

SNIA COMPUTE + MEMORY
+ STORAGE SUMMIT

Architectures, Solutions, and Community
VIRTUAL EVENT, APRIL 11-12, 2023

Thinking Memory

Mats Öberg
Associate Vice President
Marvell



Forward-looking statements

Except for statements of historical fact, this presentation contains forward-looking statements (within the meaning of the federal securities laws) including, but not limited to, statements related to market trends and to the company's business and operations, business opportunities, growth strategy and expectations, and financial targets and plans, that involve risks and uncertainties. Words such as "anticipates," "expects," "intends," "plans," "projects," "believes," "seeks," "estimates," "can," "may," "will," "would" and similar expressions identify such forward-looking statements. These statements are not guarantees of results and should not be considered as an indication of future activity or future performance. Actual events or results may differ materially from those described in this presentation due to a number of risks and uncertainties.

For factors that could cause Marvell's results to vary from expectations, please see the risk factors identified in Marvell's Annual Report on Form 10-K for the fiscal year ended January 28, 2023, as filed with the SEC on March 9, 2023, and other factors detailed from time to time in Marvell's filings with the SEC. The forward-looking statements in this presentation speak only as of the date of this presentation and Marvell undertakes no obligation to revise or update publicly any forward-looking statements.

Compute where the data is

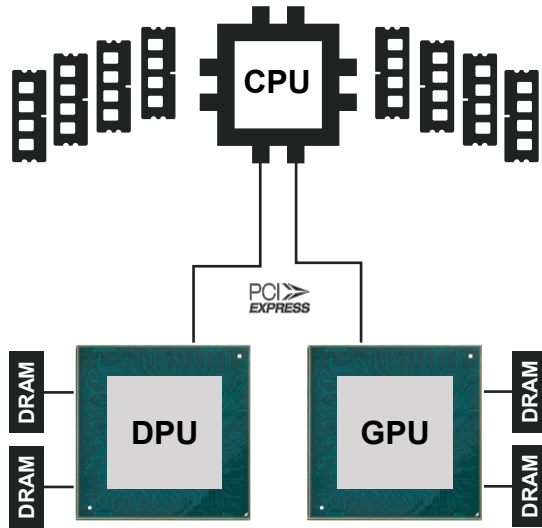
- Offload compute
- Reduce data movements
- Efficiency
 - Power
 - Performance
 - Cost



CXL memory with compute

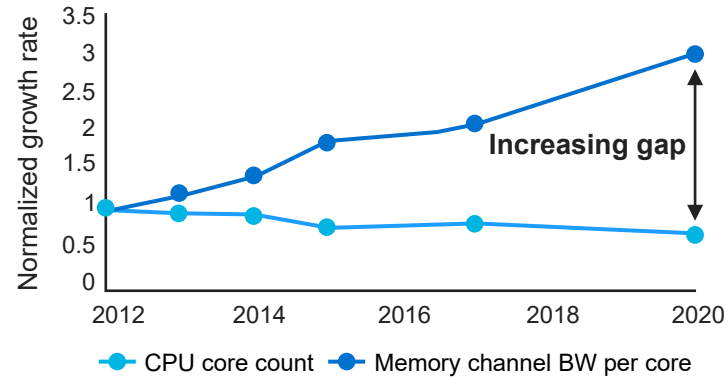
Cloud data center memory challenges

Memory tied-down to xPUs



Cannot share

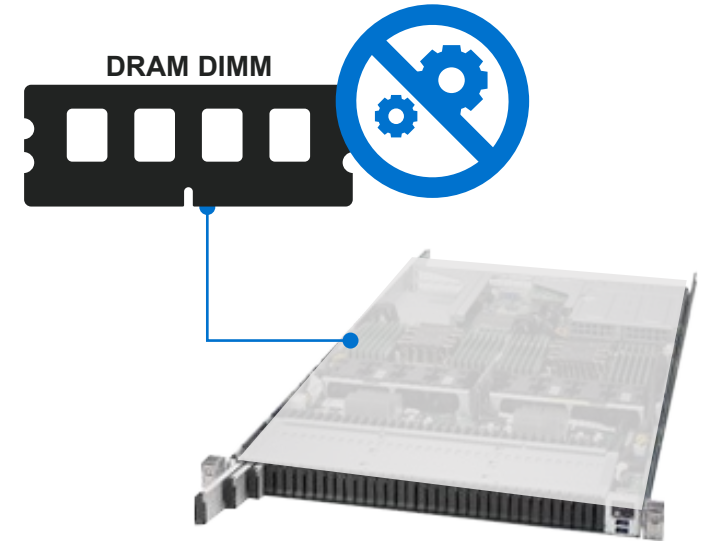
Bandwidth per core declining



Source: Meta, OCP Summit Presentation Nov 2021

Degrades efficiency

No data computation off-loading

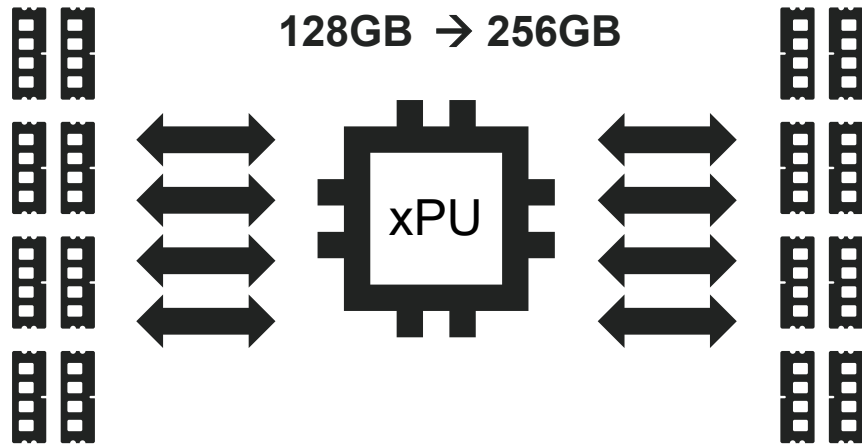


Limits performance

CXL is poised to address these issues

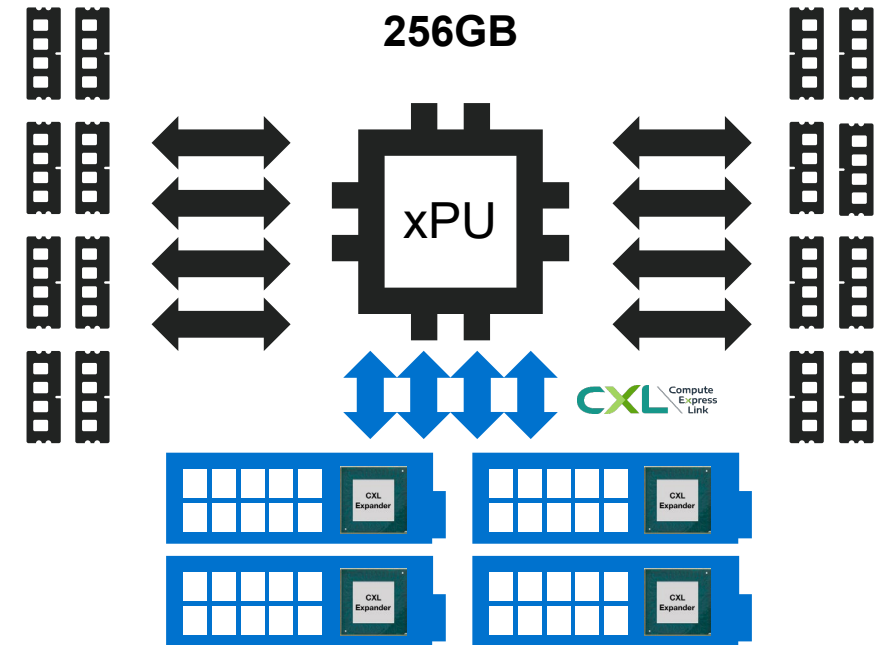
Memory expansion with CXL

Today



Capped performance

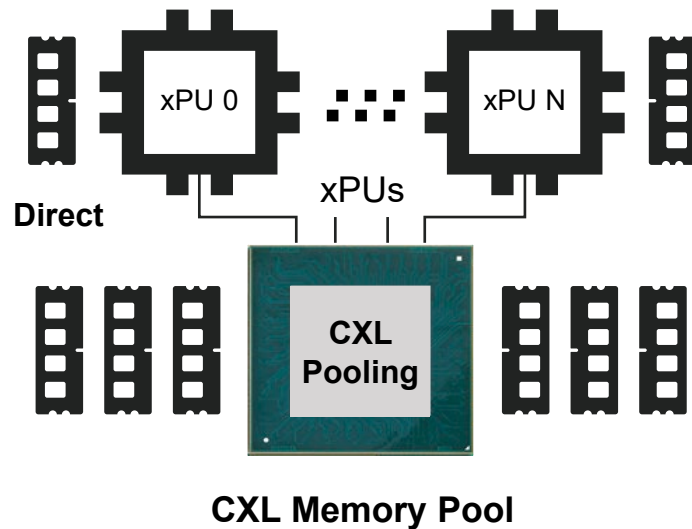
Future



Scalable performance

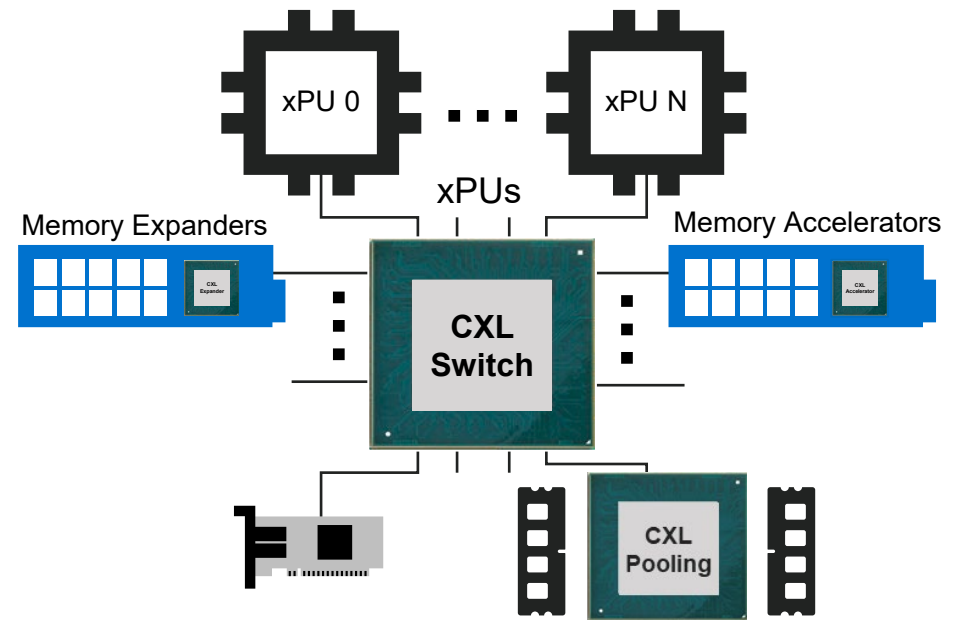
Sharing resources with CXL

CXL pooling



- Pool memory across multiple xPUs
- Rescue under-utilized DRAM
- Scale memory independent of xPUs

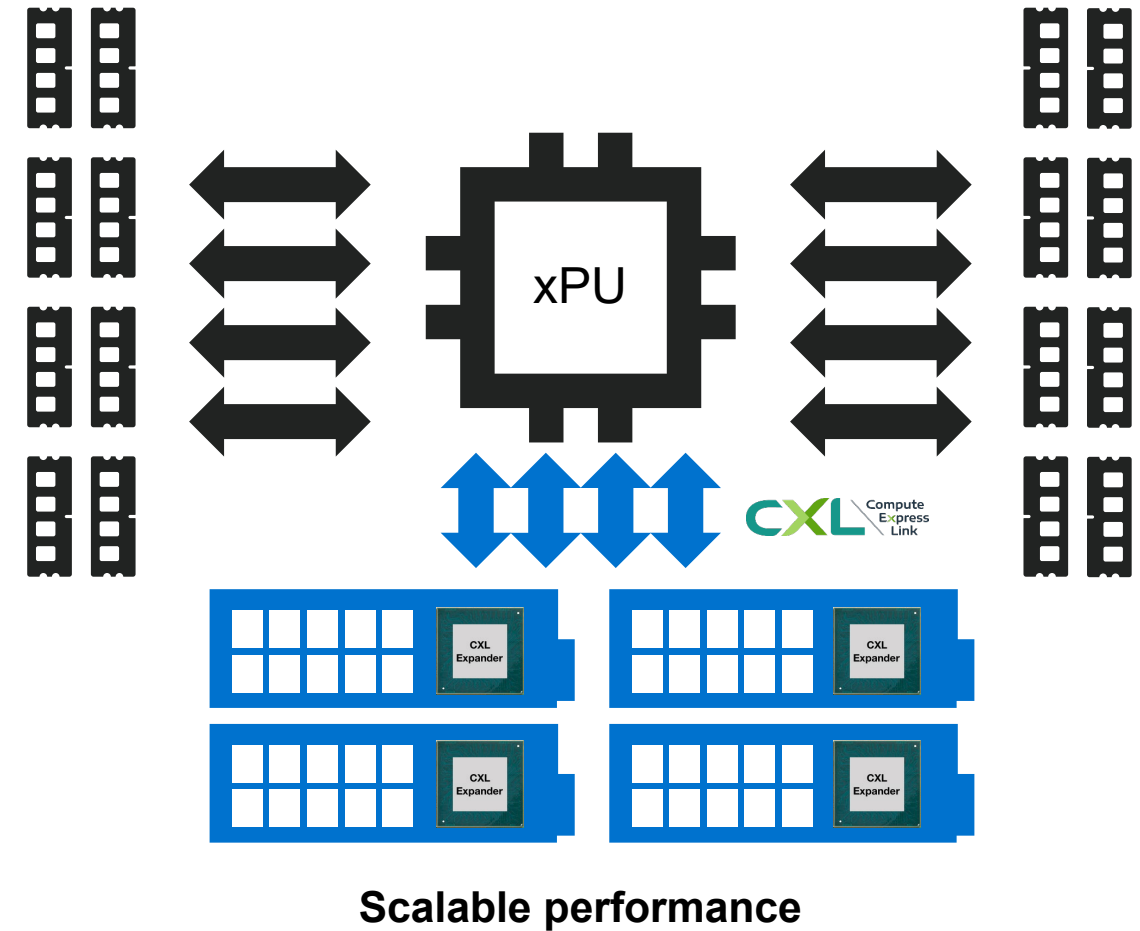
CXL switch



- Flexible to connect resources into fabric
- Scalable, serviceable
- Enables fully composable infrastructure

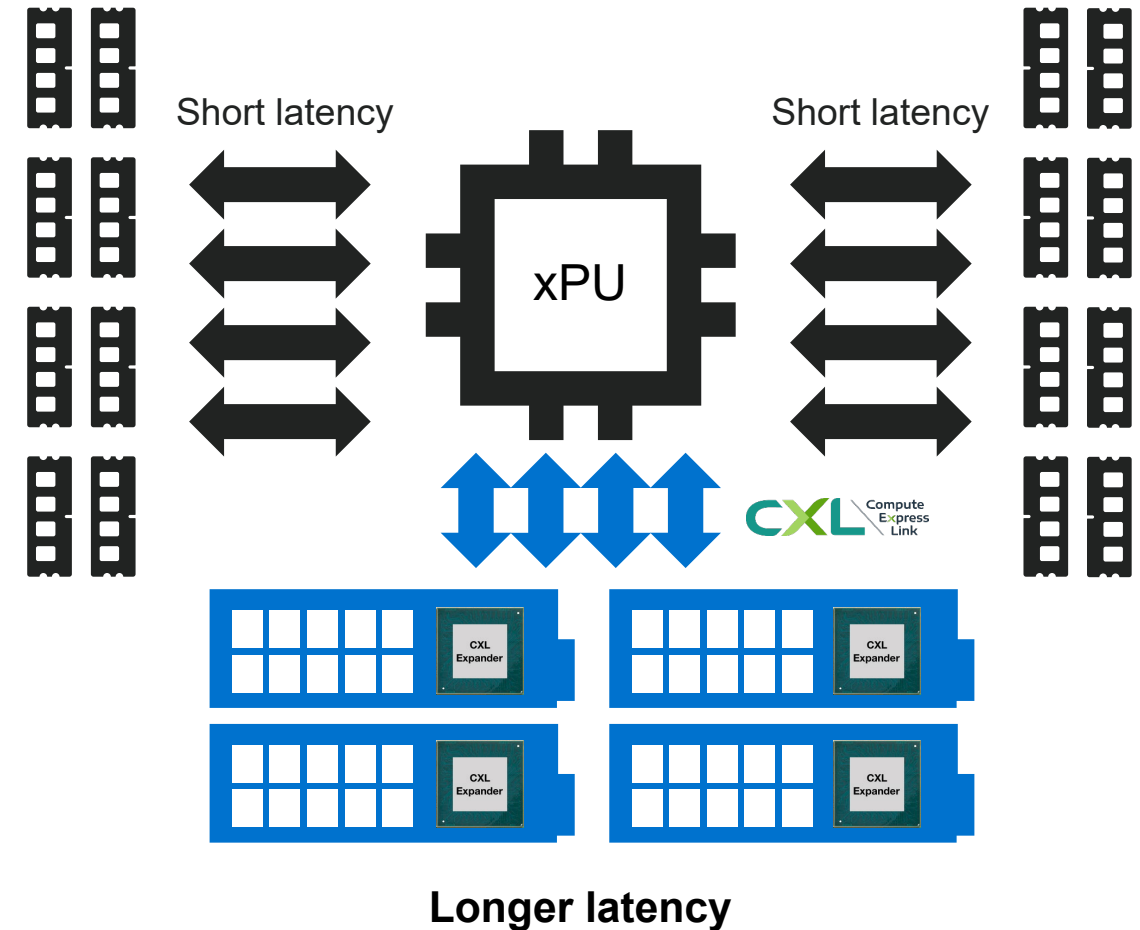
CXL memory expansion

- Low latency – practical
- More capacity
- Higher bandwidth



CXL memory expansion – additional memory tier(s)

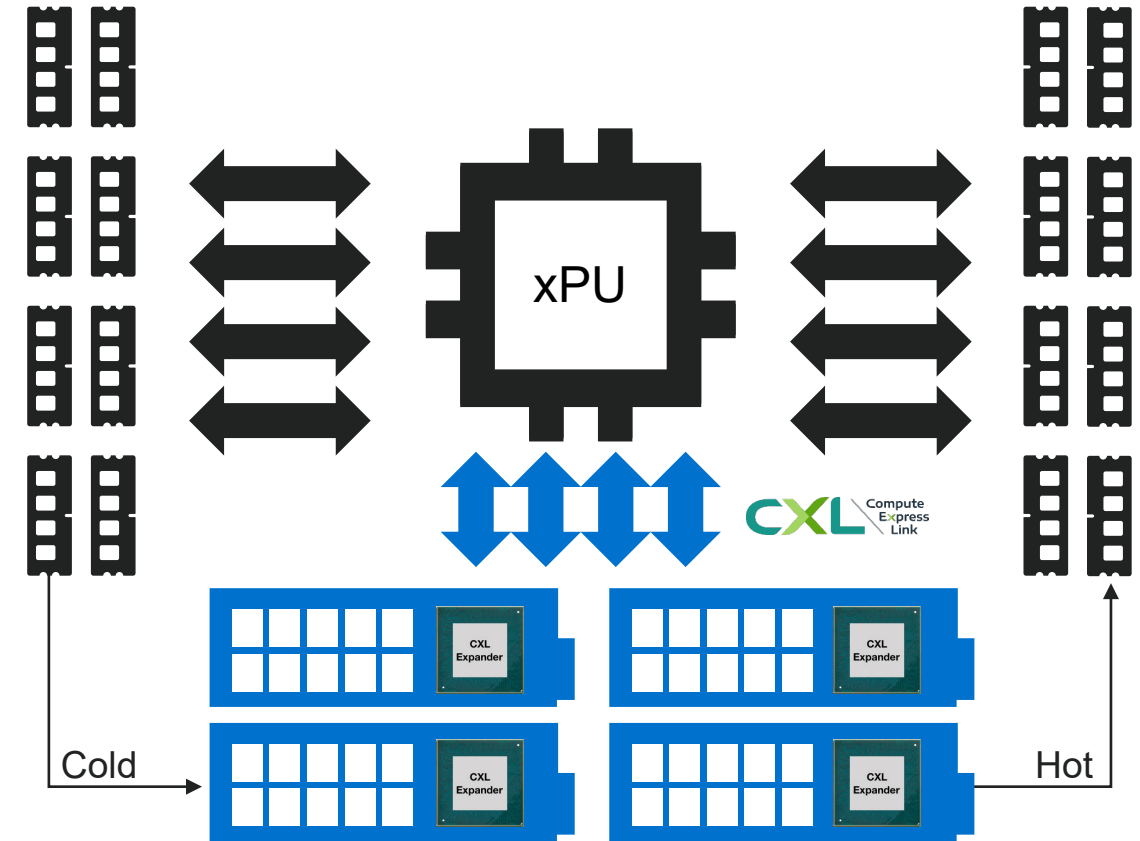
- Any latency degrades performance
- Tiered memory
- Solution
 - “Hot” memory native
 - “Cold” memory CXL
 - e.g., Transparent Page Placement^[1]



[1] Maruf, et al. “TPP: Transparent Page Placement for CXL-Enabled Tiered Memory”, <https://arxiv.org/abs/2206.02878>

Transparent page placement

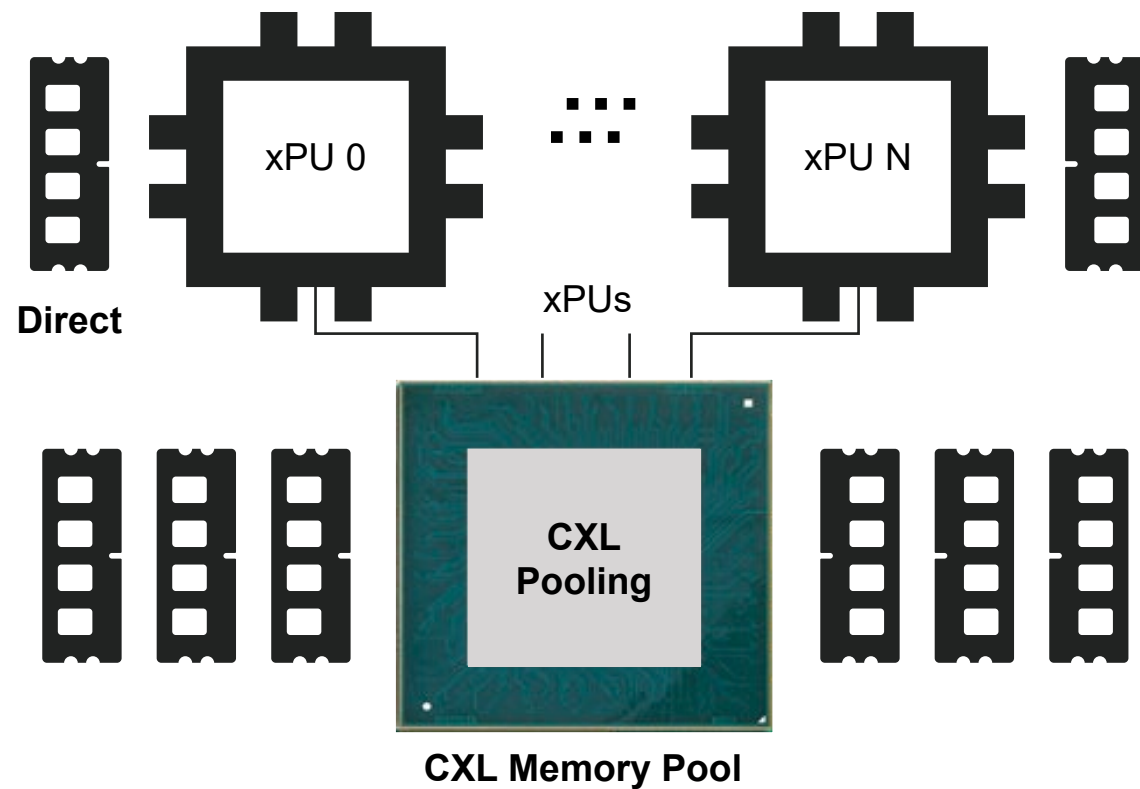
- TPP paper^[1]
 - Warm and cold pages
 - Relatively stable for long time
 - Anonymous memory hotter
 - File backed memory colder
- Operation:
 - Demote cold pages to CXL
 - Promote hot pages to local



[1] Maruf, et al. "TPP: Transparent Page Placement for CXL-Enabled Tiered Memory", <https://arxiv.org/abs/2206.02878>

Hot/cold page detection

- TPP paper – Linux kernel
- Large CXL memory
 - high kernel load
- Memory pooling – less visibility
- Hotness tracker in CXL controller
 - Most accessed memory
 - Address range tracking
- OCP CMS proposal in the works



- Pool memory across multiple xPUs
- xPU i memory access unknown to xPU j

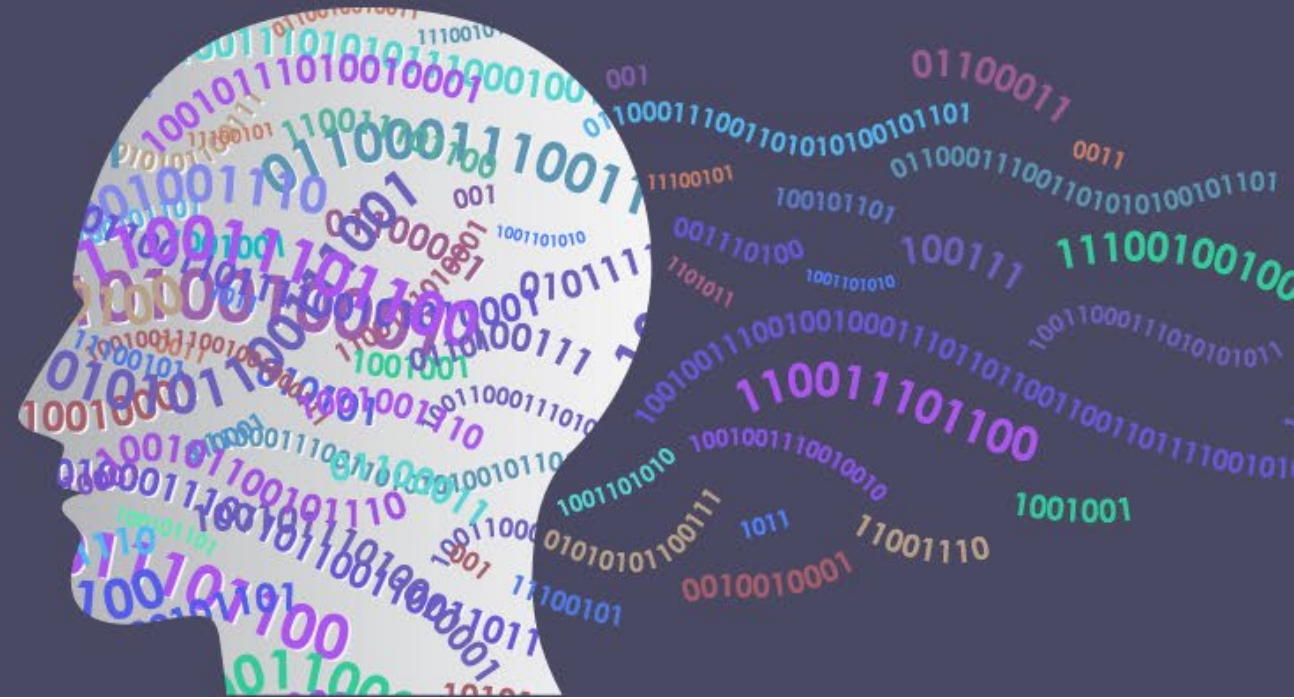
Computational memory

- Hotness tracking – simple compute offload
 - Improves CPU utilization
 - Transparent to user
- CXL memory compression
 - Compressed swap space in memory
 - Compress pages in CXL pool – free up memory
- Standardize to accelerate deployment



COMPUTE + MEMORY + STORAGE SUMMIT

Architectures, Solutions, and Community
VIRTUAL EVENT, APRIL 11-12, 2023

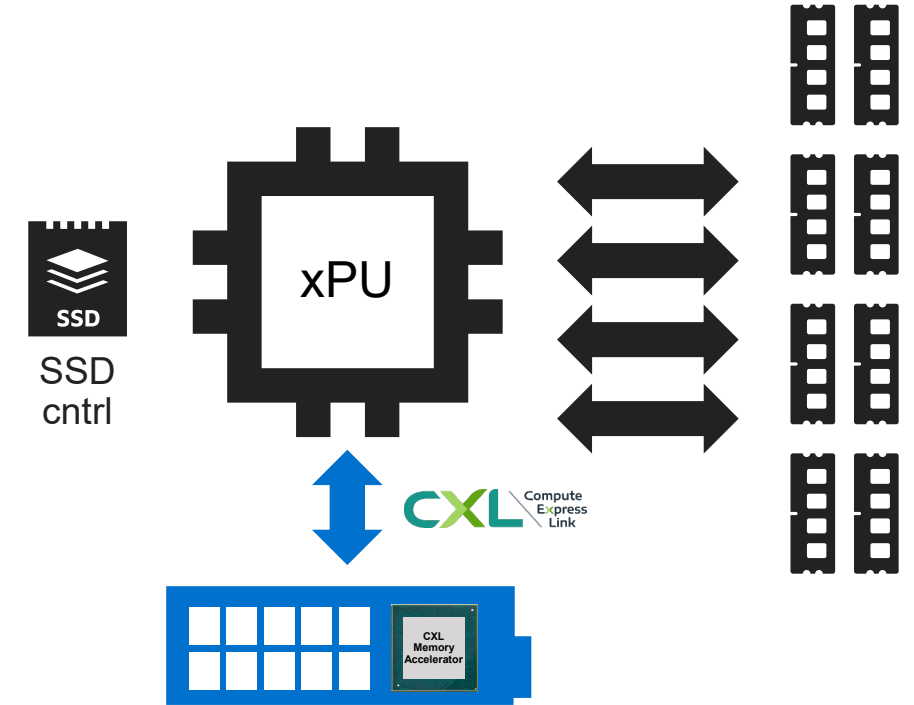


General purpose computational memory

Leverage standardization

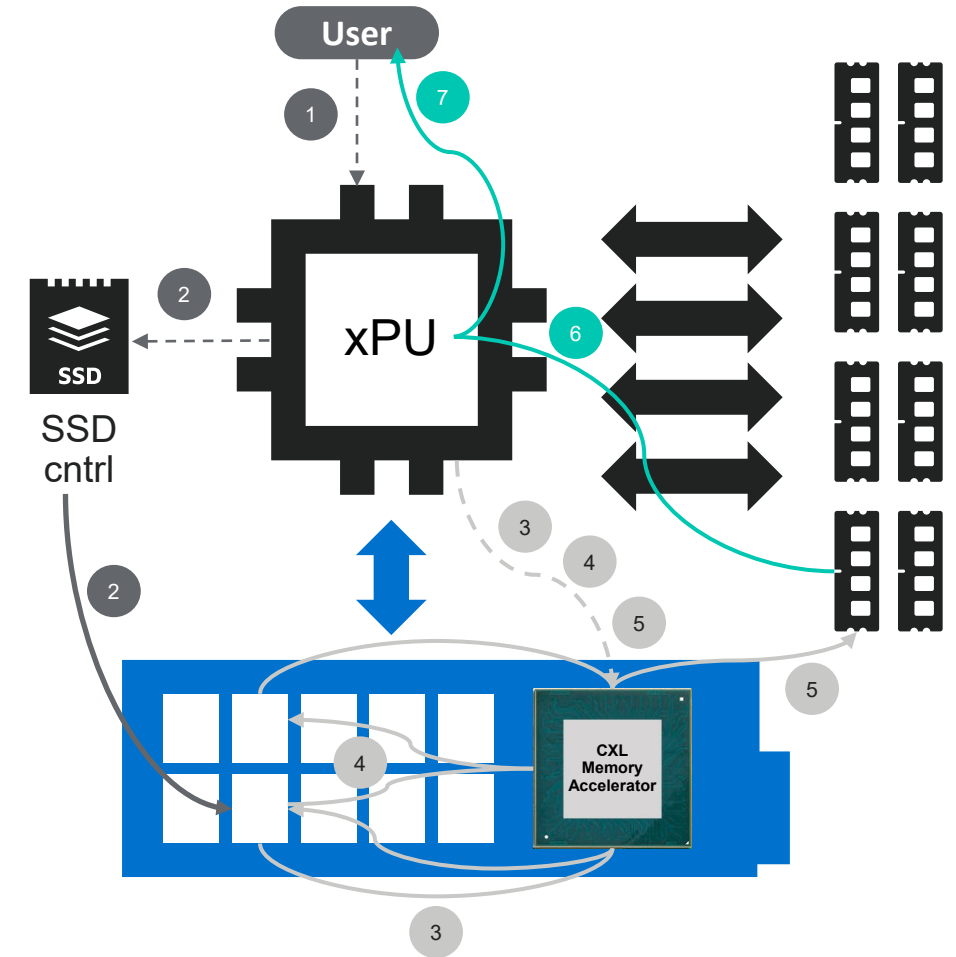
Computational memory

- Offload operations to CXL device
 - Compression, encryption, hashing
 - Filtering, regexp
- Reduces data movement
- Frees up CPU for other operations



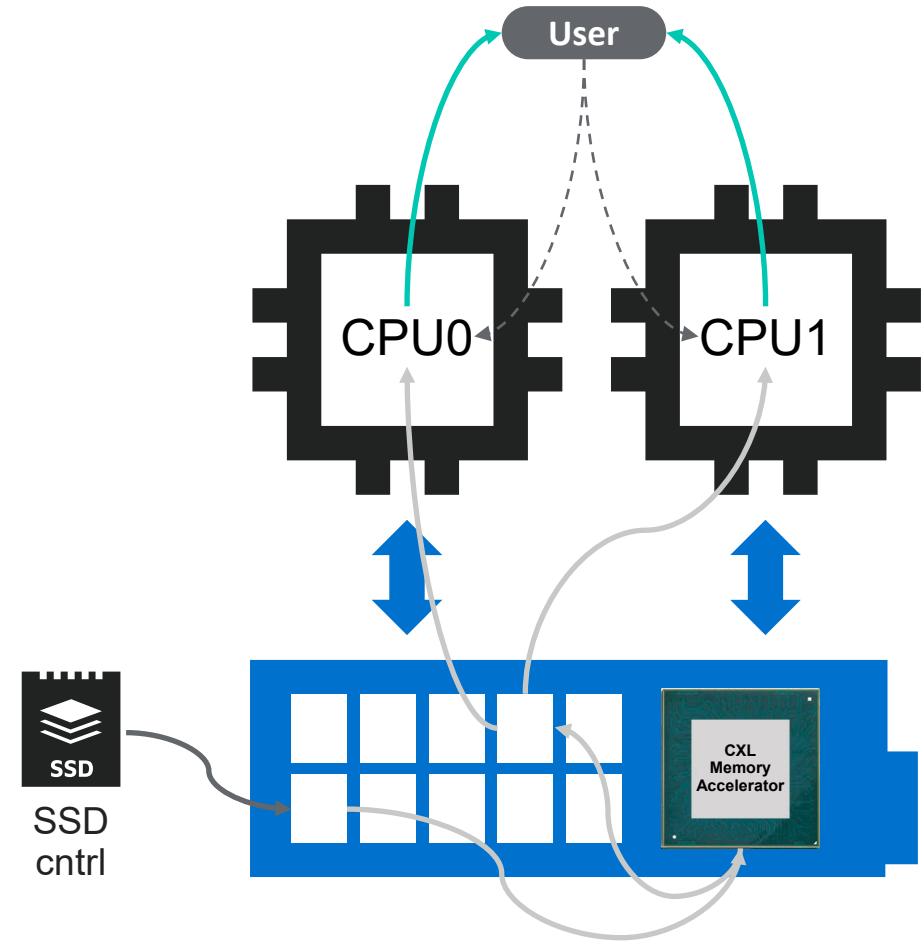
Computational memory use case

- User:
 1. “Calculate total sales in May 2022”
- CPU:
 2. Read data from SSD into CXL memory
- CXL device:
 3. Decrypt the data in CXL memory
 4. Decompress data in CXL memory
 5. Filter data for records with date “May 2022”, place records in direct attached DRAM
- CPU:
 6. Calculate total sales
 7. Return results to user

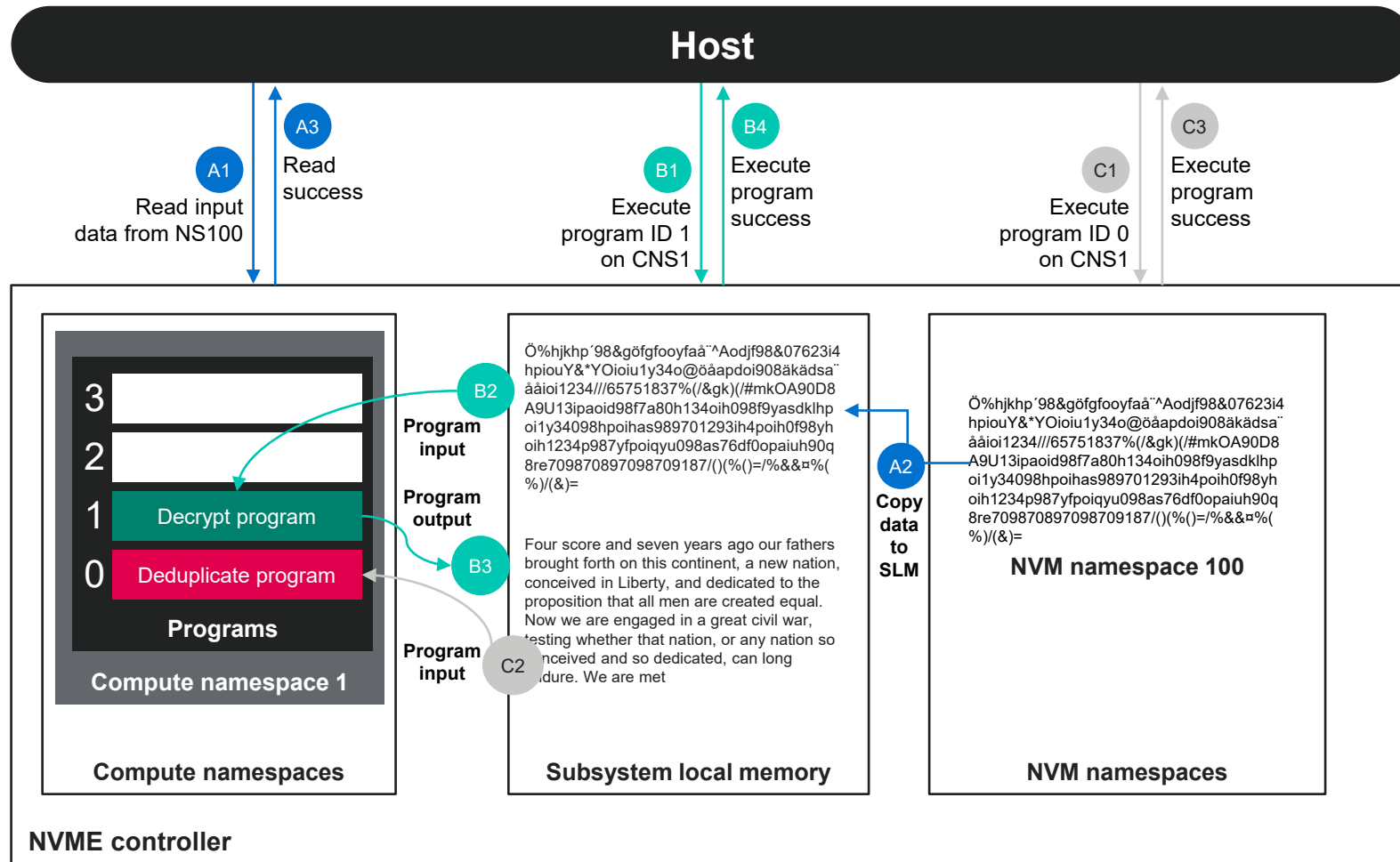


Compute on shared memory

- User:
 1. “CPU0: Get May 2022 sales data to shared CXL memory. Calculate total sales in May 2022”
 2. CPU1: “Calculate average age of buyers in May 2022”
- CPU0:
 2. Read data from SSD into CXL memory
- CXL device:
 3. Filter data for records with date “May 2022”, place records in shared CXL memory
- CPU0:
 6. Calculate total sales
 7. Return results to user
- CPU1:
 6. Calculate average age
 7. Return results to user



Flow similar to computational storage



Example of computational storage shared at FMS2022

Flow steps

- A** Read encrypted data into SLM
- B** Execute program ID 1 on CNS1
- C** Execute program ID 0 on CNS1

Repeat for next block of data

Leverage SNIA computational storage

- SNIA computational storage
 - Moves data from storage to memory
 - Computes on data in memory
- Near memory compute in CXL
 - Data already in memory
 - Computational Storage Processor with host accessible memory
- Adapt SNIA CS architecture and API for computational memory

Call to action

- Compute in CXL memory device as a tool to reduce data movement
- Processing of shared data in memory provides value for all users of data without moving data from device, and moving processed data back to the device
- Adapt SNIA Computational Storage to Near Memory Compute applications



COMPUTE + MEMORY + STORAGE SUMMIT

Architectures, Solutions, and Community
VIRTUAL EVENT, APRIL 11-12, 2023



Please take a moment to rate this session.

Your feedback is important to us.