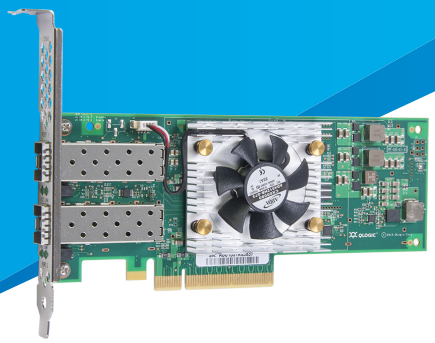
NVMe Direct

Next-Generation Offload Technology



- The market introduction of high-speed NVMe SSDs and 25/40/50/100Gb Ethernet creates exciting new opportunities for external storage
- NVMe Direct enables high-performance iSCSI-based external NVMe flash storage
- Solutions using NVMe Direct deliver full wire-speed performance over 100GbE from SSDs with extremely low latencies

EXECUTIVE SUMMARY

Cavium™ non-volataile memory express (NVMe) Direct offload is a solution for efficiently supporting high-speed, low-latency NVMe solid state drive (SSD)-based external storage over Ethernet networks. NVMe Direct offload is offered as an advanced software feature for the Cavium FastLinQ® 45000 Intelligent Ethernet Adapters.

Using the programmable engine of the Cavium FastLinQ 45000 Series Adapters, NVMe Direct allows the network interface card (NIC) to execute read/write commands directly on the SSD, bypassing the host CPU. NVMe Direct dramatically reduces CPU utilization, since the host CPU is bypassed during read/write transactions. In addition, using NVMe Direct, no host software is used in any part of the read/write operations, which minimizes latency.

The programmable engines of the FastLinQ 45000 Series Adapters are designed to support read/write operations at wire speed on 100Gb Ethernet (100GbE) with I/Os as small as 4K. Furthermore, performance scales linearly when adding more Ethernet ports.

NVMe Direct is an extension of mature target technology, and is transparent and 100% compatible with existing initiator solutions. This significantly reduces risks and streamlines the deployment of NVMe Direct, making it ideal for any application requiring Ethernet-based high-performance flash storage.

TARGET APPLICATIONS

- Storage systems, all-flash arrays (AFAs)
- Disaggregated rack storage
- Scale-out direct-attached storage (DAS)

INTRODUCTION

In an enterprise computer cluster or a cloud-based installation, hosts are commonly accessing remote storage by means of network connectivity. Typically, two-thirds of storage is spent on external storage. With flash devices transitioning from the Serial Advanced Technology Attachment (SATA) and serial-attached SCSI (SAS) interfaces to the NVMe interface, and with a network transition from 10GbE to 25GbE and 100GbE, the industry has a requirement for a new external storage device architecture. Current approaches to building a storage subsystem constrain the performance of these new devices, or burden the cost of the system unnecessarily.

Three factors suggest a market opportunity for a direct Ethernet connection to flash storage. First, flash-based external storage has grown significantly in the past two years, nearly doubling from 7% of capacity shipped to 13% of capacity shipped.[1] Second, the majority of external storage devices use Ethernet connectivity, with 63% of all ports reported to be 10GbE.[1] Finally, there is a resurgence of interest in DAS—such as Just a Bunch of Disks (JBOD) enclosures—with a 13% growth of unit shipments year to year.[1]

1 Dell'Oro, 2015

This white paper describes NVMe Direct, a unique Cavium Ethernet storage offload that allows peer-to-peer communications over PCI Express (PCIe) between the network front-end and the storage back-end on an external storage device. Using this offload, the CPU and memory subsystems no longer perform read/write operations, making it feasible to build a lower-cost Ethernet flash JBOD or a high-performance scale-out system.

USE CASES FOR NVMe DIRECT

There are three primary use cases for NVMe-based DAS as illustrated in Figure 1, in which the array with NVMe Direct is an Ethernet-connected JBOF (Just a Bunch of Flash). Traditional network storage systems and AFAs typically have requirements for the internal storage of these systems to scale beyond what can fit inside a single enclosure. Such storage systems may use either some form of internal fabric such as Fibre Channel or iSCSI, or direct connectivity to the back-end storage, such as SAS, to add additional shelves of storage. In a similar manner, there is interest in disaggregating the internal storage from rack servers, particularly in high-density servers, such as half-wide platforms, by the use of external storage. Finally, new scale-out storage systems usually have requirements for using SAN storage shared by all of the storage controllers.

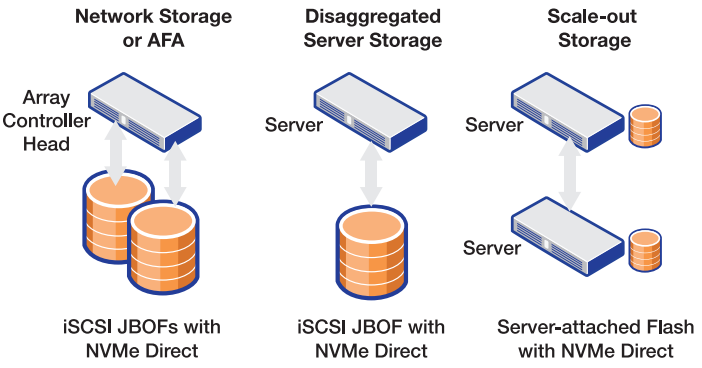


Figure 1. NVMe Use Cases

Each of these use cases can use one of several different wire protocols for the communications between the server or storage front-end to the storage device or storage back-end. The most mature of these protocols is iSCSI. While the initial implementation of NVMe Direct is focused on iSCSI due to its maturity, routability, and the use of TCP/IP for congestion management, these techniques can also be applied to Fibre Channel over Ethernet (FCoE), iSCSI Extensions for RDMA (iSER) or NVMe Fabric. NVMe Direct is orthogonal to the wire protocol being used and focused on optimizing the data path from the network to the storage device.

The extreme performance of NVMe flash devices, coupled with the extreme performance of 100GbE, place significant challenges on the CPU complex and memory subsystem of a storage device in attempting to achieve full potential of the system. Typical I/O operations without NVMe Direct require the host CPU to handle the processing for each I/O request and I/O response involved in a read or write request. NVMe Direct obviates this by performing peer-to-peer I/O-over-PCIe between the flash drive and the networking interface, entirely bypassing the CPUs for all read and write operations.

Standard implementation of the network storage element is based on server architecture. In server architecture, a packet received from the network is forwarded from the receiving NIC to the CPU, and then processed by the CPU. Next, the CPU accesses the storage device. Once the access completes, the CPU sends a response packet through the NIC.

In recent years, the rate at which both network and storage speeds have advanced far exceeds the server speed-up rate. While scaling the server is possible, it is both costly and power-hungry. As noted earlier, there has been a growing demand for DAS, particularly in scale-out architectures, which has initiated a new generation of dumb storage boxes known as JBOD. A JBOD can pack a large number of disks connected by a SAS expander. The current JBOD interface is SAS-based, which limits it to local connectivity. The market trend is to build a JBOD with an Ethernet interface. This replaces the local SAS connectivity with a more flexible and switchable high-speed connection. With efficient network protocol technology such as iSER, achieving 100GbE line rate for 4K I/O blocks or larger would require a “high-end server,” which is not a viable solution for a JBOD.

HOW NVMe DIRECT WORKS

NVMe Direct is an extension to the standard Cavium iSCSI target offload. Figure 2 illustrates a software view of a system implementing a target using NVMe Direct. Architecturally, this is nearly identical to a standard software target, such as Linux-IO Target (LIO™) or SCSI target subsystem for Linux (SCST), using an offloaded Host Bus Adapter (HBA) interface.

The software components include:

- A generic target engine such as Linux SCST or LIO
- An HBA fabric module used to connect to an offloaded NIC
- An NVMe storage module used to connect to an NVMe SSD

The only changes from a traditional software target are:

- A control connection between the HBA driver and NVMe driver as shown by the blue arrow in Figure 2.
- The peer-to-peer PCI connection between the FastLinQ 45000 Series Adapter and an NVMe SSD as shown by the red arrow in Figure 2.

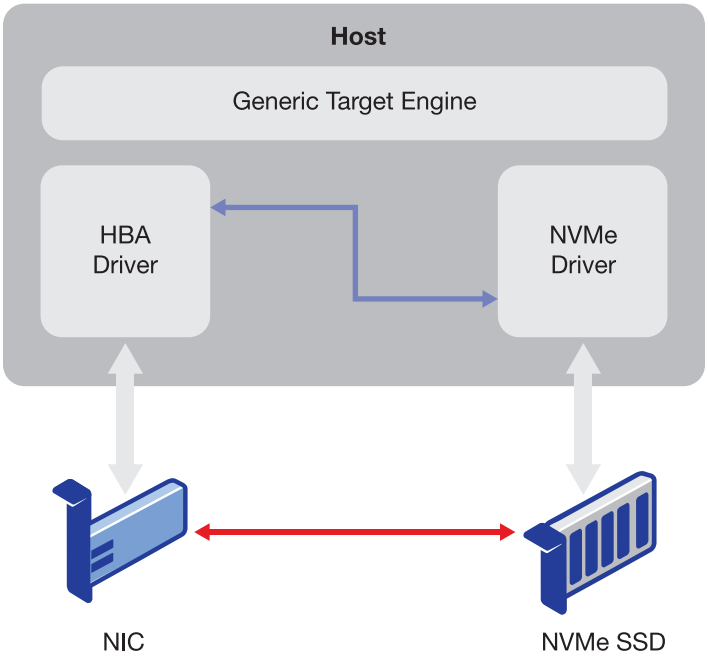


Figure 2. Software View

With NVMe Direct, the target protocol and control plane are untouched. This allows the use of standard initiators without changes to how targets and disks are discovered, controlled, and accessed. The direct connection complements, rather than replaces, the standard data path. The host can still access both the NIC and the SSD with no new limitations. It is also transparent to the NVMe device, with an unmodified NVMe upstream driver used to ease adaptation.

Under the NVMe standard, any SSD is accessed by a number of submission queues (SQs) and completion queues (CQs), also known as queue pairs (QPs). Commonly, there is one global admin QP and an additional QP per CPU core for best performance. With NVMe Direct, additional QPs are established to the NIC. The control connection, illustrated by the blue arrow in Figure 2, is used for initialization—specifically to establish or remove NVMe QPs. Using NVMe Direct, the control connection will establish additional QPs dedicated to the NIC to ensure that CPU QPs remain operational. While Figure 2 shows just one NIC and one SSD for simplicity, this scheme can be implemented with any number of NICs and SSDs as long as the PCIe and memory controller in the system can support the required bandwidth.

The target engine is adapted to support new HBA driver APIs that include:

- Add/remove offloaded LUN
- Get statistics of an offloaded LUN

NVMe read and write access is done by:

- Writing a request to the SQ with a read/write data pointer
- Writing a doorbell to the NVMe device to start operation
- Getting an interrupt indication that an operation has completed
- Reading CQ to retrieve completion status

When using NVMe Direct, read/write operations are performed by the NIC rather than by the host CPU. The NVMe completion interrupt MSI-X message is targeted at the NIC doorbell address instead of CPU interrupt logic, so that any completed commands on NIC QPs will wake up the NIC when completed. Figure 3 illustrates a read flow executed by the NIC with data buffers and QPs residing in the host's memory. Based on this flow, there is zero CPU utilization related to the read operations. Write flow is almost identical and also does not involve the CPU.

Read Request Prospecting Illustration

1. Upon receipt of the read request packet on the Ethernet interface:
   - NIC processes and strips wire protocol to SCSI layer
   - NIC validates the request
   - NIC translates SCSI to NVMe
2. NIC writes to NVMe SQ in host memory over the PCIe
3. NIC triggers NVMe doorbell using PCIe peer-to-peer communication
4. NVMe executes the read and performs a direct memory access (DMA) transfer of the resulting data and CQ element to host memory
5. Upon receipt of TX doorbell via NVMe MSI-X PCIe message:
   - NIC reads data by DMA
   - NIC prepares SCSI response and sends it out

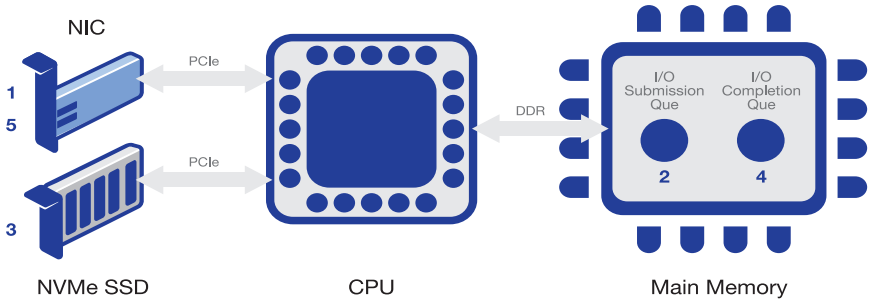


Figure 3. Flow Illustration

DESIGN CONSIDERATIONS FOR PERFORMANCE

The Cavium NVMe solution supports iSCSI and NVMe Direct over 100GbE at line speed for 4K and larger block sizes, which are typical for many applications. The solution scales linearly when adding more ports, as each FastLinQ 45000 Series Adapter is powerful enough to support 100GbE. Multiple cards can be installed in the same system, providing there is sufficient PCIe and memory bandwidth. The maximum IOPS required to support full line rate (wire speed) can be calculated by dividing the line speed by the number of bytes per I/O on the line. Figure 4 shows the theoretical performance per line rate based on the use of a 4KB I/O request.

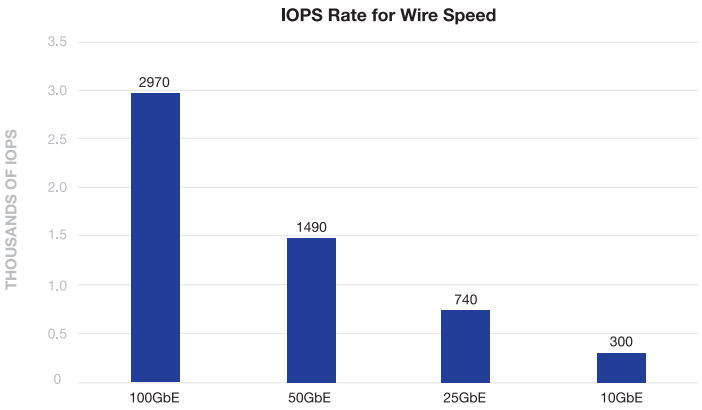


Figure 4. IOPS Rate for Wire Speed

ACCESS LATENCY

The NVMe Direct technology has a two to three-times latency advantage over the conventional approach. When remotely accessing SSD over the network, the majority of the latency is from the NVMe flash drive, while the latency of the FastLinQ 45000 Series Adapter with NVMe Direct access is much smaller. With a light traffic load, the latency using NVMe Direct is about 6.5 microseconds (usecs). This latency refers to the following:

- The time it takes to receive the network request packet until posting the doorbell to the NVME device (after writing the SQ command)
- The time from getting the doorbell from the NVME device (after the NVME device wrote the CQ element) until transmitting the information on the wire

This latency of 6.5 usecs compares to 13 to 20 usecs or more when not using NVMe Direct The latency improvement of NVMe Direct over the conventional approach in remote access would be between two and three times with current SSD technology. As SSD latency reduces over time, the improvement factor increases.

STORAGE SYSTEMS AND DISAGGREGATED RACK SOLUTIONS

Figure 5 and Figure 6 describe two alternatives of designing a storage shelf based on Cavium NVMe Direct using FastLinQ 45000 Series Adapters.

In Figure 5, the QPs and data buffers reside in the main system's memory. This dictates a CPU that has enough memory bandwidth and sufficient number of PCIe lanes to support this application. Table 1 lists these requirements as a function of the interface line rate.

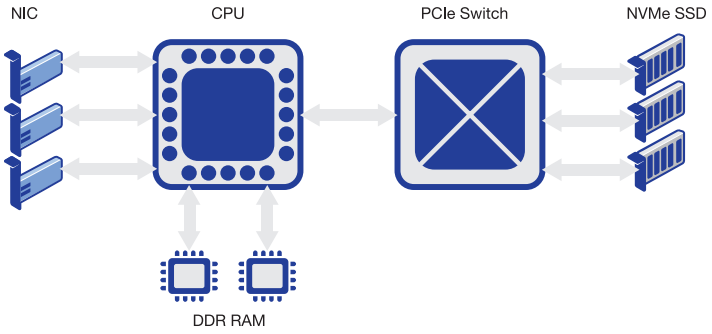


Figure 5. Storage Shelf Using Main Memory Buffers

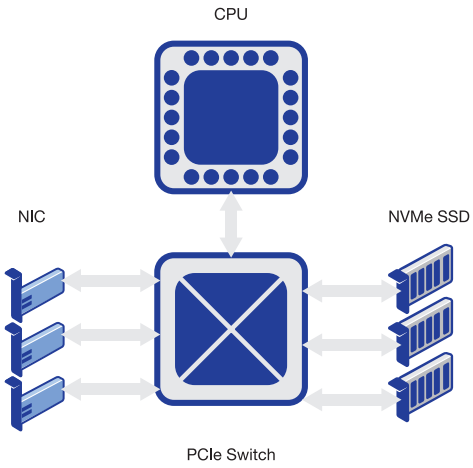


Figure 6. Storage Shelf Using Memory Buffers on NVMe Devices

Table 1. System on a Chip (SOC) Requirements

Line Rate	Memory Bandwidth	PCIe Lanes
100Gbps	64GB	32
50Gbps	32GB	16

In Figure 6, the QPs and data buffers are stored in a memory buffer on each NVMe disk as defined by the NVMe v1.2 optional controller memory buffer (CMB) feature. This significantly lowers the CPU requirements because, as in this schema, the CPU is used for management only and does not have to tunnel the data through main system memory. Using this approach, there are no specific requirements for the CPU memory and PCIe interface.

Using either alternative will guarantee price/performance gains as described above. Additionally, by using a target implementation as previously discussed, JBODs can be quickly deployed using the same mature storage semantics with low integration effort.

SCALE-OUT DAS SOLUTION

In a scale-out DAS deployment, a mesh of servers share direct-attached local storage devices as a network shared pool. Implementing NVMe Direct as shown in Figure 3—where a Cavium FastLinQ 45000 target adapter is installed in a standard server—allows for a higher performance storage pool while freeing valuable CPU resources. To utilize this deployment scheme, Cavium is open for partnerships and integration with scale-out storage solution providers.

FEATURES SUMMARY

- Terminates iSCSI read/write operation
- Supports link speeds of 10/25/40/50/100GbE
- Supports full line rate for I/O block sizes of 4K and greater
- Supports any I/O size

SUMMARY

The availability of Cavium FastLinQ 45000 Series Controllers with NVMe Direct enables storage system developers to deliver very efficient flash-based arrays offering the highest performance and lowest latency. Cavium FastLinQ 45000 Adapters can deliver all common Ethernet block storage transports including iSCSI, iSER, FCoE and NVMe over Fabric. The implementation of NVMe Direct over iSCSI is just the first possible implementation of NVMe Direct. NVMe Direct is offered as an advanced software feature for Cavium FastLinQ 45000 Adapters.

Using the programmable engine of FastLinQ 45000 Series Adapters, NVMe Direct allows the NIC to execute read/write commands directly on the SSD, bypassing the host CPU. This offloaded data path of NVMe Direct dramatically reduces CPU utilization, since the host CPU is bypassed during read/write transactions. In addition, when using NVMe Direct, no host software is used in any part of the read/write operations, minimizing latency.

The programmable engines of FastLinQ 45000 Series Adapters are designed to support read/write operations at wire speed on 100GbE with I/Os as small as 4K. Furthermore, performance scales linearly when adding additional Ethernet ports.

NVMe Direct is an extension of mature target technology, and is transparent and 100% compatible with existing initiator solutions. Leveraging mature technology and existing initiators significantly reduces risks and streamlines the deployment of NVMe Direct, making NVMe Direct ideal for any application requiring Ethernet-based high-performance flash storage, including storage systems and AFAs, disaggregated rack storage, and scale-out DAS.

ABOUT CAVIUM

Cavium, Inc. (NASDAQ: CAVM), offers a broad portfolio of infrastructure solutions for compute, security, storage, switching, connectivity and baseband processing. Cavium's highly integrated multi-core SoC products deliver software compatible solutions across low to high performance points enabling secure and intelligent functionality in Enterprise, Data Center and Service Provider Equipment. Cavium processors and solutions are supported by an extensive ecosystem of operating systems, tools, application stacks, hardware reference designs and other products. Cavium is headquartered in San Jose, CA with design centers in California, Massachusetts, India, Israel, China and Taiwan.

Follow us:

Corporate Headquarters Cavium, Inc. 2315 N. First Street San Jose, CA 95131 408-943-7100

International Offices UK | Ireland | Germany | France | India | Japan | China | Hong Kong | Singapore | Taiwan | Israel