here*). KIOXIA RAID Offload earned a Best of Show award in the 'Most Innovative Technology' SSD technology category at FMS 2024, demonstrating an industry need for this innovative and efficient solution.

What resources enable KIOXIA RAID Offload?

There are three foundational resources required to enable KIOIXA RAID Offload technology: (1) NVMe controller memory buffer (CMB); (2) Direct memory access (DMA) engines; and (3) Parity compute engines, as described below:

NVMe Controller Memory Buffer (CMB)

This buffer is used for DRAM offload, exposing some of the controller memory in the NVMe subsystem for host application use.

Direct Memory Access (DMA) Engines

These engines are used to transfer data from one memory buffer to another and use the SSD DMA engine to move data from host DRAM to the SSD CMB to reduce system resource usages.

NOTE: *This data movement from the DMA engine to the SSD CMB is optional as the host can use the CPU or its own DMA engine to move data.*

Parity Compute Engines

These integrated, power-efficient parity compute engines are used primarily to free up resources on the host. The CPU normally computes parity computation of memory buffers, while the parity engine computes parity, relieving significant CPU loads that result in a more efficient process.

What are the key capabilities of KIOXIA RAID Offload, and how can it be adopted?

Many RAID solutions use a variety of RAID geometries, such as diagonal RAID, de-clustered/clustered RAID, variable-sized RAID stripes, etc. KIOXIA RAID Offload is RAID geometry-agnostic, designed to support all RAID geometries.

For EC, many RAID solution providers use propriety polynomials to generate the Galois coefficients. KIOXIA RAID Offload does not require providers to disclose their IP to use it.

Source and output buffers used in parity computations can include the SSD CMB, peer SSD CMBs, host DRAM or other memory. KIOXIA RAID Offload works with all memory buffers to perform parity computations. The technology also provides a correctness check capability of computed parity to the host.

Drive performance depends on the offloaded RAID implementation within each SSD. KIOXIA RAID Offload does not impact the current read/write performance of an SSD and can be used in latency or throughput sensitive I/O workloads.

KIOXIA RAID Offload works in a mixed configuration where some SSDs support this capability, and some do not. It is also compatible with security measures recommended by the NVMe™ specification. The technology can work in virtualized environments and can be adopted in aggregated and disaggregated storage architectures.

What are the key benefits of KIOXIA RAID Offload?

The primary benefit of KIOXIA RAID Offload is its ability to offload system computation and memory resources to SSDs. Offloading compute-intensive, high DRAM bandwidth operations frees up host CPU, cache and memory resources to deliver optimal performance to primary applications. Complementary data scrubbing and rebuilding processes also utilize these offload capabilities to reduce system resource usage.

Improvement in system-level performance is another key benefit of KIOXIA RAID Offload. It initializes and rebuilds a RAID volume at the maximum sequential write performance of an SSD without consuming system resources. The technology also removes some serialization when computing parity that can enhance write throughput.

Efficient scaling is yet another benefit of KIOXIA RAID Offload where RAID parity compute throughput scales proportionally as the number of SSDs increase in the system. The use of SSD CMBs in the KIOXIA RAID Offload configuration helps address memory wall issues that occur when the speed of a processor outpaces the speed of the memory that supplies it with data, limiting overall system performance.

When deploying KIOXIA RAID Offload, it is RAID geometry-agnostic and easily integrates with the existing infrastructure, utilizing both the mature RAID stack and the user interface. The technology delivers increases in server and storage system power efficiency, providing improvements in total cost of ownership.

Notes:

[1] 1 RAID (redundant array of independent disks) is a data storage technology that stores identical data in different locations on SSDs to protect the data in the case of a drive failure. It combines multiple SSD components into one or more logical units for data redundancy and/or performance improvement.

[2] Erasure code is a forward error correction code and a method of data protection that breaks data into sectors, expands and encodes the sectors with redundant data pieces, and stores the redundant pieces across different storage media. It improves data availability and durability, and when compared to RAID, can tolerate more SSD or server node failures, recover data faster, handle larger and more diverse data sets, and support different and flexible configurations.

[3] Definition of capacity - Kioxia Corporation defines a megabyte (MB) as 1,000,000 bytes, a gigabyte (GB) as 1,000,000,000 bytes and a terabyte (TB) as 1,000,000,000,000 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 and 1TB = 2^{40} bytes = 1,099,511,627,776 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.