Compute Express Link (CXL)

The Interconnect Revolution

Sameer Ahmad
sameerahmad@cse.iitb.ac.in

Raja Gond
rajagond@cse.iitb.ac.in

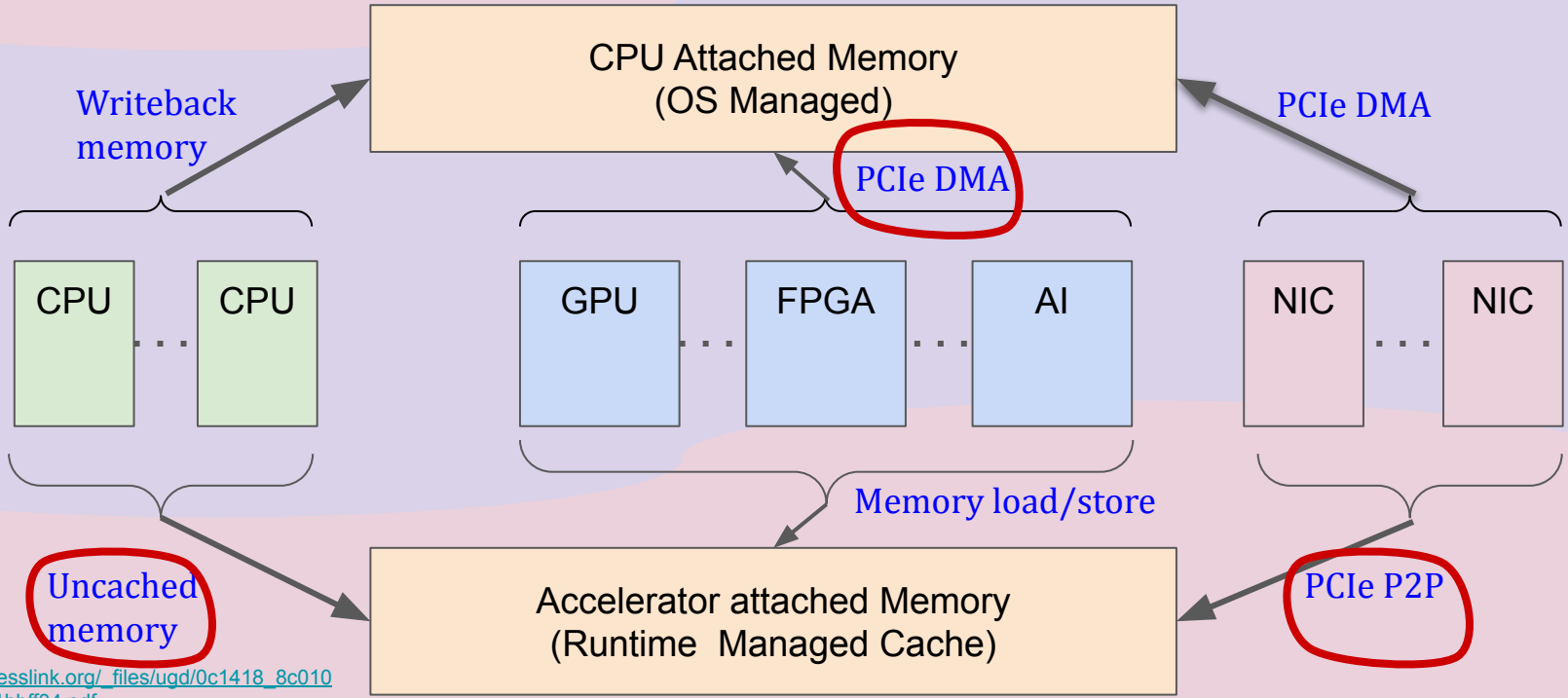
Seminar - RnD - Spring 2023

Outline

1. Life before CXL
2. Why CXL?
3. CXL-101
4. Use Cases
5. Conclusion

Life Before CXL

- With *PCIe (Peripheral Component Interconnect Express) Only*



https://www.computeexpresslink.org/_files/ugd/0c1418_8c010eed2bac49438e2168ec71bbff34.pdf

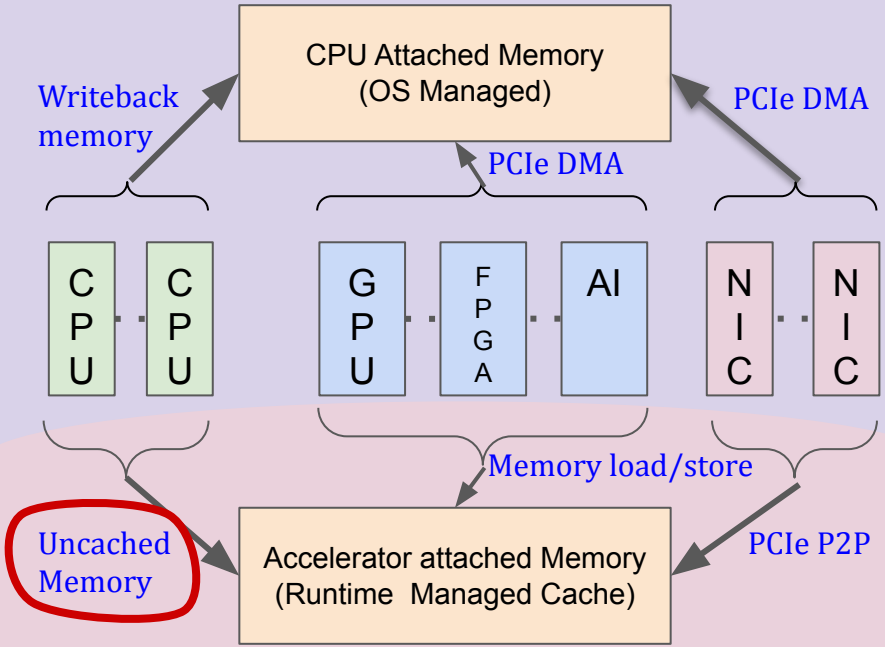
Life Before CXL

- With *PCIe (Peripheral Component Interconnect Express) Only*

➔ Memory connected to CPU
- **Cacheable**

➔ Memory connected to PCIe device
- **Uncacheable**

➔ Works well for a lot of applications with existing software



Life Before CXL

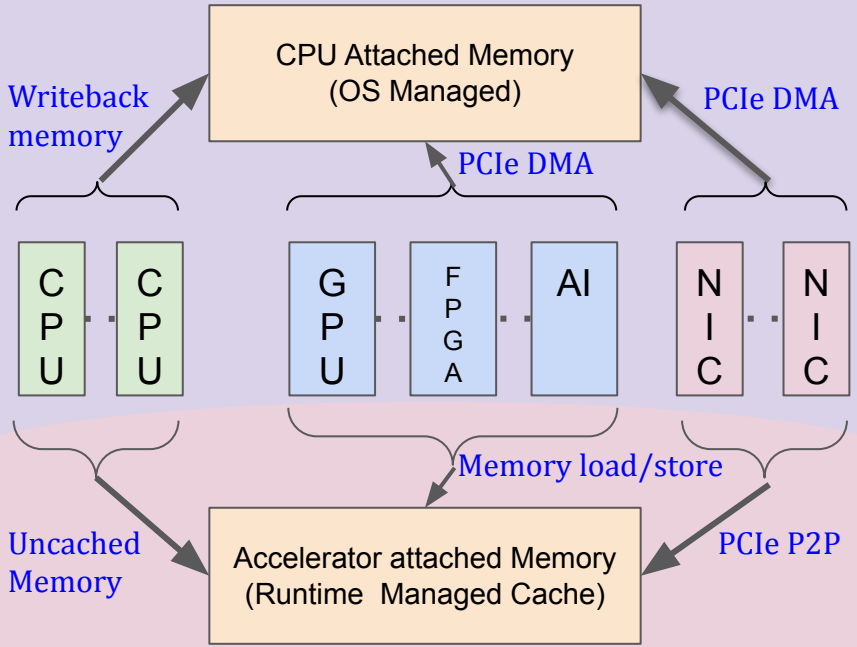
- Challenges with this model

➔ Heterogeneous computing and disaggregation

- **Efficient resource sharing including memory**

➔ Memory Bandwidth and capacity extension on PCIe I/O

- **Memory tiering and different memory types**



Why CXL?

➔ Industry mega-trends are driving demand for faster data processing and next-generation data center performance:

- Proliferation of Cloud Computing
- Growth of Artificial Intelligence and Analytics
- Cloudification of the Network and Edge

Why CXL?

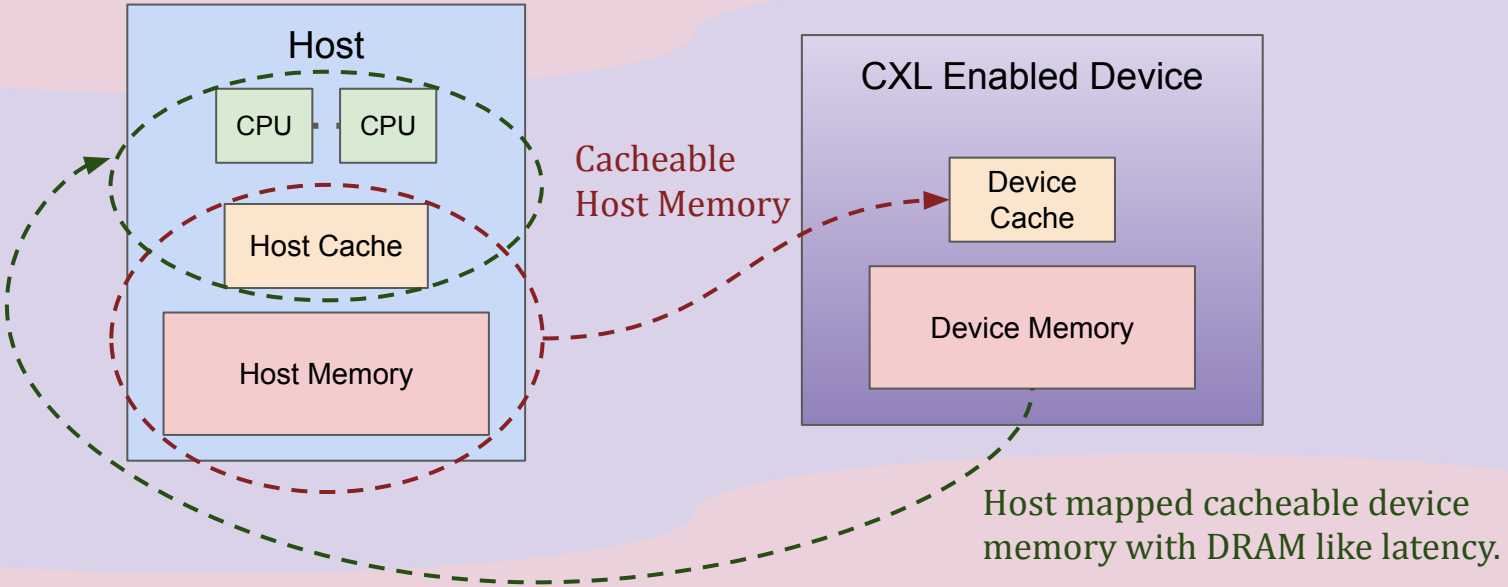


Fig: CXL-enabled environments

Why CXL?

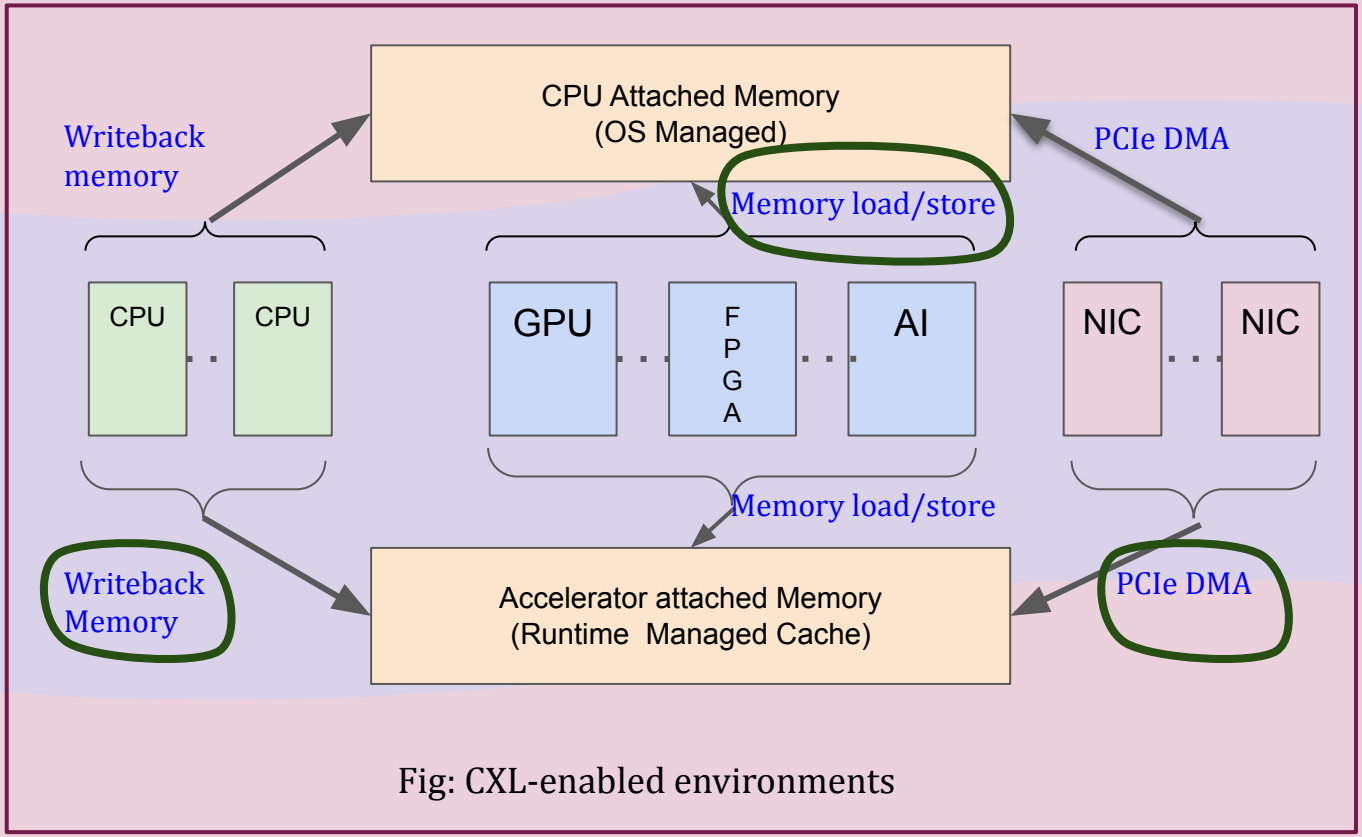


Fig: CXL-enabled environments

Why CXL?

➔ Need a new class of interconnect for heterogeneous computing and disaggregation usages:

- Efficient resource sharing
- Shared memory pools with efficient access mechanisms
- Enhanced movement of operands and results between accelerators and target devices
- Significant latency reduction to enable disaggregated memory

Why CXL?

➔ The industry needs open standards that can comprehensively address next-gen interconnect challenges

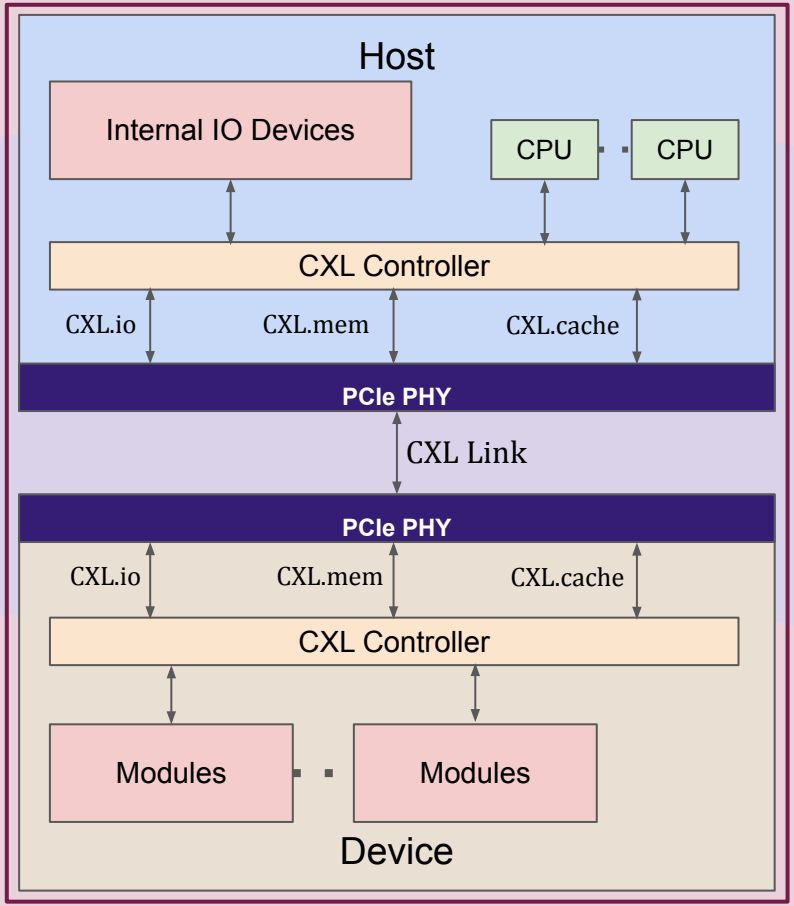
CXL-101

CXL: A new class of open standard interconnect

- Industry Open Standard for High Speed Communications
- 150+ Member Companies
- All CPU, GPU and memory vendors in consortium
- CXL has a bright future and will be a game-changer in the industry!!
- https://www.computeexpresslink.org/

CXL-101

- Processor Interconnect:
  - Open industry standard
  - High-bandwidth, low-latency
  - Coherent interface
  - Leverages PCI Express®
- Targets high-performance computational workloads
  - Artificial Intelligence
  - Machine Learning
  - HPC
  - Comms



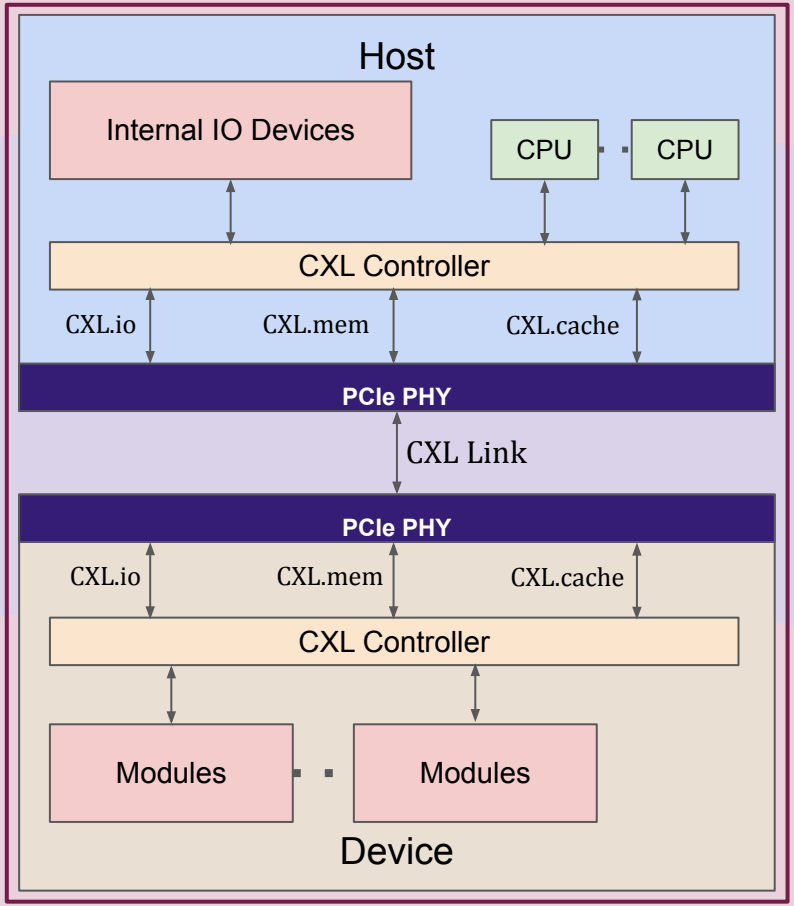
CXL-101

- CXL is a new class of **interconnect** for device connectivity and cache coherent interface using PCIe, enabling memory expansion and heterogeneous memory for disaggregated computing platforms.
- *Disaggregated computing* is a computing architecture that separates compute, memory, and storage resources into distinct physical devices connected by a high-speed network.

CXL-101

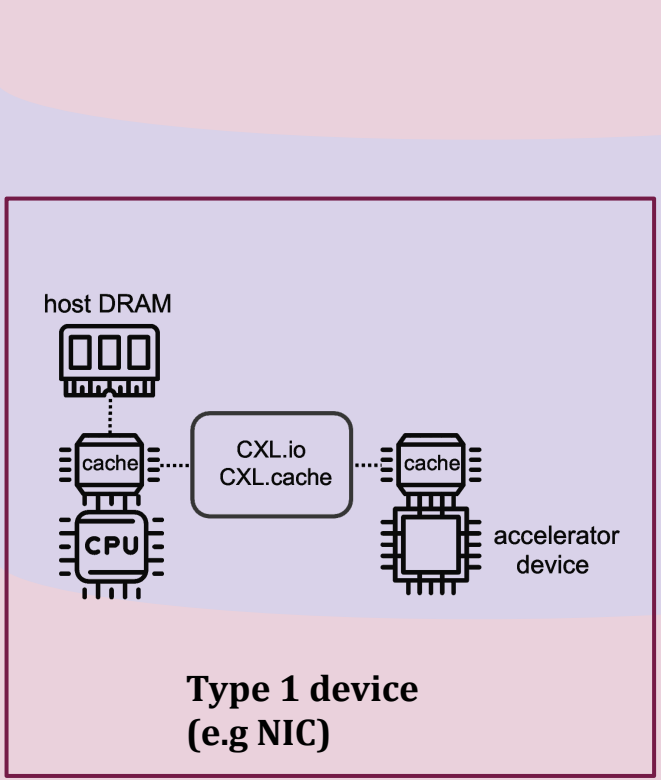
CXL has an alternate **protocol** that runs across the standard PCIe 5.0 physical layer, consisting of three protocols;

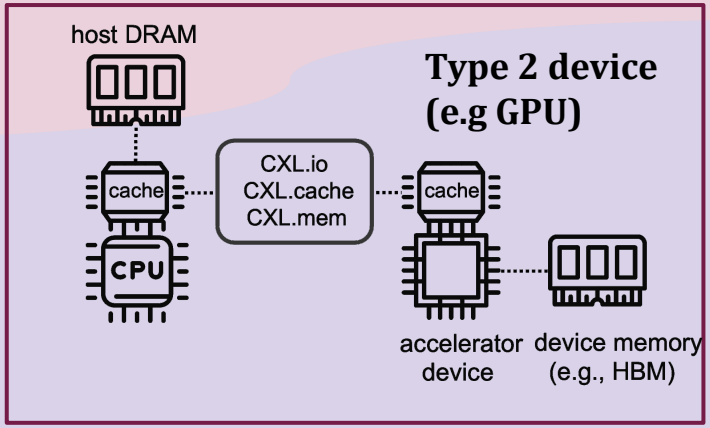
1. **CXL.io** for discovery, configuration, register access, and interrupt.
2. **CXL.cache** for device access to processor memory, and
3. **CXL.memory** for processor access to device attached memory.

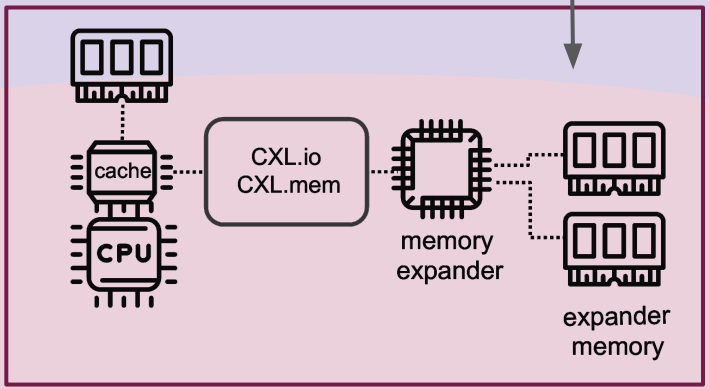


CXL-101

Type of CXL devices



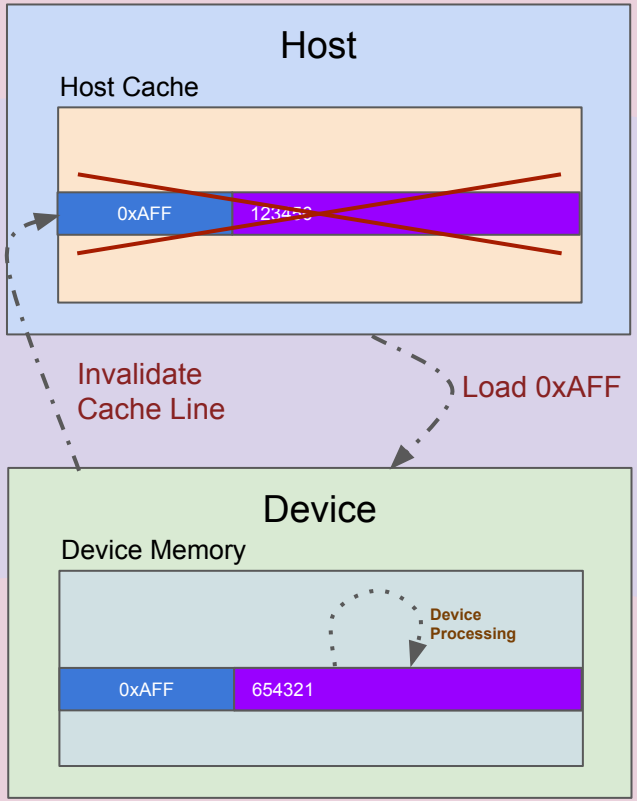




https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10015193

CXL-101

- CXL is a cache coherent interconnect technology.
- All device types of CXL compatible with most existing PCIe devices, including SSDs.
- Even though CXL is built upon PCIe, it basically guarantees that all the caches across different computing complexes in the same CXL hierarchy are coherent.



Time for meme :)

Persistent Memory (Optane Machine)

CXL

Intel decommissioned Optane	
CXL is Future	
CXL hardware not yet available	

Use Cases

- TOOLSET
- Memory Pooling
- Memory Expansion
- Memory Sharing
- Potential works with CXL

TOOLSET

CXL Compute Express Link™ + QEMU + Linux = YES

CXL hardware not yet available

Possible but under development

Emulate CXL devices with **Qemu**

- Also require support from **Linux** Kernel
- Already in active development to support **CXL** devices.
- Load and Store aren't yet supported

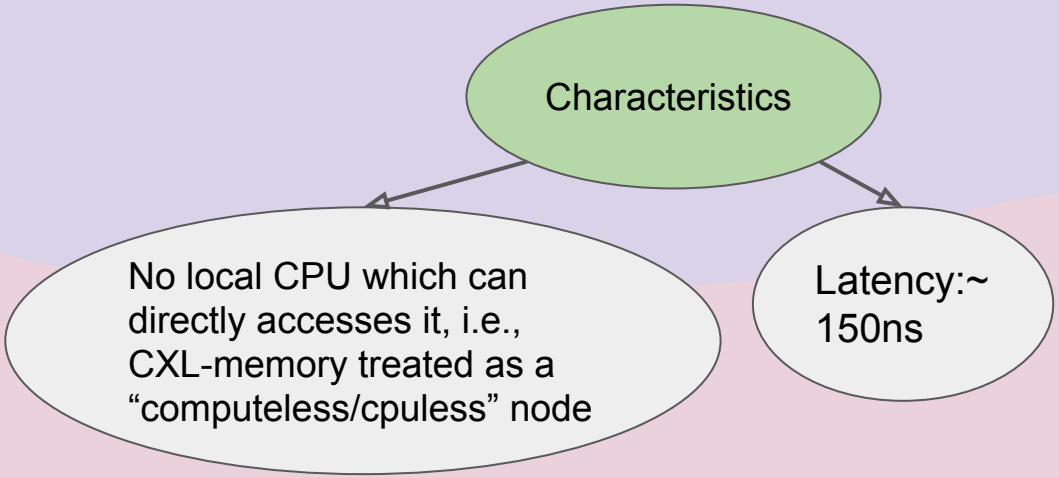
https://www.computeexpresslink.org/

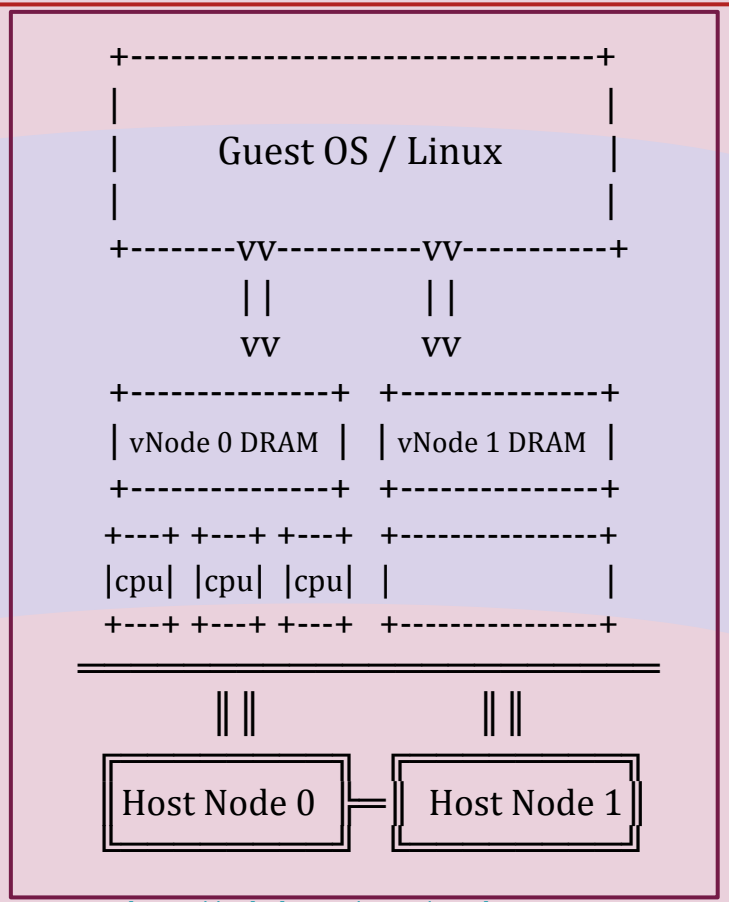
https://www.kernel.org/

TOOLSET

CXL Emulation on regular 2-socket (2S) server systems

- A paper called **Pond** by Microsoft emulate the following two characteristics of Compute Express Link (CXL) attached DRAM:

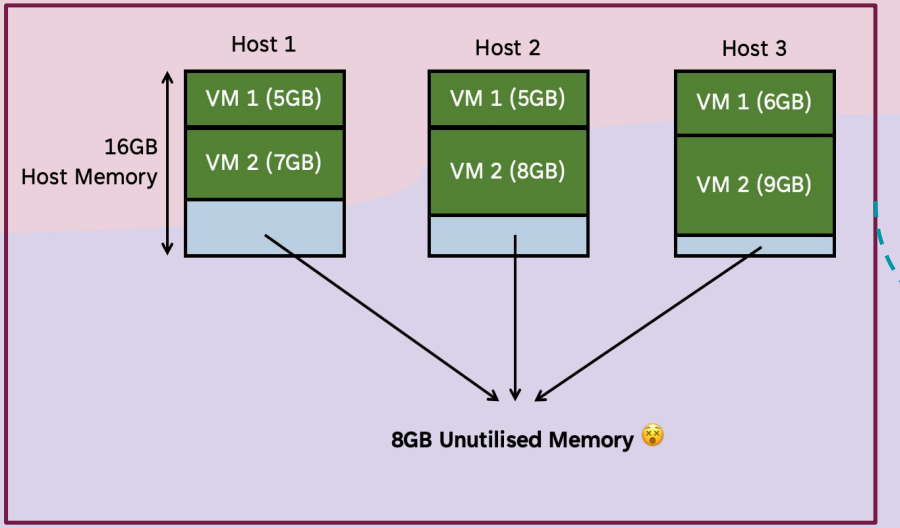




https://github.com/vtess/Pond

synerg@cse.iitb.ac.in

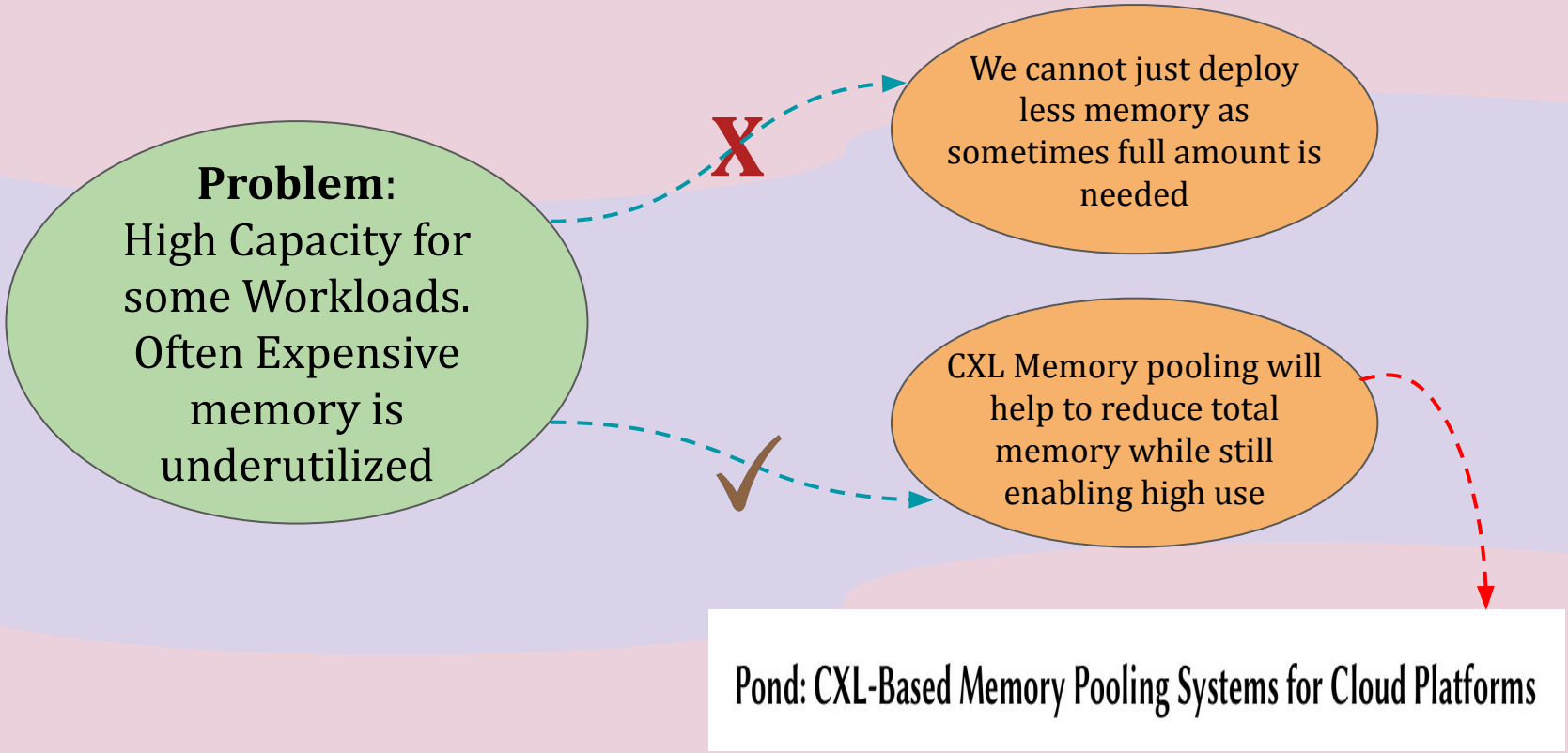
Memory Pooling



➔ Memory is becoming the largest portion of server costs for cloud datacenters
- DRAM can account for upto 50% of server costs

Reasons:
1. Some other resource (e.g CPU) has saturated.
2. Overcommitting limit reached

Memory Pooling



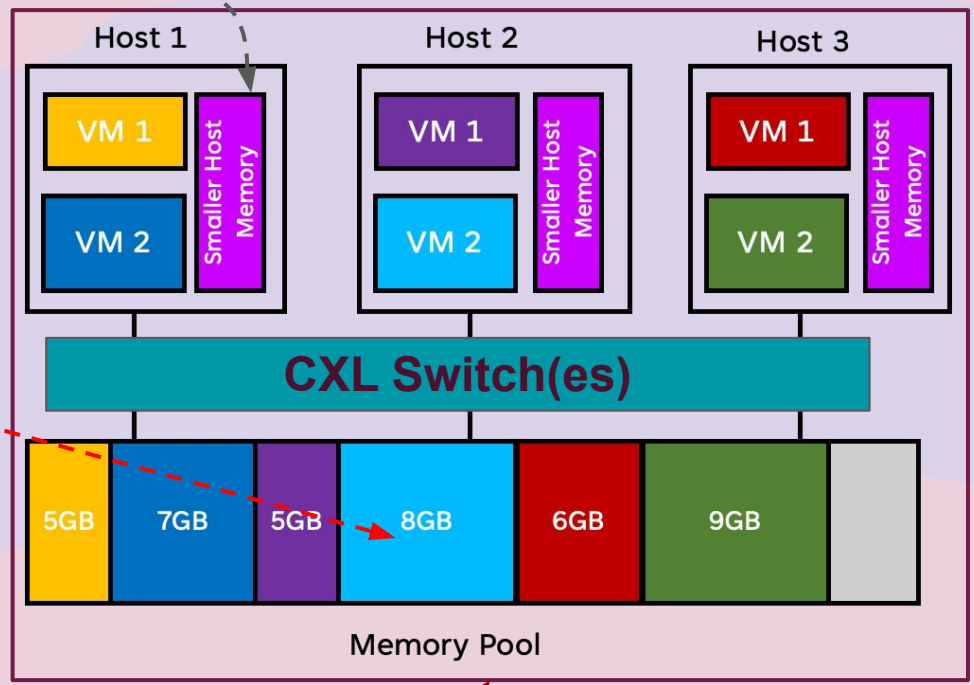
https://github.com/vtess/Pond

Memory Pooling

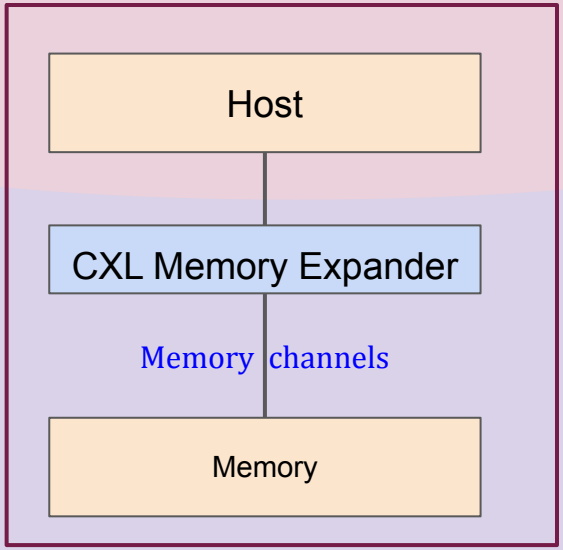
Each Host has a small host memory attached to it

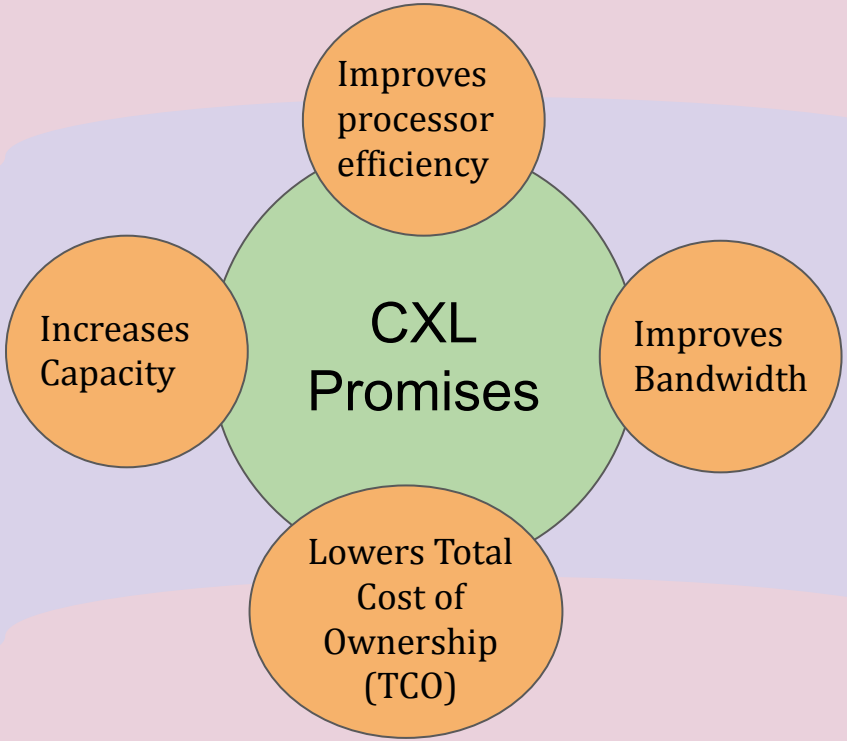
Remaining memory requirements of the VMs

Different policies can define how much, when and how to allocate memory from the pool



CXL Memory Expansion





CXL Memory Expansion

Enabling CXL Memory Expansion for In-Memory Database Management Systems

Minseon Ahn
Donghun Lee
Jungmin Kim
minseon.ahn@sap.com
domg.hun.lee@sap.com
jungmin.kim@sap.com
SAP Labs Korea

Oliver Rebholz
oliver.rebholz@sap.com
SAP SE
Walldorf, Baden-Württemberg
Germany

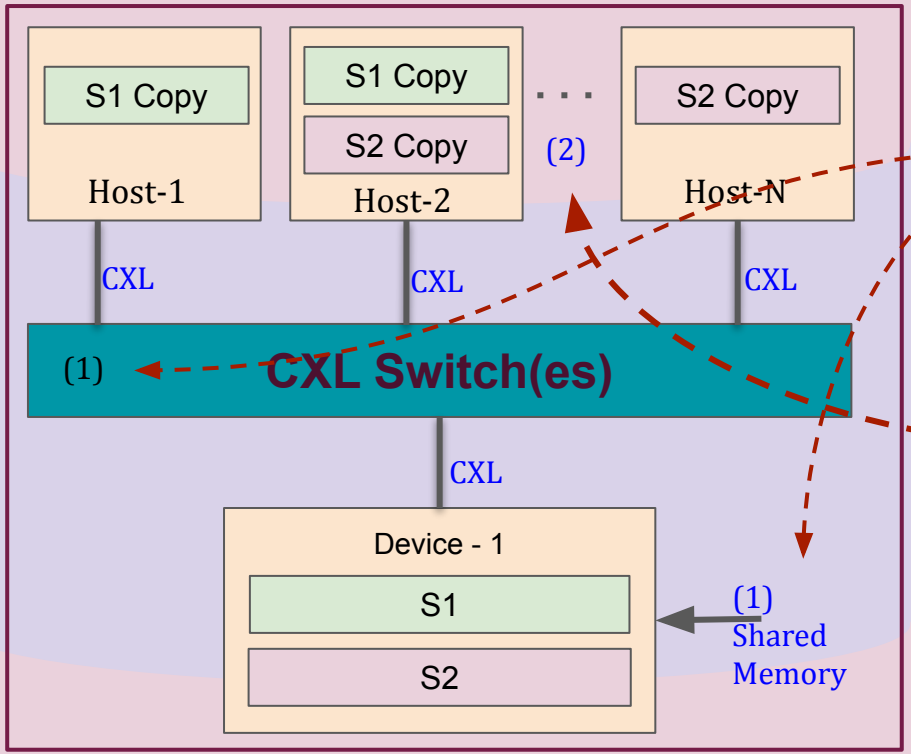
Andrew Chang
Jongmin Gim
Jaemin Jung
Vincent Pham
Krishna T. Malladi
Yang Seok Ki

Limited memory volume is always a performance **bottleneck** in an **in-memory database management system** (IMDBMS) as the data size keeps increasing.

This work proposes a flexible CXL-based **memory expansion** with potentially **lower TCO** in an IMDBMS as one of the significant **use cases** of CXL memory.

https://dl.acm.org/doi/10.1145/3533737.3535090

Coherent Memory Sharing



(1) Device **memory can be shared by all hosts** to increase data flow efficiency and improve memory utilization

(2) Host can have a **coherent copy of the shared region** or portions of shared region in host cache

https://www.computeexpresslink.org/_files/ugd/0c1418_998df4f459734f319e7a12cc2163b943.pdf

Potential works with CXL

- ➔ Development of CXL library that can provide an interface to user applications for performing operations on CXL devices.
- ➔ Creating a standard emulator for different types of CXL devices(Currently there is only Type 3 device emulator available).
- ➔ Modification of device drivers to provide support for CXL-enabled Accelerator devices.

Conclusion

CXL specification is gaining wide traction in the industry due to the simplicity of implementing low-latency caching and memory semantics on a well-established **PCIe** infrastructure.

References:

- Compute Express Link™ and CXL™ Consortium
- https://www.computeexpresslink.org/
- Craig Rodgers' presentation on CXL

Thank you for listening!
Questions?

synerg@cse.iitb.ac.in