SNIA DEVELOPER CONFERENCE

SDC 24

BY Developers FOR Developers

September 16-18, 2024
Santa Clara, CA

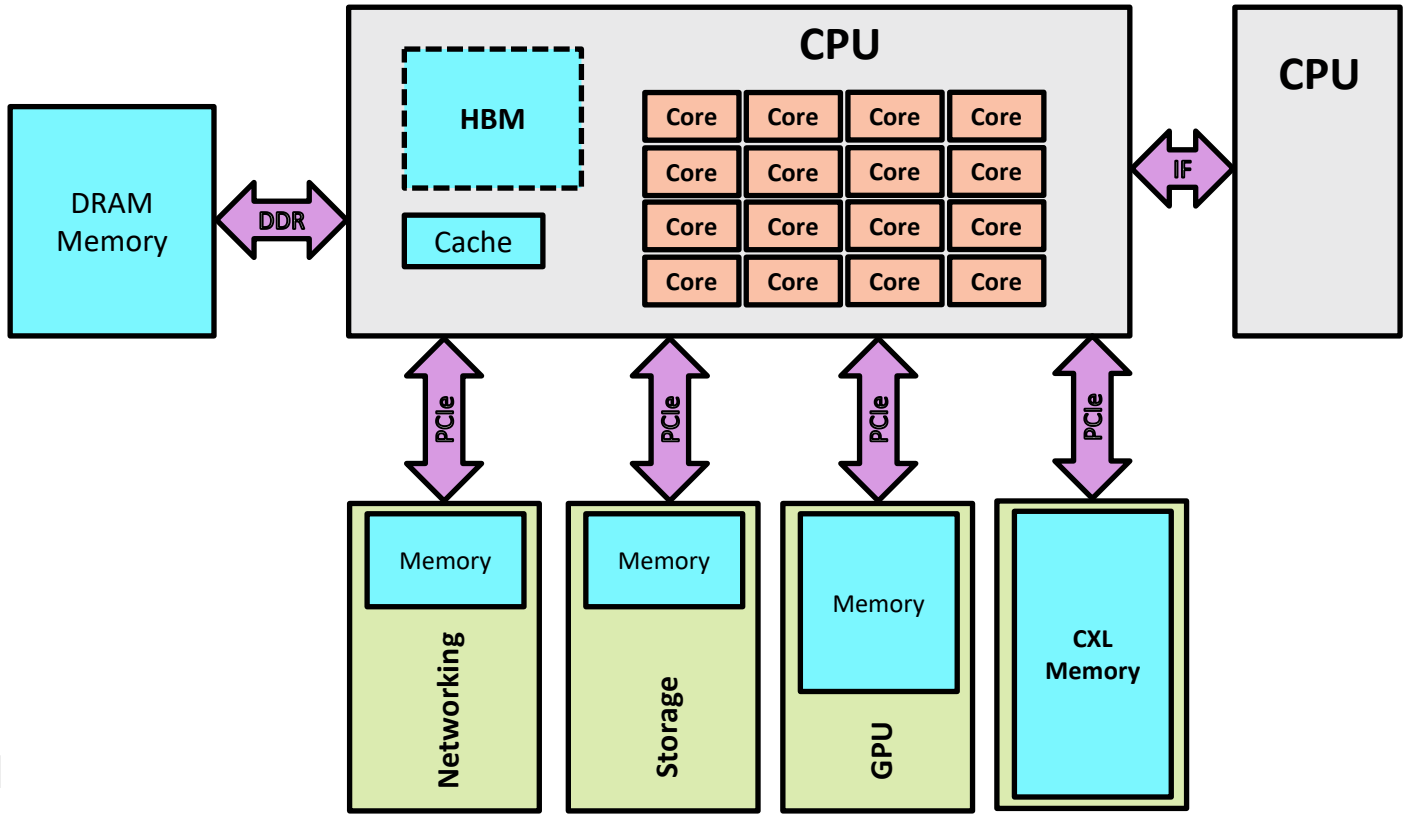
SDXI + Computational Storage Overview

And Panel Discussion

Shyam Iyer, Fred Knight, Bill Lynn, Jason Molgaard, Mats Oberg

Contemporary Enterprise System

- Chiplet based SoC with a large number of cores
  - Could be single or dual socket
  - Memory coherent link to second CPU
  - Multiple levels of internal cache
- System Memory
  - Has multiple DDR channels to increase capacity and bandwidth
  - Could have HBM integrated in package
  - DRAM attached through CXL
- Storage attached using PCIe
  - Storage may be external or local to the system
  - Storage has almost completely transitioned to flash

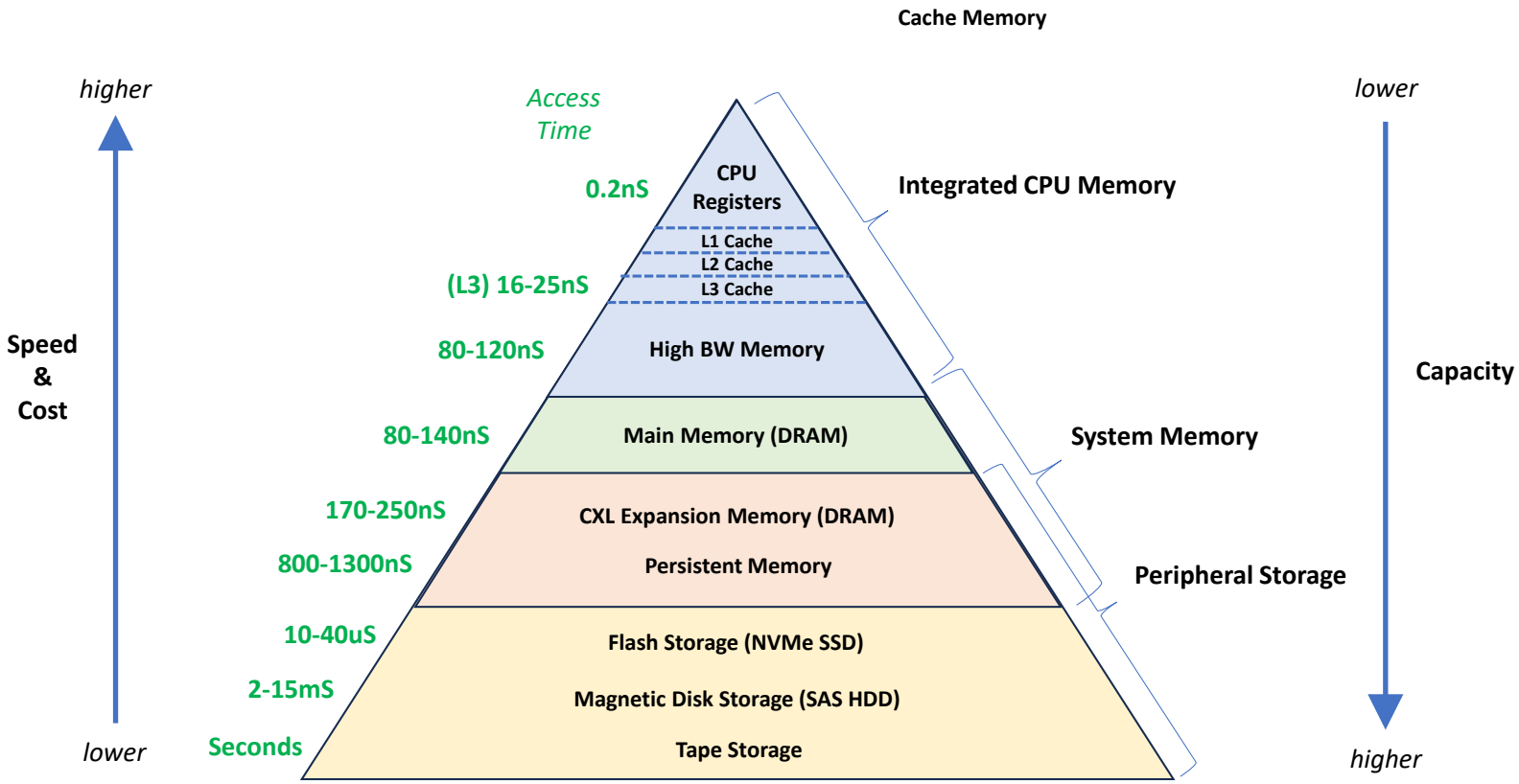


Peripherals tending to have significant amounts of local memory and processing power

Contemporary Memory Pyramid

Larger number of levels

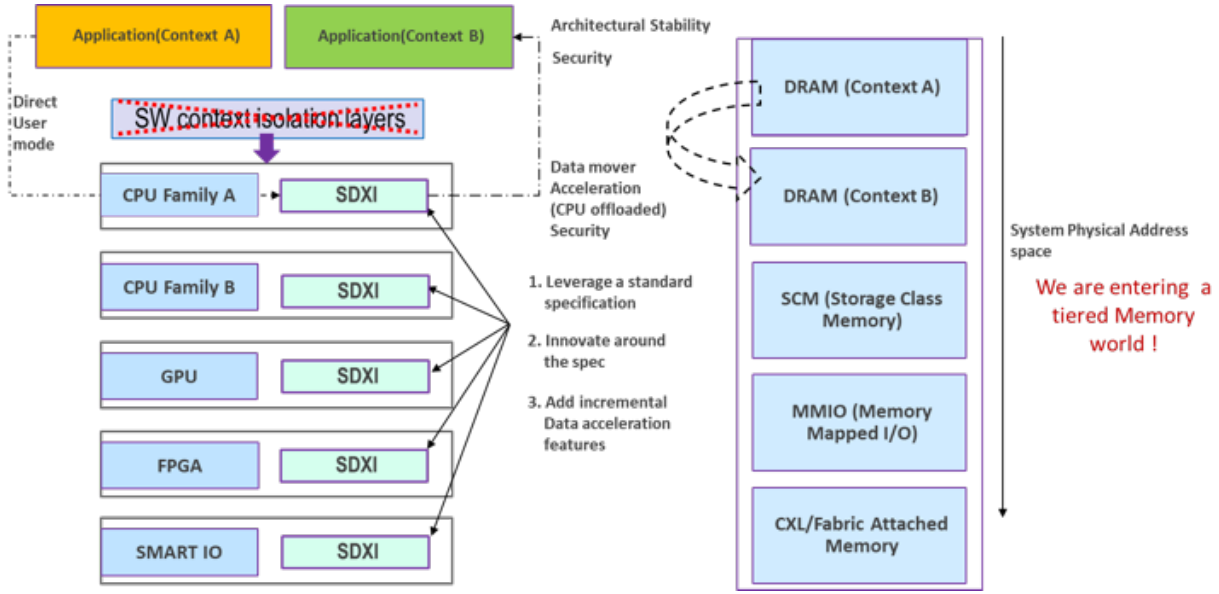
- Levels tend to overlap
- Boundary between system memory and storage becoming fuzzy.



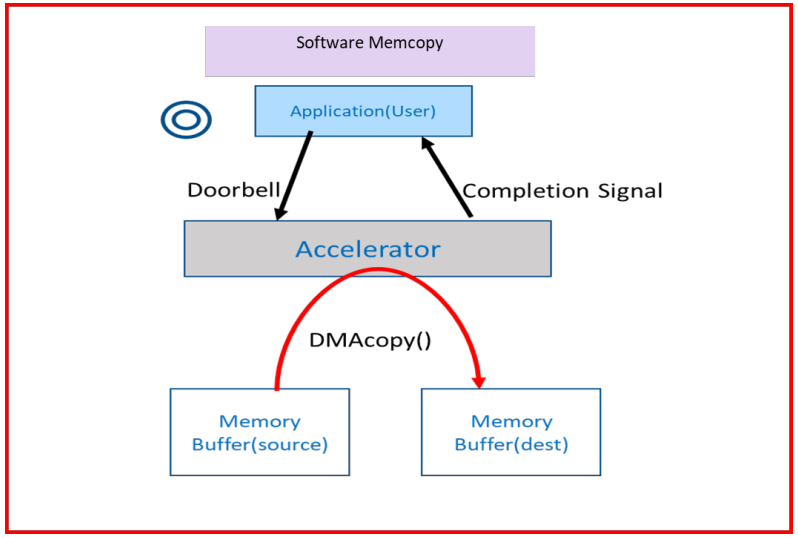
With many layers of memory having a standardized data mover becomes a requirement

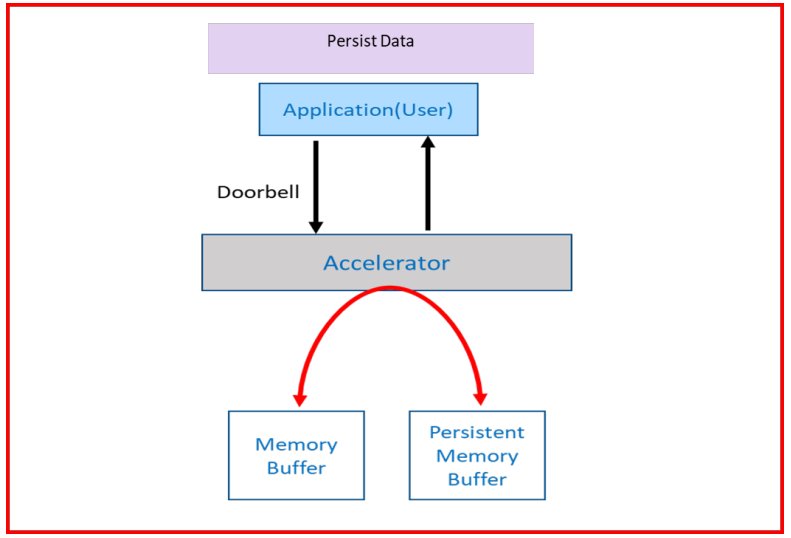
SDXI Intro

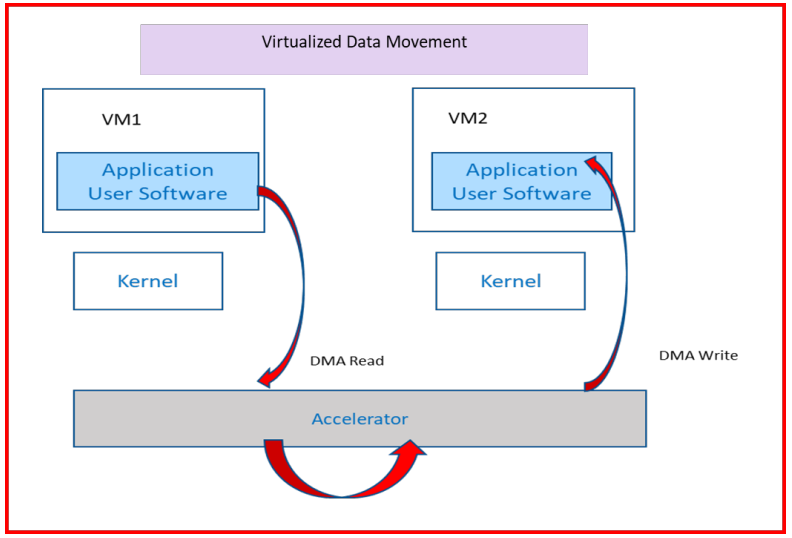
- Smart Data Accelerator Interface (SDXI) is a SNIA standard for a memory to memory data movement and acceleration interface that is -
  - Extensible
  - Forward-compatible
  - Independent of I/O interconnect technology
  - Features:
    - Virtualized address space to address space data movement
    - 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

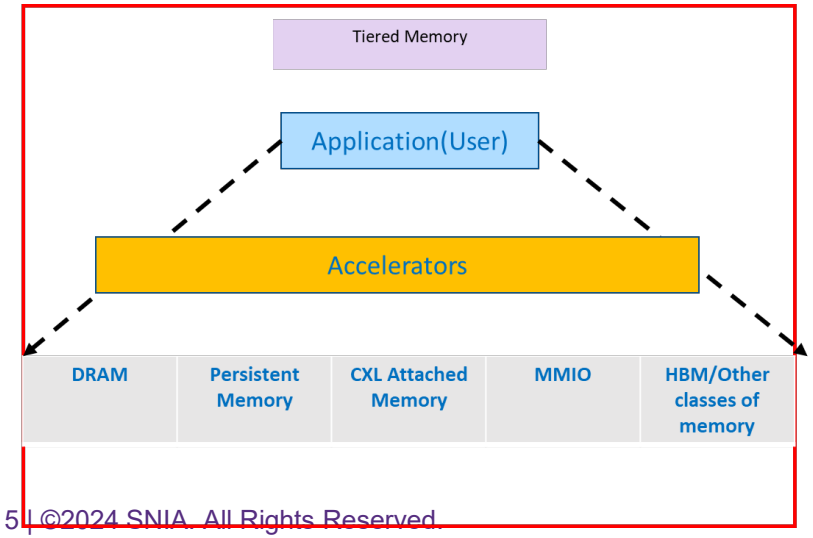


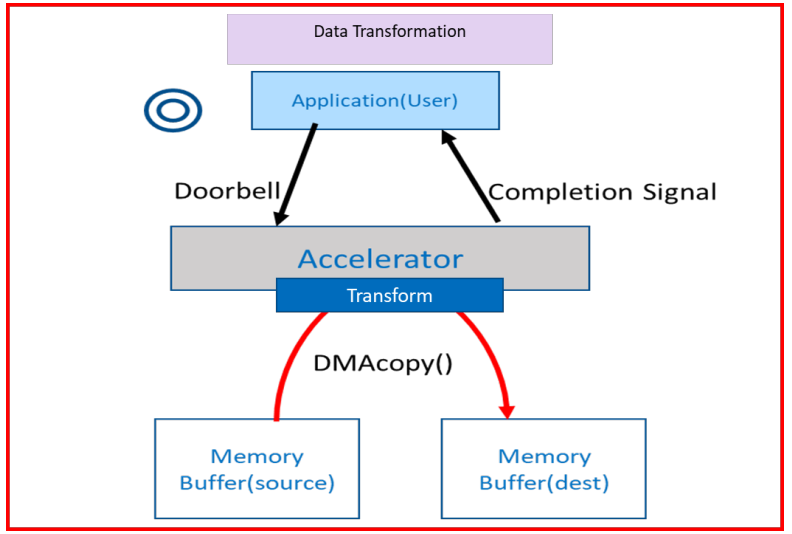
Use cases

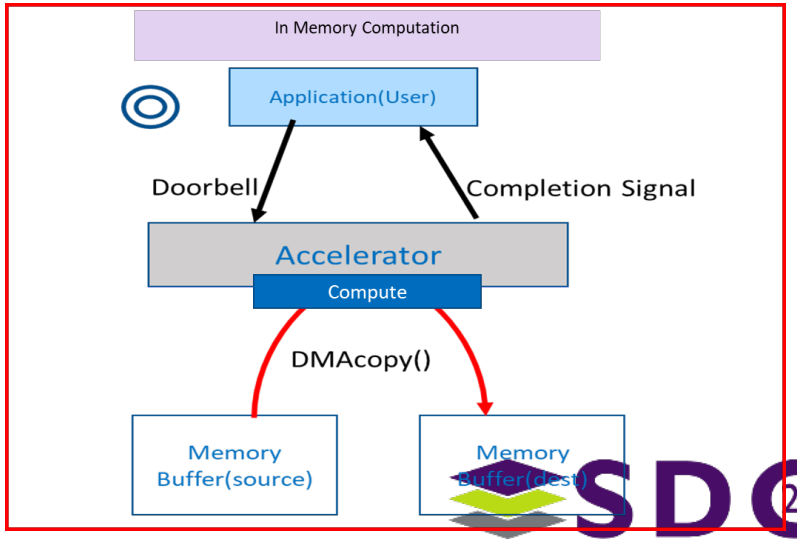






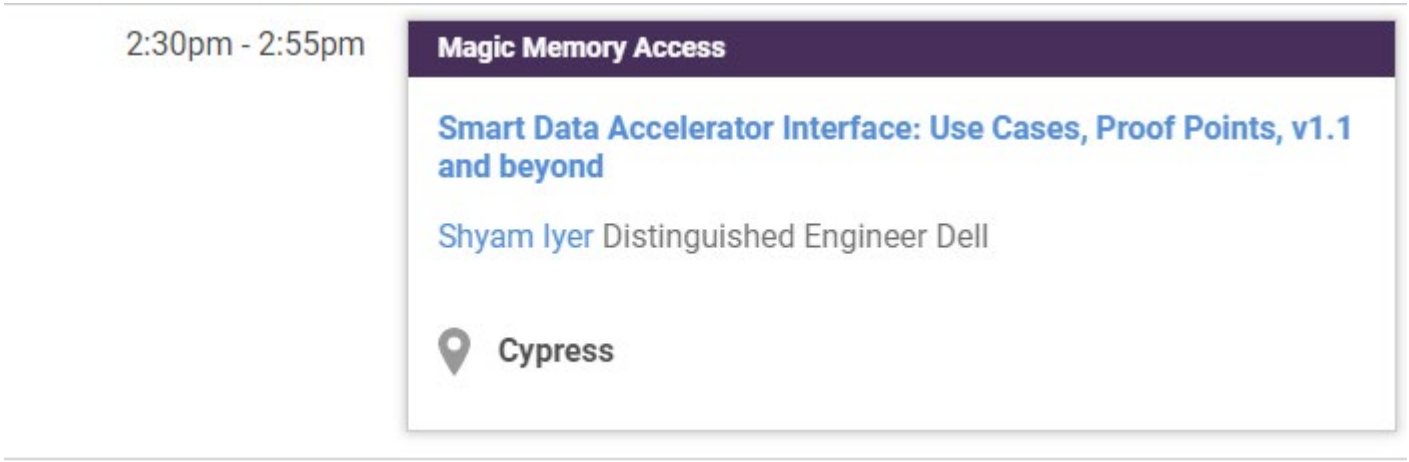






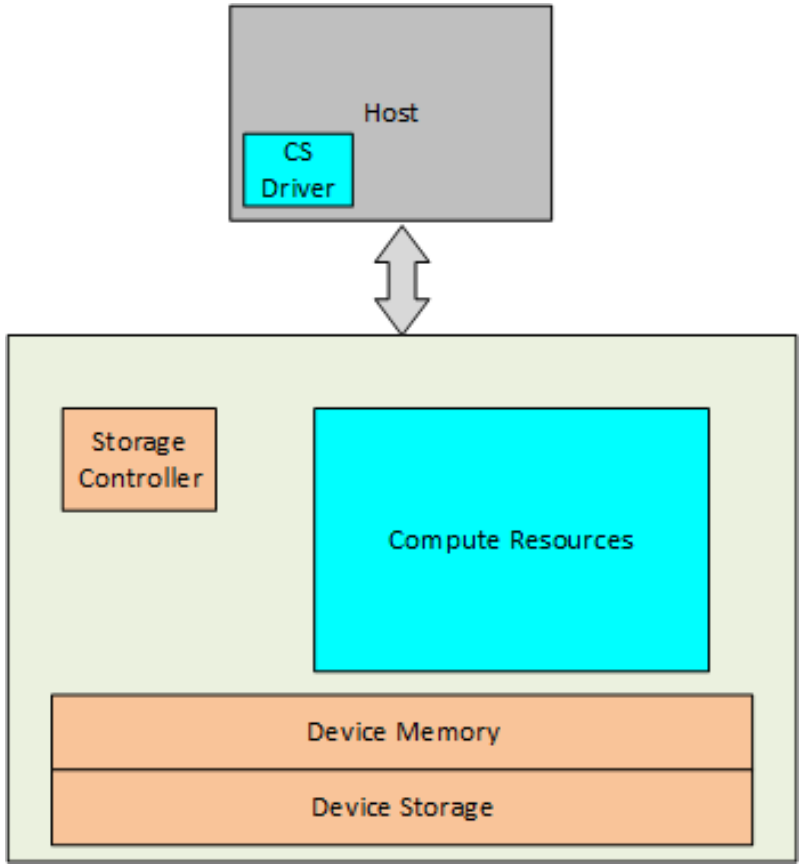
More SDXI ?

- Come join the session “Smart Data Accelerator Interface: Use Cases, Proof Points, v1.1 and beyond”
  - 2:30 – 2:55pm, Cypress



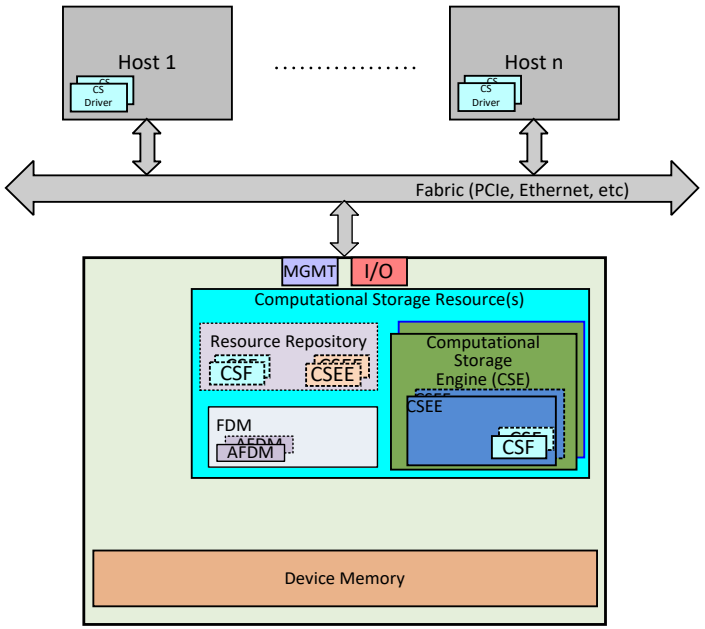
What is Computational Storage?

Computation coupled to storage, offloading host processing or reducing data movement



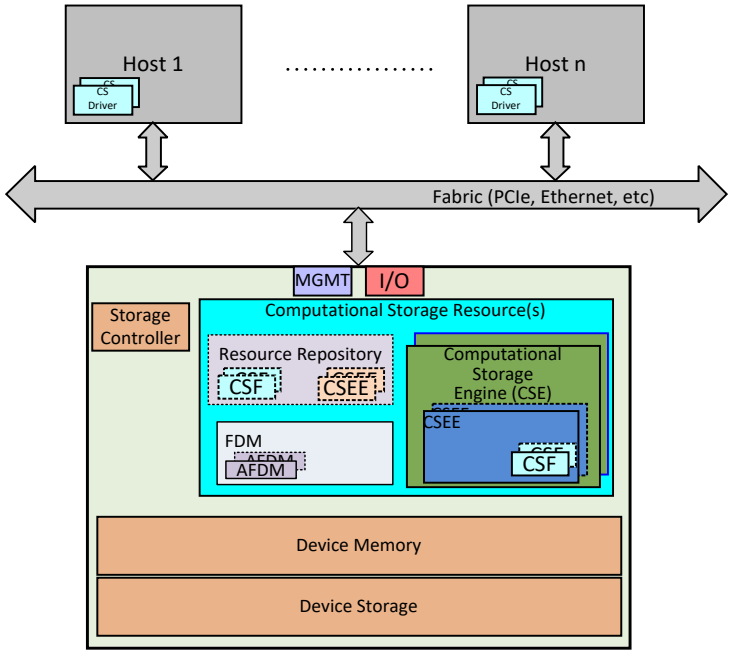
Computational Storage Architecture

Computational Storage Processor



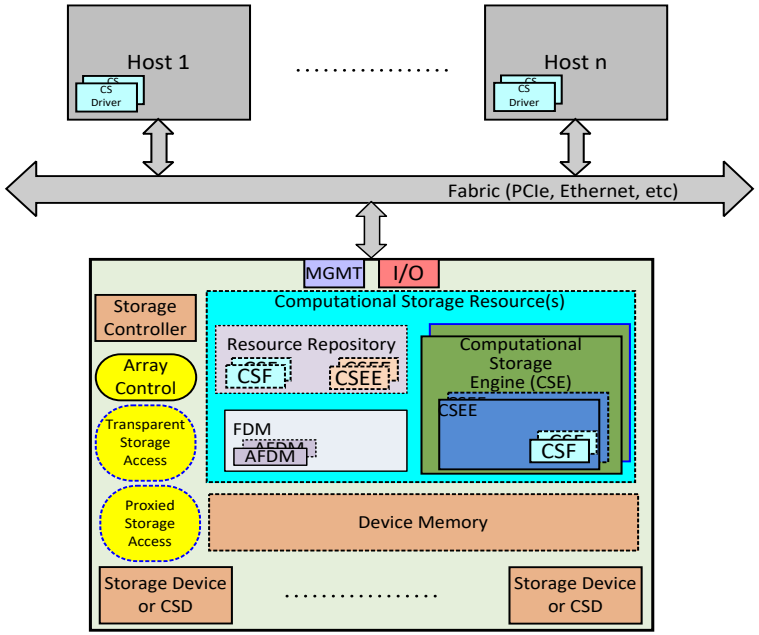
Computational Storage Processor (CSP)

Computational Storage Drive



Computational Storage Drive (CSD)

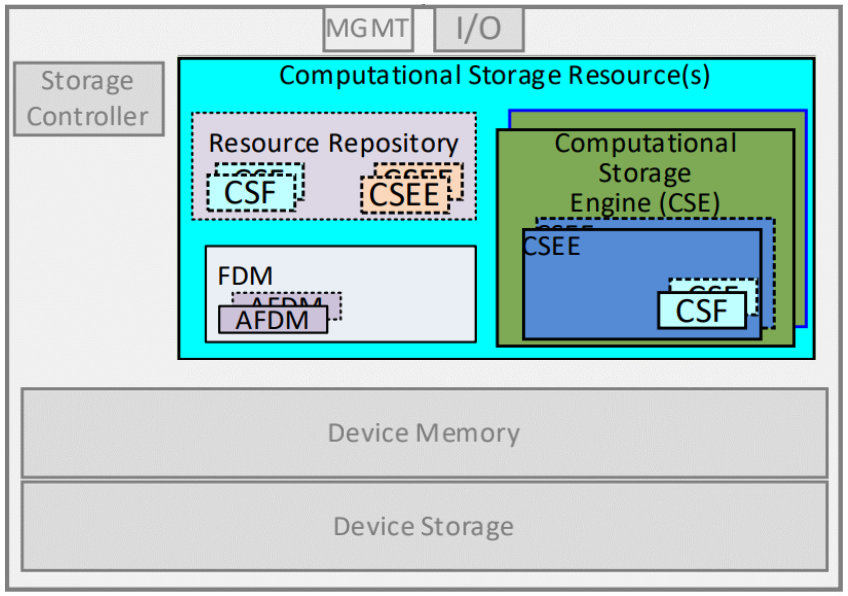
Computational Storage Array



Computational Storage Array (CSA)

CSx = Computational Storage **Device** – CSP or CSD or CSA

A Deeper Dive of the CSx Resources



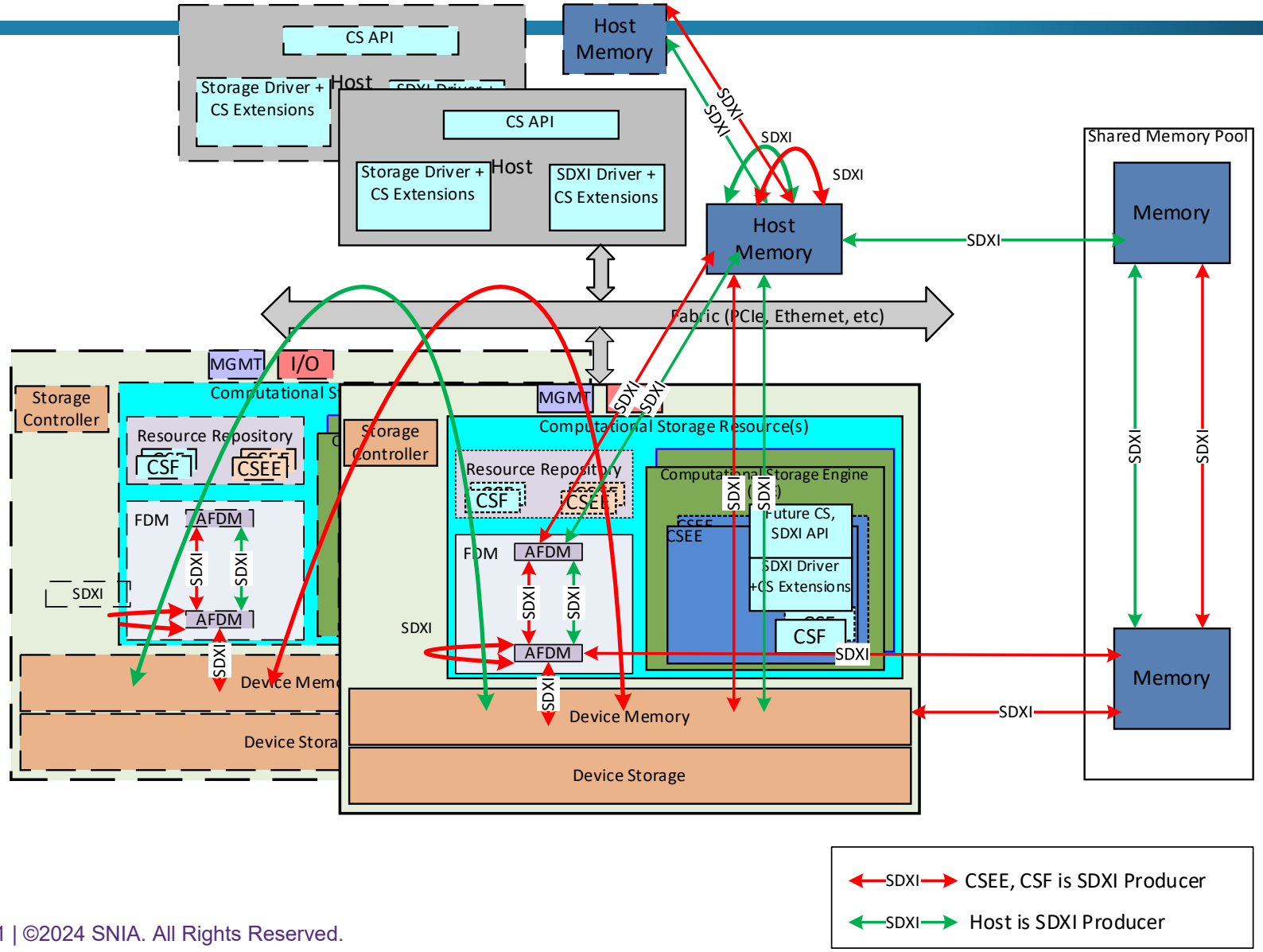
Computational Storage Drive (CSD)

- **CSR -** Computational Storage Resources are the resources available in a CSx necessary for that CSx to store and execute a CSF
- **CSF -** A Computational Storage Function is a set of specific operations that may be configured and executed by a CSE in a CSEE
- **CSE -** Computational Storage Engine is a CSR that is able to be programmed to provide one or more specific operation(s)
- **CSEE -** A Computational Storage Engine Environment is an operating environment space for the CSE
- **FDM -** Function Data Memory is device memory that is available for CSFs to use for data that is used or generated as part of the operation of the CSF
- **AFDM -** Allocated Function Data Memory is a portion of FDM that is allocated for one or more specific instances of a CSF operation
- **Resource Repository –** Resources that are available but not activated

SNIA SDXI+CS Subgroup

- What is the Subgroup
  - The CS TWG and SDXI TWG collaborating to combine CS and SDXI
- 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

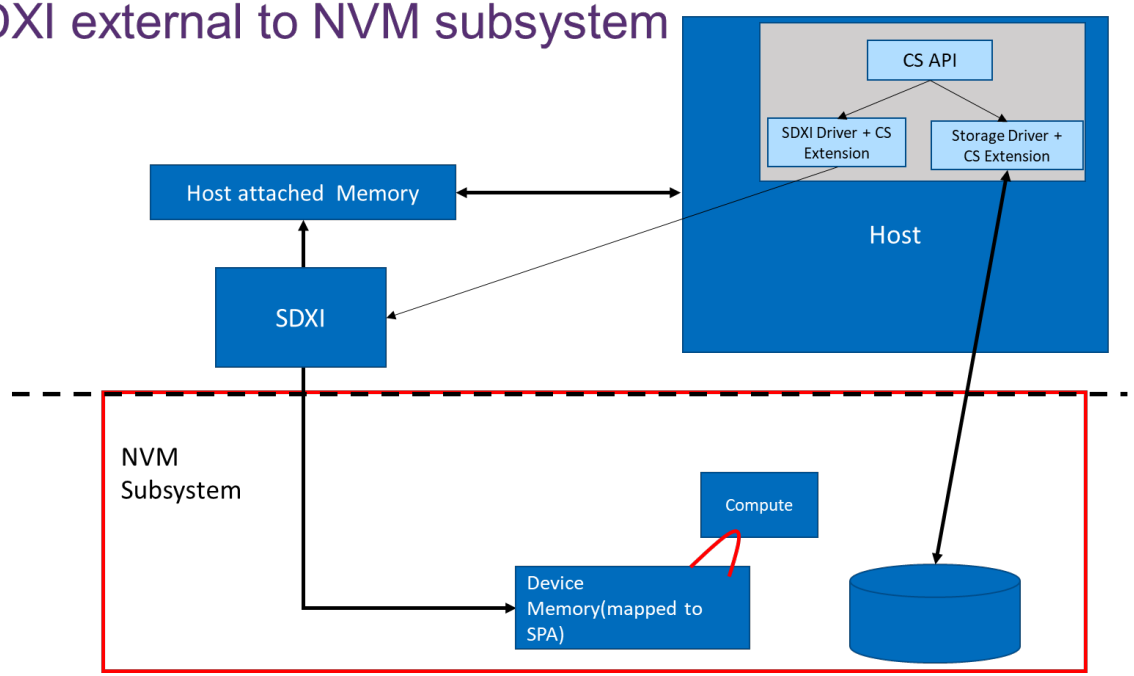
Combined SDXI+CS Architecture



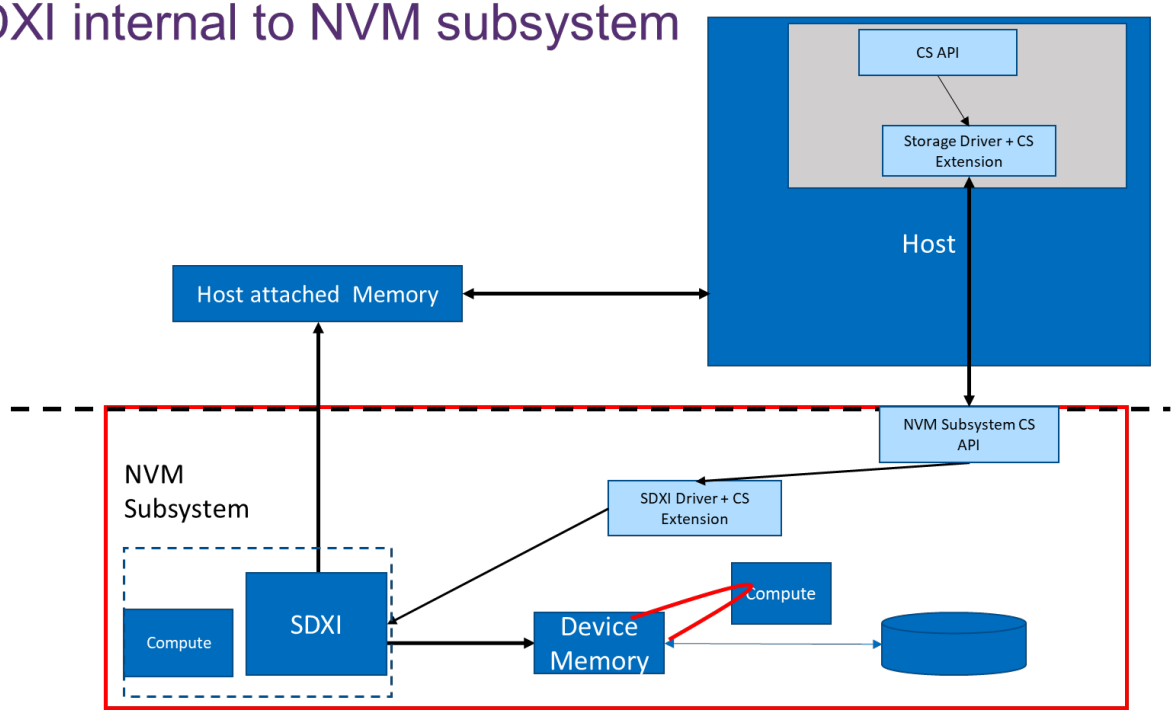
- SDXI used for data movement with Computational Storage used for compute
- Multiple SDXI producers in a CS Architecture
- SDXI enables data movement across multiple AFDM regions

SDXI and NVM Subsystem

SDXI external to NVM subsystem



SDXI internal to NVM subsystem

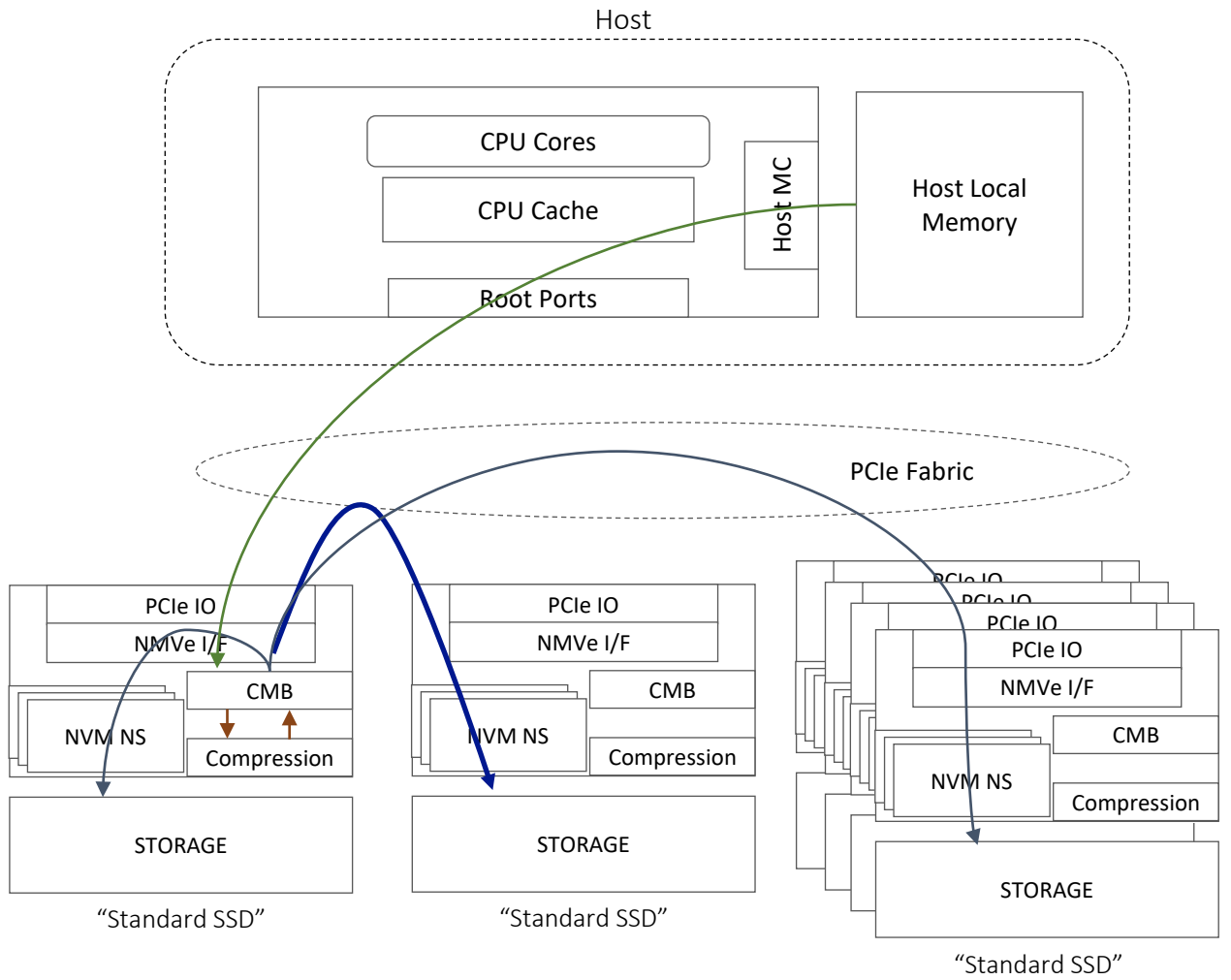


Compression Use Case

- There are many different compression algorithms
- The industry lacks a standard that enables more innovations in compression algorithms
  - Custom drivers, custom implementations do not enable general purpose use of compression accelerators
- Compression is a well-known and frequently requested CSF
- SDXI v1.1 is working towards standardizing memory to memory transformations
  - Compression is one candidate operation under consideration
  - SDXI enabled compression could be the CSF
    - Move data from internal memory to another internal memory, compressing on the movement

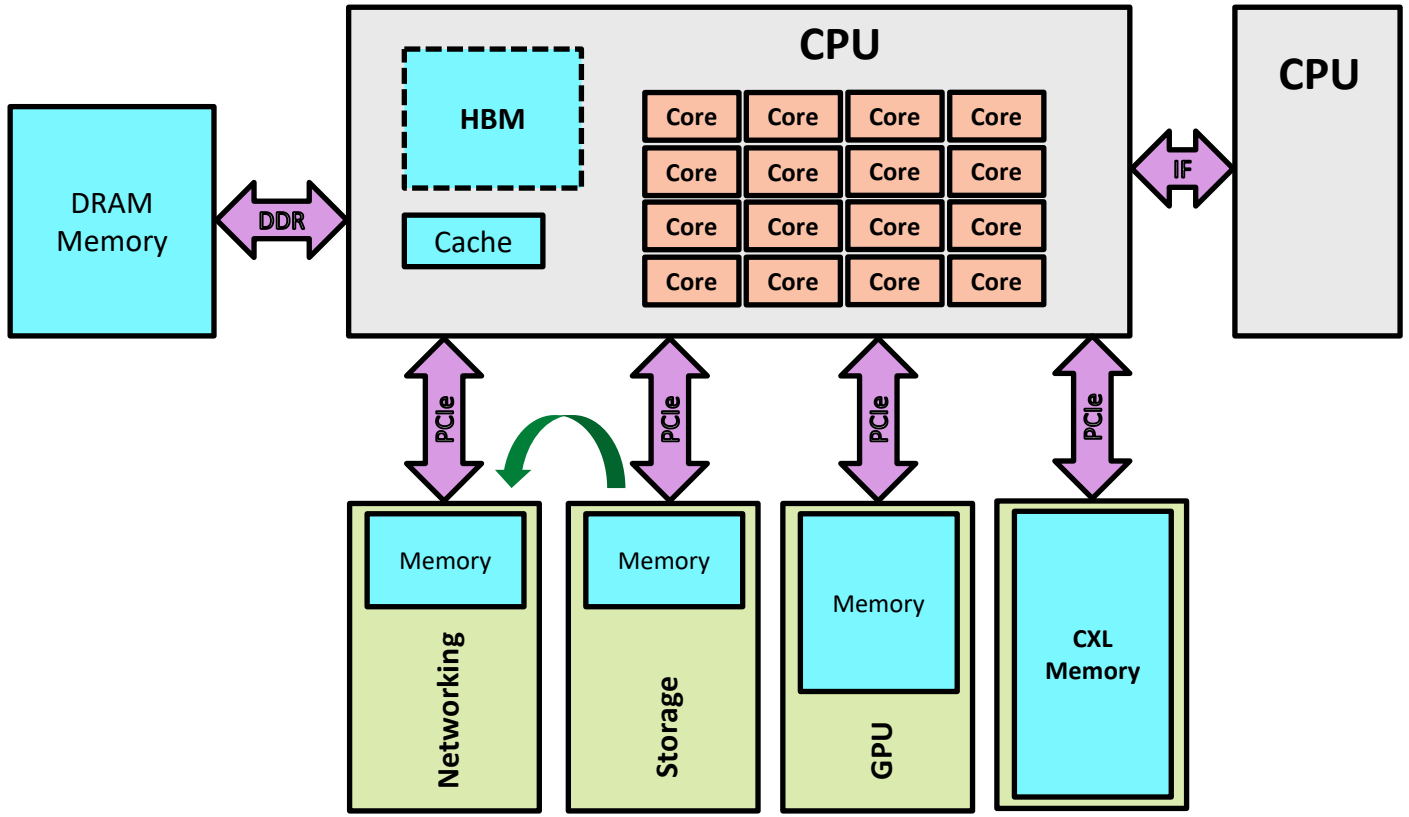
NVMe – Compression

- Trying to determine ways to make SDXI operations more versatile
- Allow SDXI operations to be connected to NVMe commands
- Still work to do to determine what that would look like
- Very early ideation efforts of SDXI enablement



Peer-to-Peer Use Case

- Data is P2P moved from a storage device to a networking device for transmission without moving to a host buffer
  - SDXI moves the data from the storage device to the networking device, bypassing the host



Summary and Call to Action

- SNIA is developing memory data movement (SDXI) and Computational Storage
  - SDXI and CS are members of the SNIA Accelerate pillar
- These SNIA initiatives can help NVMe workloads with data acceleration
- System view is important to find the right fit for various data acceleration and computation requirements
  - Data acceleration for AI requires a system view optimized with innovation from group collaboration
- Call to action:
  - Come join the SDXI + CS Subgroup and contribute to the collaboration

SDC 24

Panel Discussion

And Questions

Please take a moment to rate this session.

Your feedback is important to us.

