

SNIA DEVELOPER CONFERENCE



By Developers FOR Developers

Hyatt Regency Santa Clara, CA
September 15-17, 2025

A decorative graphic consisting of a series of dots forming a wave that flows from left to right across the middle of the slide. The dots are colored in a gradient from purple to yellow to light blue.

Can SDXI Work with NVMe?

Shyam Iyer, Dell & co-chair SDXI + CS Subgroup

Jason Molgaard, Solidigm & co-chair SDXI + CS Subgroup

www.sniadeveloper.org

SNIA SDXI+CS Subgroup

➤ What is the Subgroup

- The SDXI and CS TWGs collaborating to combine NVMe, SDXI, and CS

➤ Objectives

- Develop a unified block diagram that imagines a combined CS and SDXI system and architecture
- Develop use cases for SDXI-based CS devices
- Consider if enhancements to NVMe are necessary to enable this combination
- Develop a white paper discussing the combination of NVMe and SDXI

SDXI + NVMe White paper Goals and Overview

➤ White paper Goals

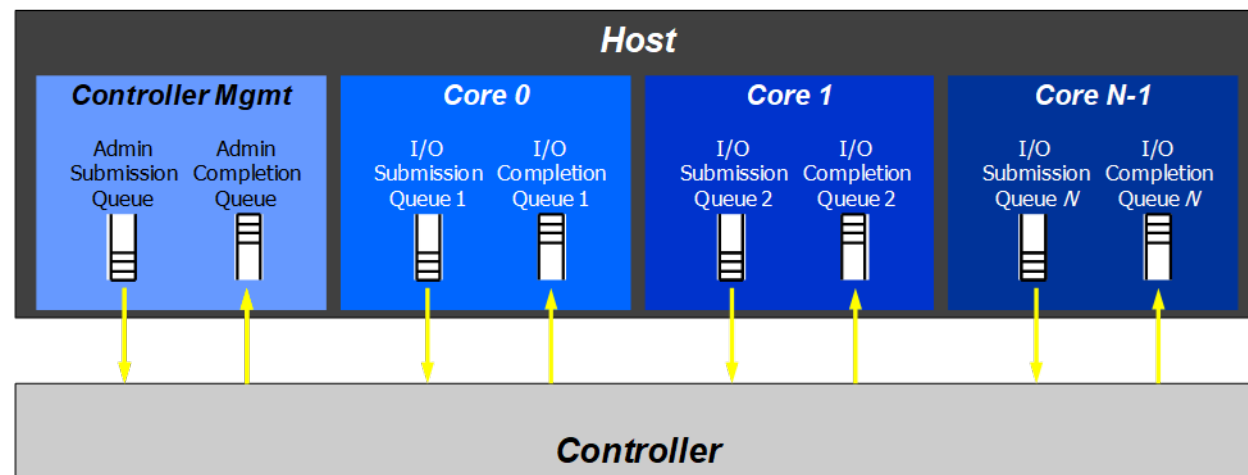
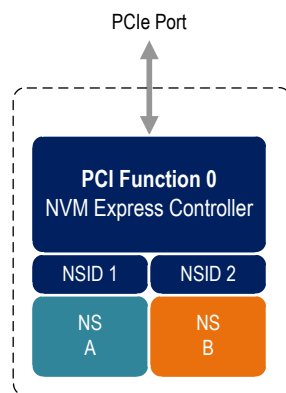
- Apply new emerging features of SNIA SDXI standard for memory data movement to the widely adopted NVMe standard for storage
- Enable NVMe developers to complement system design with SDXI's data movement features:
 - User space addressing
 - Memory targets for source and destination
 - Data Transformations
 - Virtualization ready features
 - Extensibility beyond storage use cases

➤ White paper Overview

- NVMe high-level overview
- Typical NVMe I/O
- SDXI high-level overview
- SDXI Use Cases
- Typical SDXI Operation
- Application NVMe Write I/O flow with SDXI
- Application NVMe Read I/O flow with SDXI
- Peer-to-peer flows with NVMe and SDXI
- Security

NVMe Introduction

- Basic NVMe architecture involves an NVM Express Controller on a PCIe function and namespaces connected to storage media
- NVMe I/O Requests:
 - Requests are placed into submission queues
 - Completions are placed into completion queues
- Storage flows involve **data movement**:
 - From host memory to device memory (NVMe Controller)
 - From device memory (NVMe Controller) to host memory



SDXI Introduction

Smart Data Accelerator Interface (SDXI) is a SNIA standard for a memory to memory data movement and acceleration interface that is -

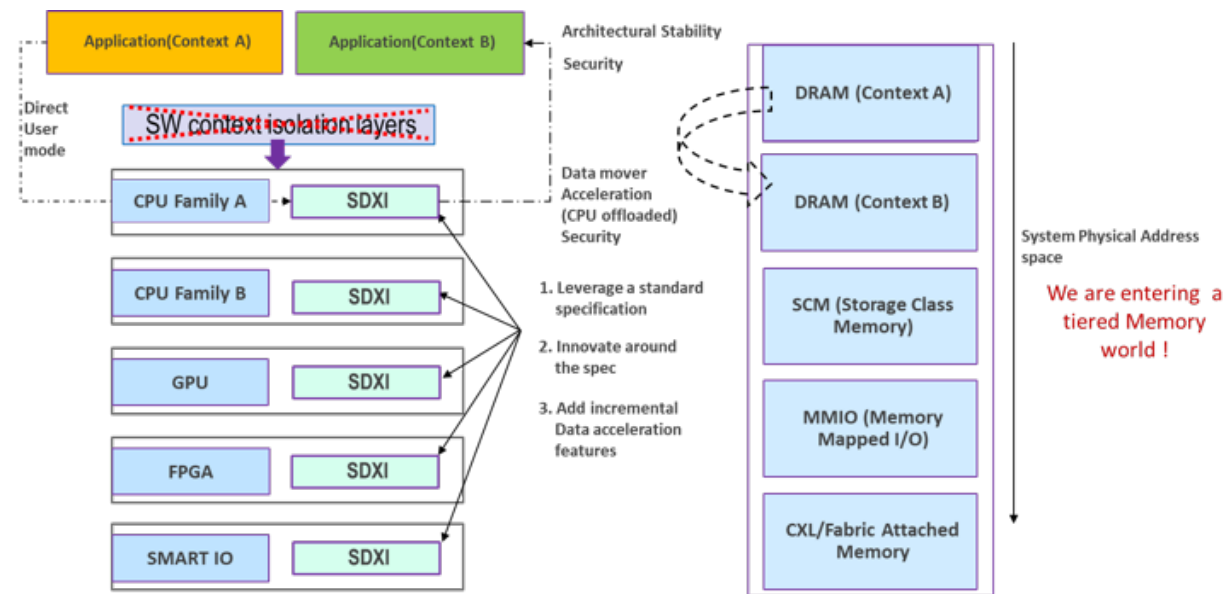
- Extensible
- Forward-compatible
- Independent of I/O interconnect technology
- Features:
 - Virtualized address space to address space data movement
 - Offloads data movement, common memory operations, and data transformations while moving data
 - Offloads data movement while preserving address space and context isolation.
 - Standardized interfaces and architected states for DMA engine
 - Standardized for user-level software.

v1.0 released!

<https://www.snia.org/sdxi>

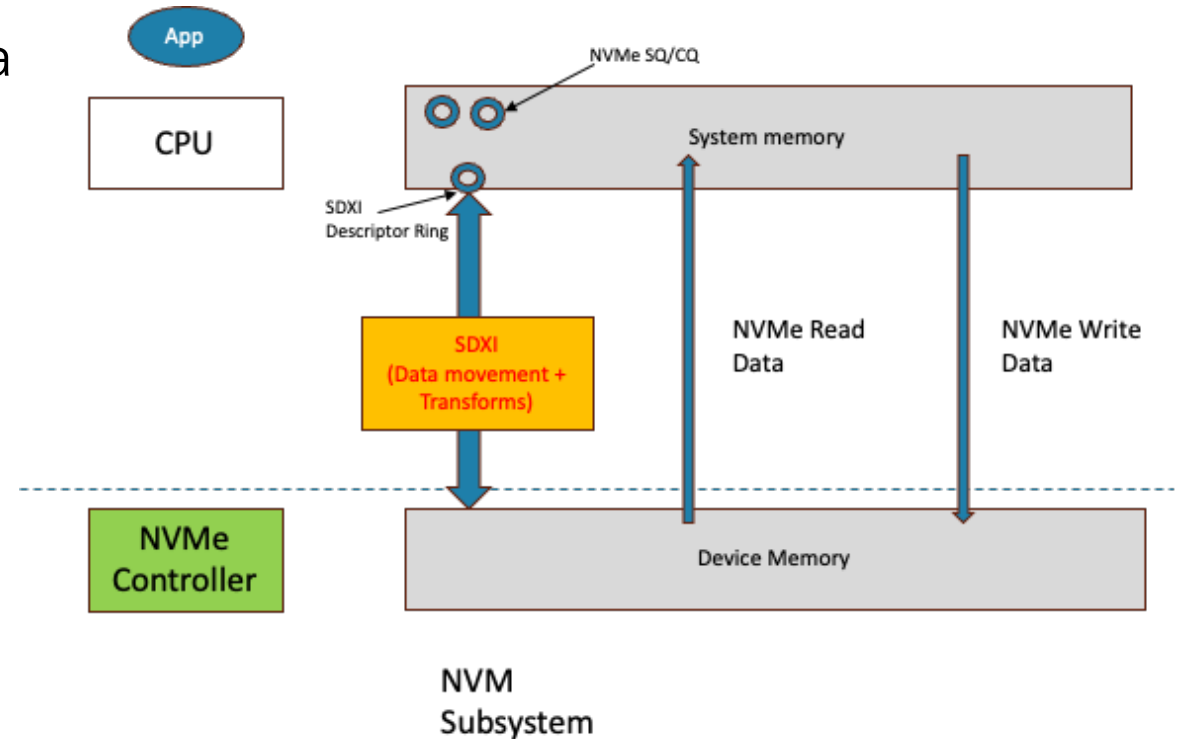
SNIA's SDXI TWG is now working on v1.1 now

- SDXI TWG also has a software focused group that is working on a reference libsdxi implementation



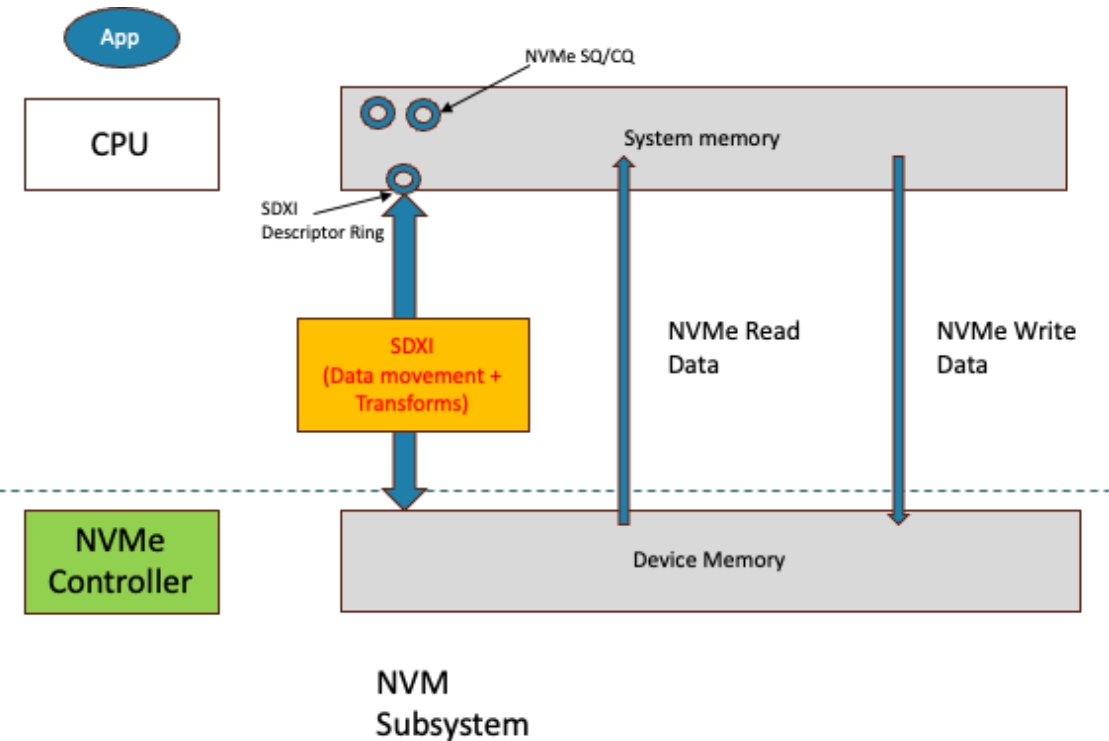
NVMe + SDXI Write IO Flow

- The host issues an SDXI descriptor to perform a memory-to-memory copy of data from the host memory to the NVMe controller's CMB/PMR
 - This device memory is host addressable and can easily be referenced by SDXI in the host
- While transferring the data, SDXI engines can optionally perform transformations to the data before writing the data to CMB/PMR
- Once the data has been written to CMB/PMR, the host issues an NVMe write command or NVMe copy command that refers to the data in CMB/PMR to transfer the data to persistent storage



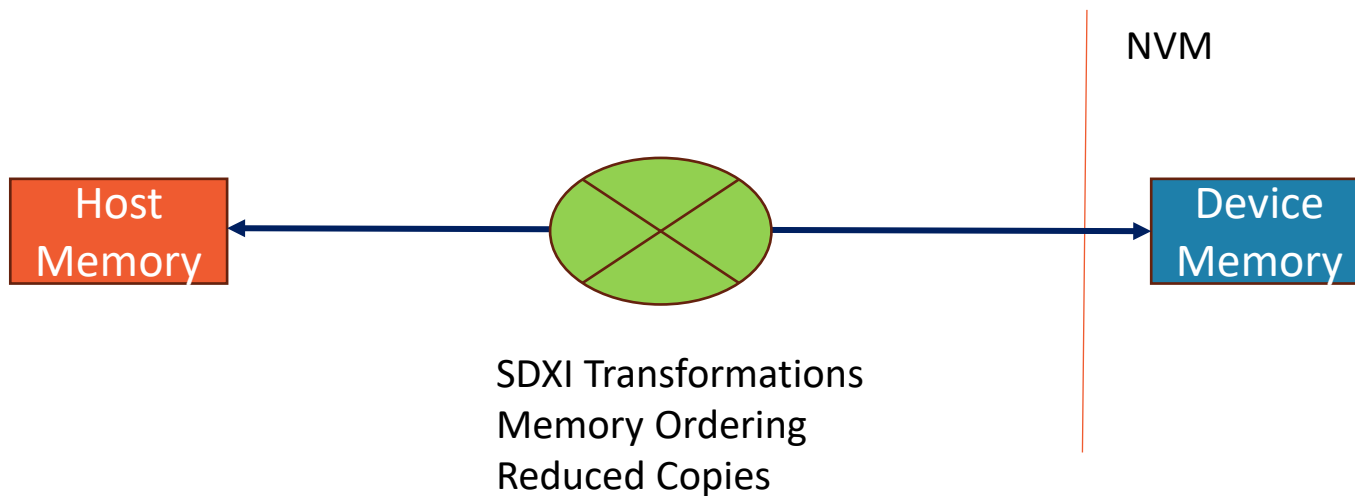
NVMe + SDXI Read IO Flow

- The host issues an NVMe read command to read the data from persistent storage and write the data to CMB/PMR
 - The CMB/PMR device memory is host addressable and can easily be referenced by SDXI in the host
- Now that the data is in CMB/PMR, the host issues an SDXI descriptor to transfer the data from CMB/PMR to host memory
- While transferring the data, SDXI can optionally perform transformations to the data before writing the data to host memory



Benefits of NVMe + SDXI IO

- SDXI data movement can leverage built-in transformations to manipulate data while it is transferred
- SDXI is designed for memory access, memory ordering intricacies and byte-size addressability
- SDXI reduces buffer copies



NVMe + SDXI Security

- NVMe + SDXI does not materially change the security posture of the interaction of the NVM subsystem with the host
- SDXI enables isolation of memory address spaces (airgap)
 - An airgap between memory in different address spaces is a common security practice
- SDXI may encrypt data as it is being moved
 - Other technologies may be used to provide link encryption, such as PCIe IDE

Summary and Call to Action

- SNIA's SDXI+CS subgroup is working on a white paper
 - White paper imagines a system architecture
 - Augments NVMe I/O flows with SDXI data movement and transformations
- System designers and developers can use this white paper to design new architectures involving NVMe and SDXI subsystems.
- Join the subgroup!



Thank you for attending!

Please remember to rate this session. You get access the presentations at
<http://sniadeveloper.org/conference>