How CXL Transforms Server Memory Infrastructure

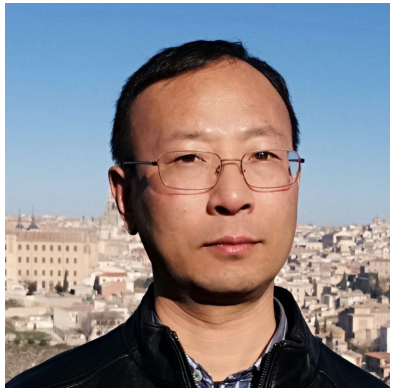
October 8, 2025

Meet our Presenters

Anil Godbole
CXL Consortium MWG Chair

Siamak Tavallaei
Sr Principal Engineer, System Architecture
Samsung

Geof Findley
VP, Business Development/Sales
Montage Technology



JP Jiang
Co-Founder, SVP, Business Development, Operation, Product Marketing
Xconn Technologies

Growth of the CXL

CXL Integrators List

Device Type		Feature Set		Spec Revision		Function	
Type 1	7	CXL Core 1.1	41	CXL 1.1	30	Accelerator	3
Type 2	8	CXL Core 2.0	26	CXL 2.0	30	Host	11
Type 3	57					IP	14
						MEM Expander	35

Show 15 entries

Search:

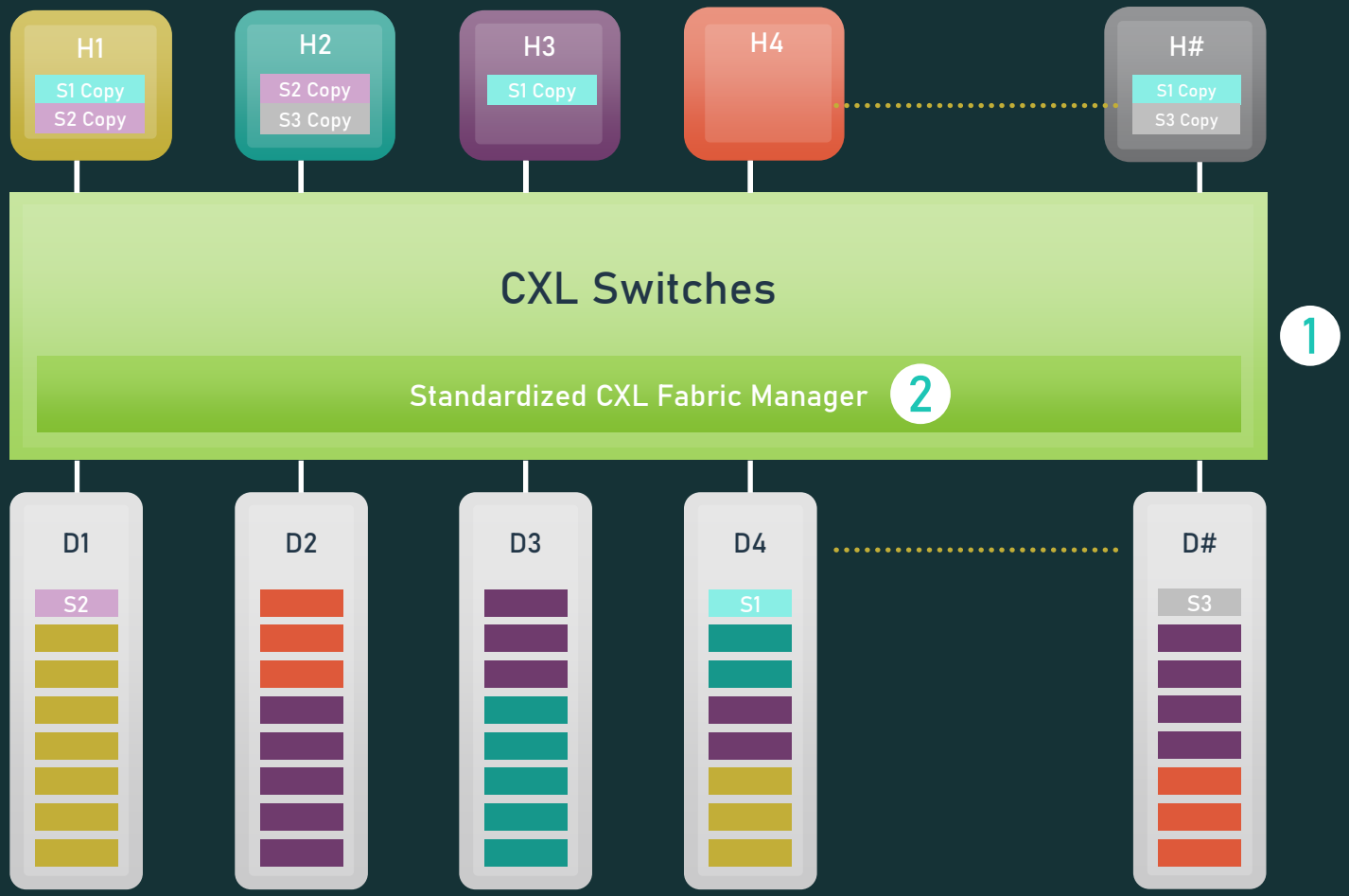
Company Name	Product Name	Device ID	Device Type	Feature Set	Spec Revision	PHY Speed	Max Lane	Form Factor	Function	Compliance Event (CTE) Approved
Micron Technology, Inc.	CZ122	6404	Type 3	CXL Core 1.1, CXL Core 2.0, MEM 2.0	CXL 2.0	32GT/s	x8	EDSFF	MEM Expander	CTE 006
Micron Technology, Inc.	Micron Rev B	6400	Type 3	CXL Core 1.1	CXL 1.1	32GT/s	x8	EDSFF	MEM Expander	CTE 001
Micron Technology, Inc.	Micron Rev A	6400	Type 3	CXL Core 1.1	CXL 1.1	32GT/s	x8	EDSFF	MEM Expander	CTE 001
Montage Technology Co., Ltd	Memory Expander	C001	Type 3	CXL Core 2.0	CXL 2.0	32GT/s	x8	CEM	MEM Expander	CTE 006
Montage Technology Co., Ltd	Memory Expander	C002	Type 3	CXL Core 2.0	CXL 2.0	32GT/s	x8	CEM	MEM Expander	CTE 006
Montage Technology Co., Ltd	CXL Memory Expander Controller (MXC)	0xC001	Type 3	CXL Core 1.1	CXL 1.1	32GT/s	x8	CEM	MEM Expander	CTE 001
Phison Electronics Corporation	PS7201 PCIe 5.0 Retimer	PS7201	Type 3	CXL Core 1.1, CXL Core 2.0	CXL 2.0	32GT/s	x16	CEM	Retimer	CTE 007

OEMs offering CXL-capable servers:

- US / EMEA
  - Dell
  - HPE
  - Lenovo
  - Supermicro
- APAC
  - Advantech
  - Giga
  - Quanta
  - AIC

Scan the QR code to view the Integrators List

CXL 3.0: Pooling & Sharing



Benefits:

- Low-latency data sharing
  - No shuffling of data
- Total Cost of Ownership

Applications:

- Distributed databases
- RAG
- Graph database
- KVCache storage
- Algo Trading

Industry Efforts Around CXL-enabled Solutions

Samsung Semiconductor Inc.

Presented by: Siamak Tavallaei

Bottlenecks in AI/ML Systems

1. Memory Footprint *(Capacity)*
   - Billions of high-dimensional vectors require terabytes of storage
   - Full in-memory storage improves speed but hits capacity limits

2. Memory Bandwidth *(Avoid bottlenecks)*
   - CPUs become memory-bound, not compute-bound (data access time dominates)
   - Responsiveness in interactive applications

3. Data Transfer Overheads *(Considerations for time and energy per transferred bit)*
   - Vectors must cross PCIe or NVLink to reach GPUs
   - Multi-GPU setups face traffic congestion and NUMA issues

4. Resource Fragmentation *(increased consistency overhead)*
   - Redundant dataset copies across distributed nodes
   - Shared, dynamic memory will help

5. Generation Phase Limits *(Long-context inference)*
   - Result: truncated contexts, smaller batches, underused model potential

System Requirements to Mitigate Bottlenecks

1. Shared Memory Semantics & Direct Load/Store
   Eliminate redundant vector copies → reduce memory footprint, fragmentation, overflow risks
   Efficient communication between accelerators removes software overhead
2. High-Throughput, Low-Latency Fabrics with Efficient Protocol
   Real-time token generation & interactive retrieval-to-generation cycles
   Enable large-scale vector access, parallel search, and fast compute-memory transfers
   Cuts down overhead → improves dense similarity search and memory-bound ops
3. Predictable Response Time *(Low Jitter)*
   Ensures consistent inference performance and task synchronization
4. Fine-Granularity Flow-Control
   Manages traffic spikes, avoids congestion, buffer overflows, and Head-of-Line blocking
5. Memory Ordering Guarantees *(Ensures correctness in distributed long-context generation & training)*
6. Reliable, Loss-less Transfer *(Sustains throughput under load; avoids costly retransmission cycles)*
7. Confidential Compute & ***Security*** *(*Protects data at-rest and in-motion for enterprise-grade privacy)

RAG Pipeline: *a growing opportunity example*

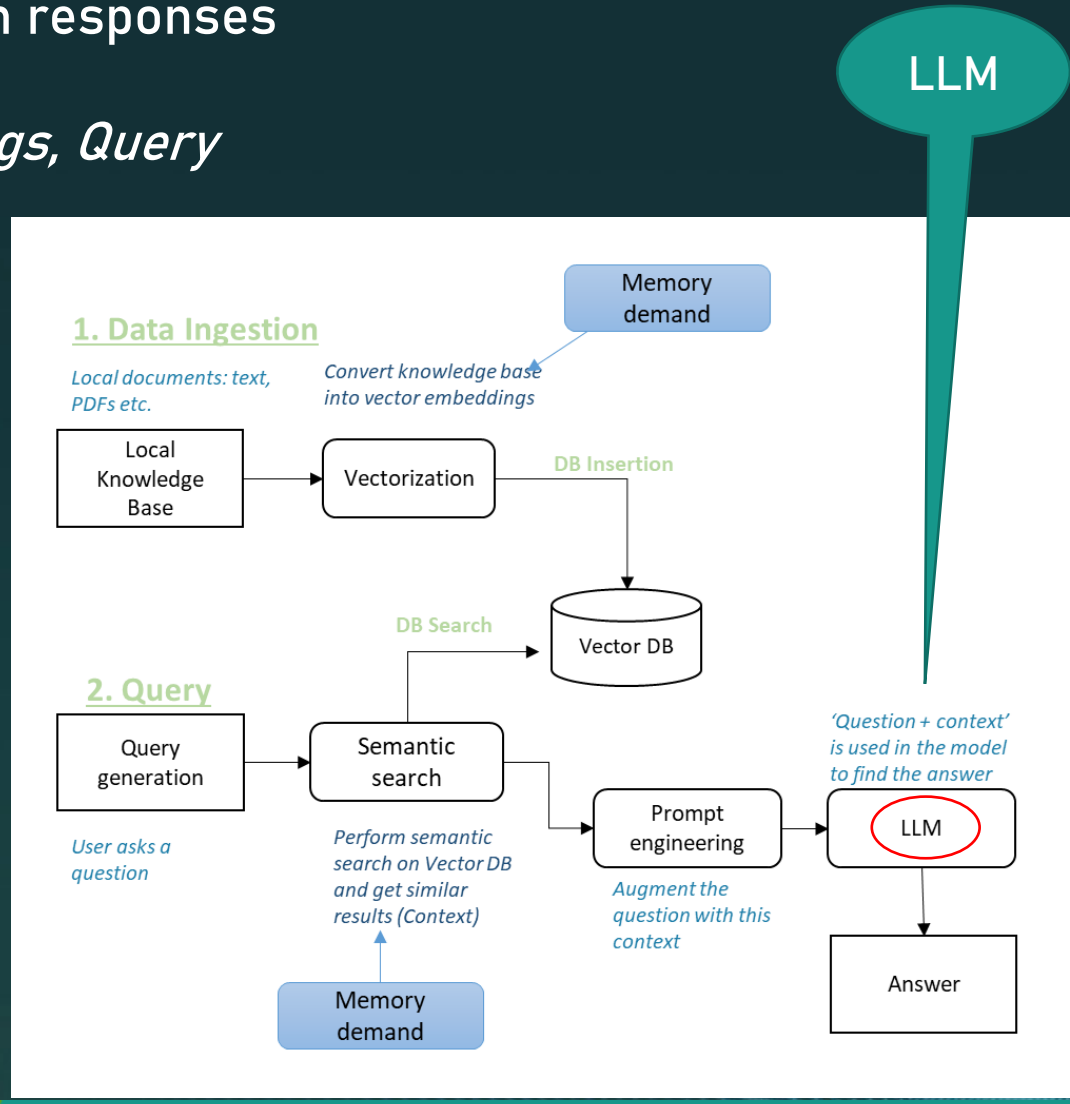
RAG is a machine-learning architecture that integrates information-retrieval with generative models to provide accurate, grounded, and context-rich responses

A pipeline of Core components: *Data, Model, Embeddings, Query*

1. Traditional Search and Database Techniques
   - Indexing Subsystem *(Embedding)*
   - Retrieval Engine *(Finding Relevant Information)*

2. Emerging AI/ML techniques
   - Generation Component *(LLM)*

Observations
- Transient memory demands
- during various pipeline stages



Reference: FMS 2025 AI/ML Track: AIML-304-1
By: Ardavan Sherafat, Aug 7th, 2025

Observations throughout the pipeline

Data-retrieval & Search Techniques (Databases)

- MemCacheD
- Vector DB
- Key-Value Store and KV-Cache

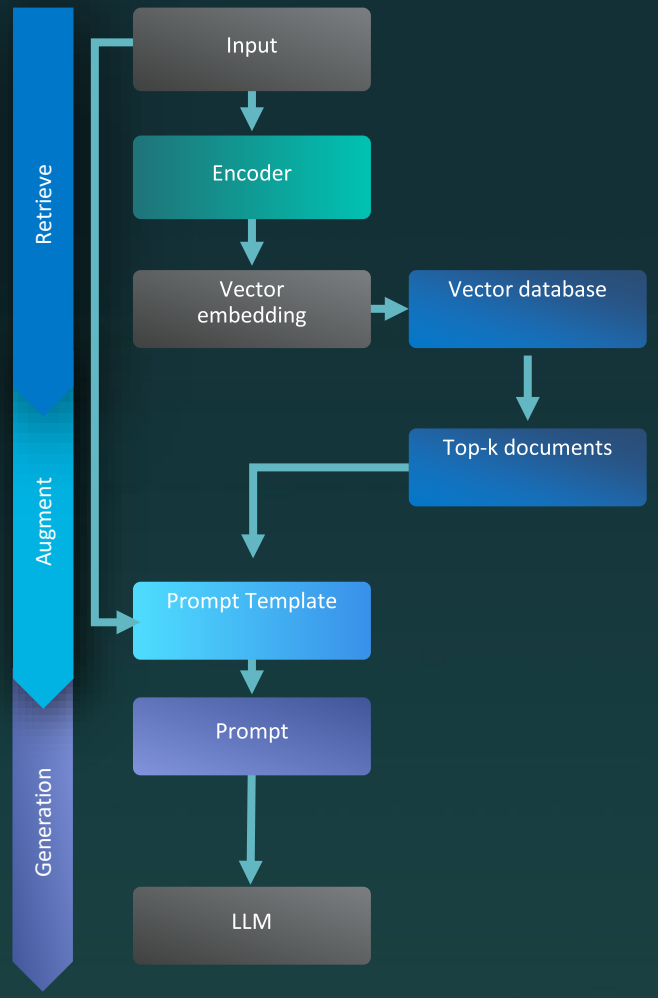
Pre-processing

- Encoding, Embedding
- Document processing
  - organizing and structuring unstructured data from different input types
  - Reducing data-transfer through summarization and compression
  - All-Reduce, All-Gather, …
- Check-pointing & Restart
- Agentic Role-back

Hardware elements that can help

- Mem expansion (large memory footprint to keep processed data and to process more!)
- Memory pooling (expand and deflate memory footprint for different phases of the flow)
- Memory sharing (for reducing communication time and energy overhead)
- Near-memory & In-memory Processing

RAG Flow Chart

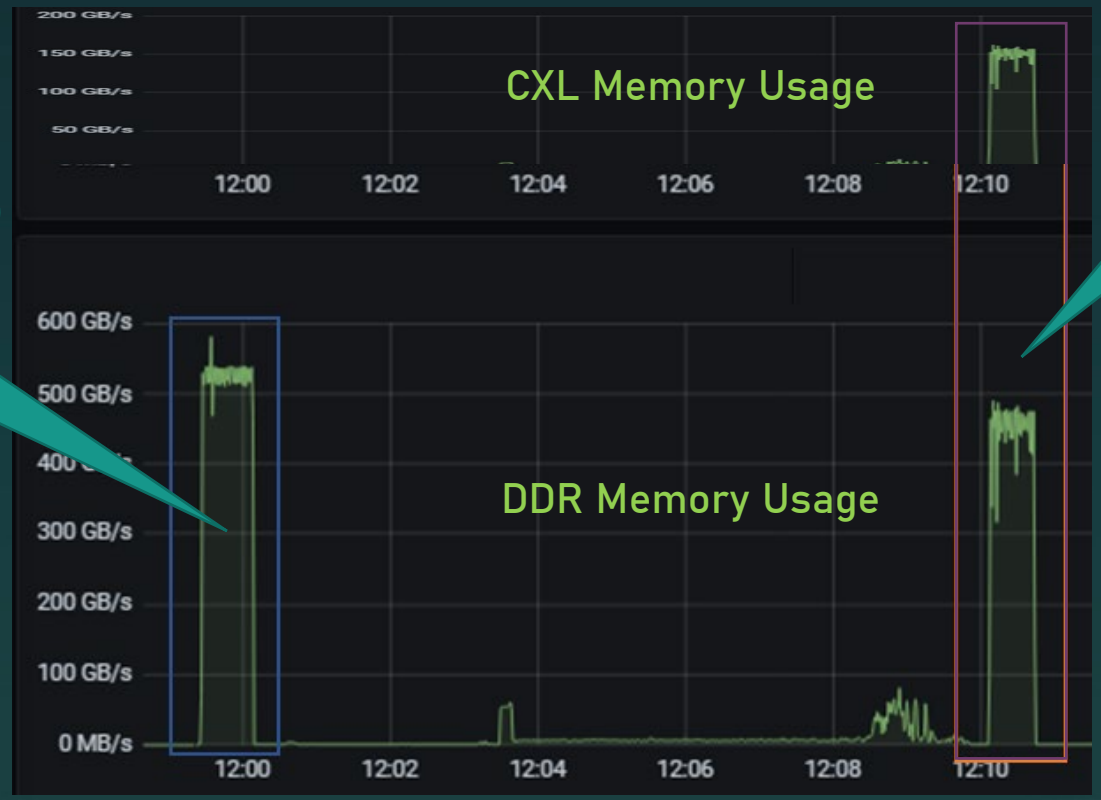


Source: Samsung FMS presentation, Aug 2025

Memory Bandwidth Utilization

Memory System DDR5 Bandwidth
(DDR5-only Memory)

Read B/W : 539GB/s



Memory System Bandwidth
(DDR5 + CXL Memory)

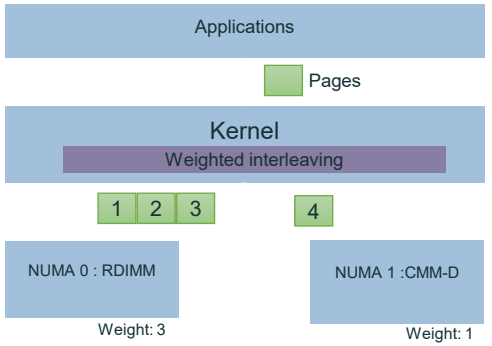
Aggregated Read B/W of
Weighted Interleaving (4:1)
636GB/s

Advantages with CMM-D in RAG Cluster

Up to 19% higher performance with CXL-connected DRAM (CMM-D) in VectorDB search compared to Local-DRAM-only case in Milvus RAG cluster

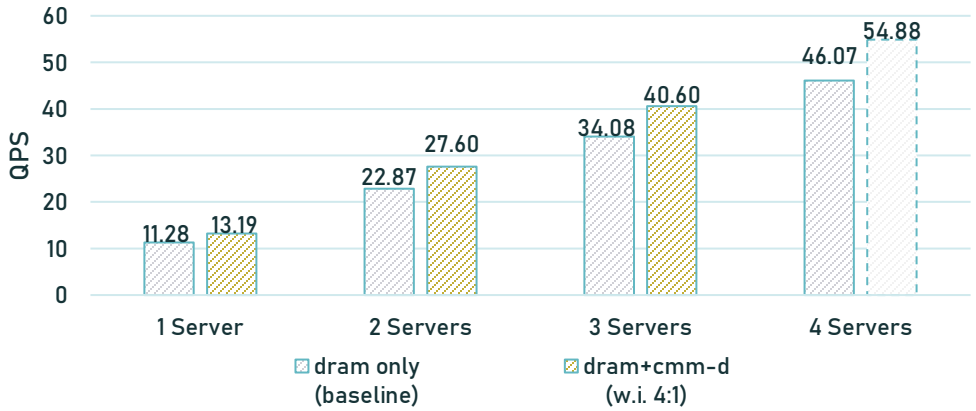
- Performance gain with bandwidth expansion through the CMM-D in Milvus RAG Cluster
- Using SW interleaving (between Local DRAM and CMM-D DRAM) to achieve optimal CXL bandwidth performance

- Linux kernel SW weighted interleaving provides opportunity to define an interleave ratio to best utilize DRAM and CXL memory for optimal performance in a workload
- Included in Kernel Mainline (v6.9)

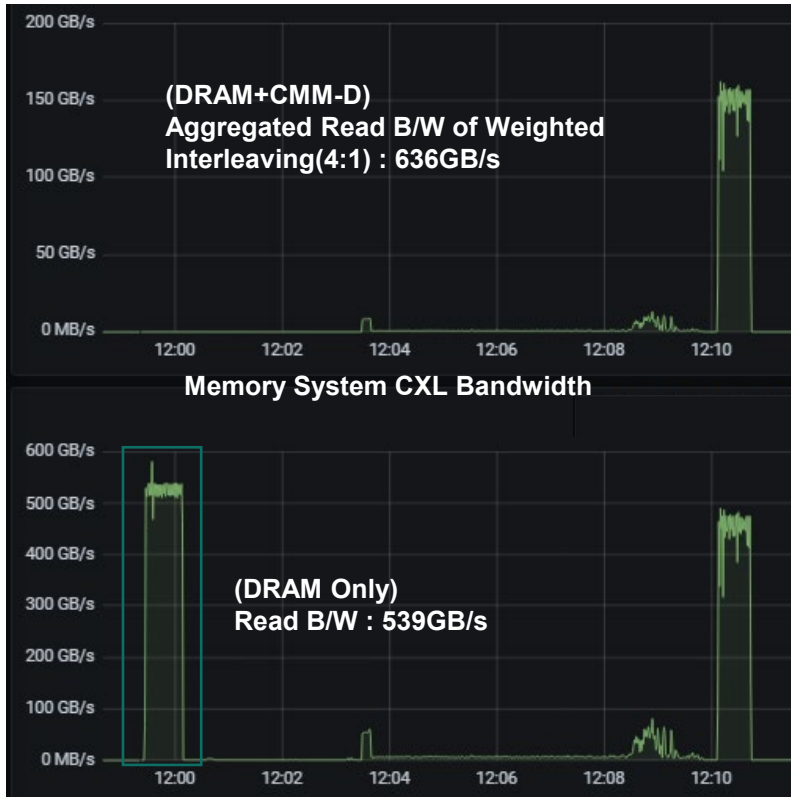


Vector Search Performance Result with CMM-D

Comarison of QPS by Number of Servers



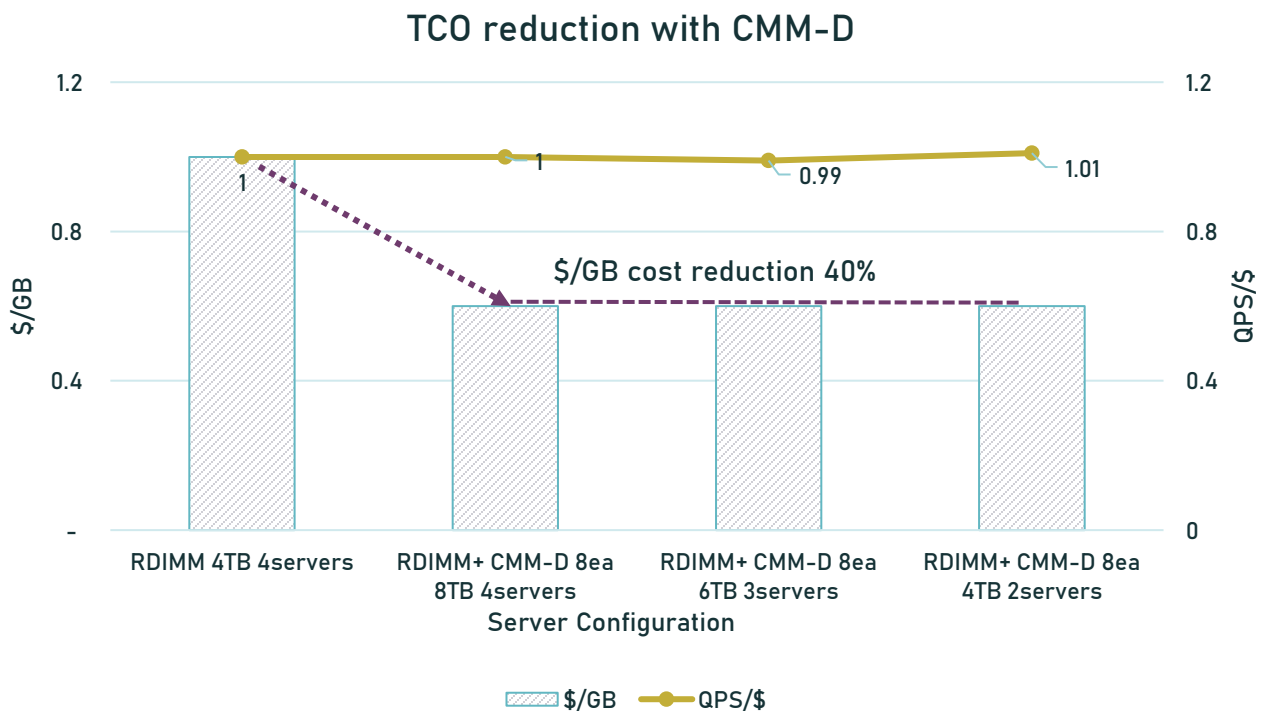
B/W monitoring results in one data server



Advantages with CMM-D in RAG Cluster

TCO reduction effect and memory expansion effect can be secured

- Equivalent QPS/$ and 40% reduction in $/GB cost
- Operating Power reduction through application can reduce operating cost



Dataset : MSMARCO-V2

Raw Size	Indexing Size(HNSW)	Entity Count	Dimension	Precision	Vector Size
290GB	673GB	138 Million	1024 (cohere)	FP32	4096B

Reference TCO Calculator :https://v0-cxl-tco-2-nvdatd.vercel.app/

Observations

- We need more memory *(compute capability grows with increased memory)*
- Workloads use different amounts of the available memory footprint during various phases in the pipeline
- If we don't provision enough memory, the size of problems we can solve is limited
- If we maximally provision memory, we end up with under-utilized resources
- Resource-pooling based on disaggregated computing helps inflate and deflate available memory for each processing element at the appropriate phase during the pipeline
- The results show:
  - Memory capacity and bandwidth utilization throughout the pipeline stages
  - Performance gain when enough memory is available during the critical phase
  - With reasonable size of deployed resources (TCO)
- Driving solutions based on simulation and lab analysis through the open-source efforts

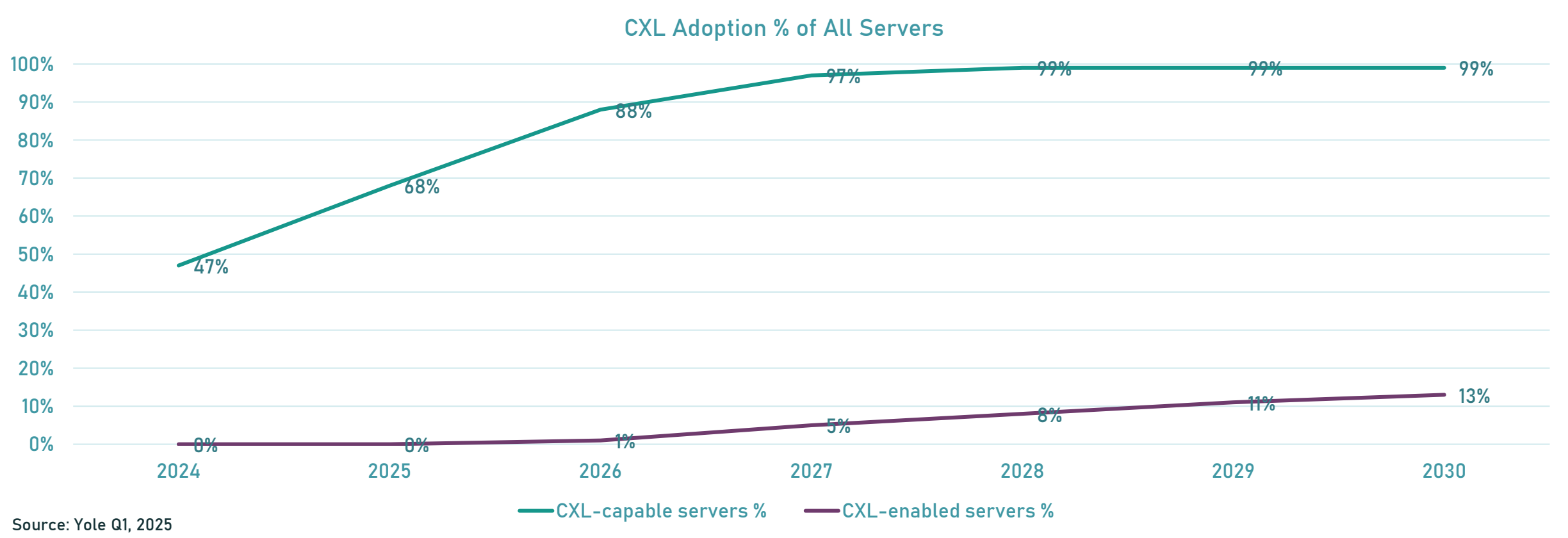
Montage Technology

Presented by: Geof Findley

Montage Technology and CXL Memory Expansion Controller – MXC

- Montage more than 20 years in memory products - leading the industry. Second largest PCIe GEN5 and GEN4 Retimer supplier. Largest and 1st to ship CXL controllers…Gen 1, 2, and now Gen 3
- MXC newest product based on deep understanding of both DDR and PCIe technologies
- MXC GEN1 supports CXL2.0 and DDR4-3200/DDR5-4800 (In mass production)
- MXC GEN2 supports CXL2.x and DDR5-6400 (In mass production)
- MXC GEN3: Shipping M88MX6852 Type3 CXL® Memory eXpander Controller (Industry 1st)
  - CXL 3.1 compatible
  - PCIe Gen6 speed up to 64GT/s
  - CXL x8 port with bifurcation to 2x4 ports
  - Up to DDR5-8000, with two independent memory controllers
  - Enhanced RAS capability
  - Security with IDE/TSP/DICE
  - Rich management features

CXL Support Available NOW from Intel and AMD

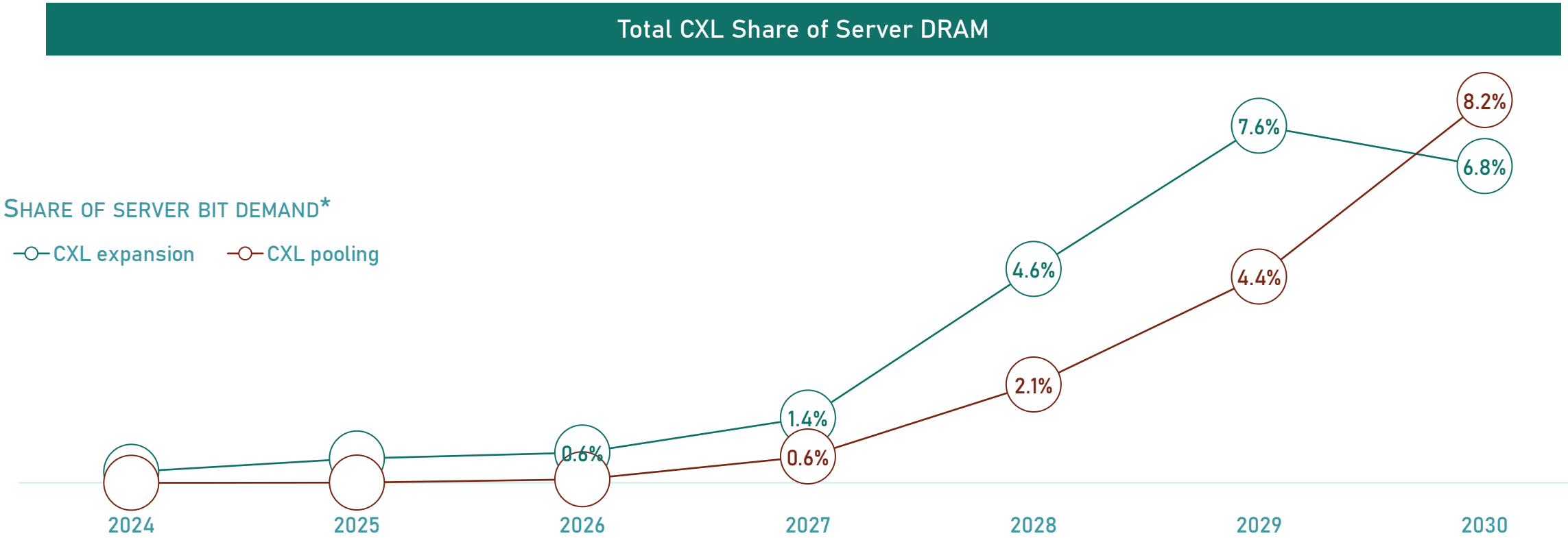


Source: Yole Q1, 2025

Two Thirds of Servers today can support CXL products by end of '26 well over 90%

CXL adoption in the datacenter…its Started

- 32Gb monolithic die (and corresponding 128GB RDIMM) and MRDIMM (with higher bandwidth) are alternatives to CXL expansion. Expansion being considered when high DRAM content is desired
- Memory pooling is getting developed and deployed with CXL 2.0 today and will explode when CXL3.x is available next year



Source: Techinsight Q2, 2025

TCO Savings Examples with CXL Memory

Avoid High-Cost DIMMs

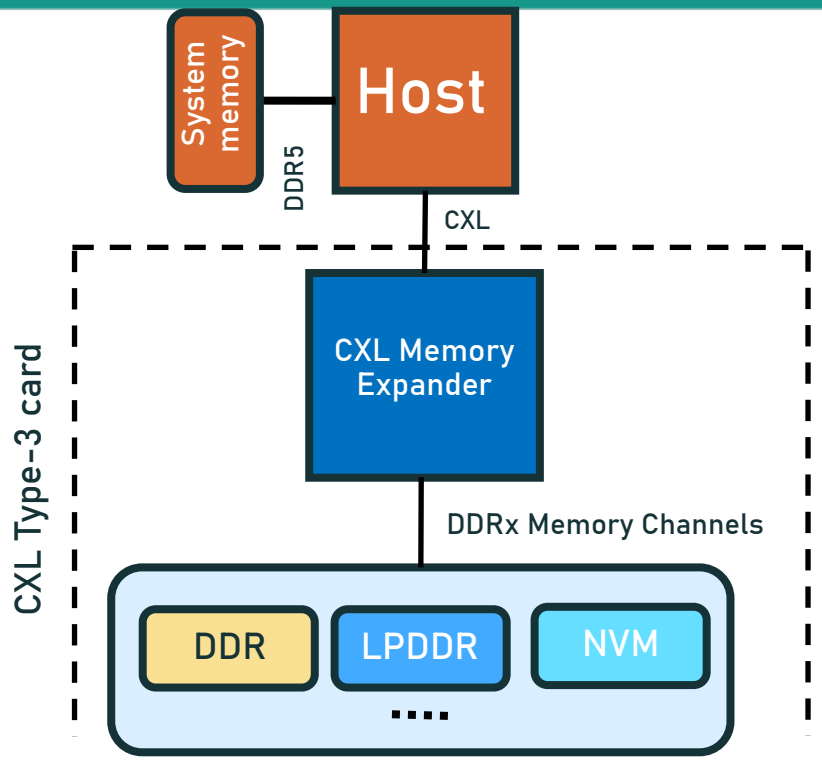
- 256GB DIMMs have high price premiums due to the need to 'stack' the DRAM
- CXL add-in cards with DIMM slots provide more total channels per socket → same system capacity with lower priced DIMMs
- Same concept applies regardless of mode: Intel Flat Memory Mode & SW-based tiering

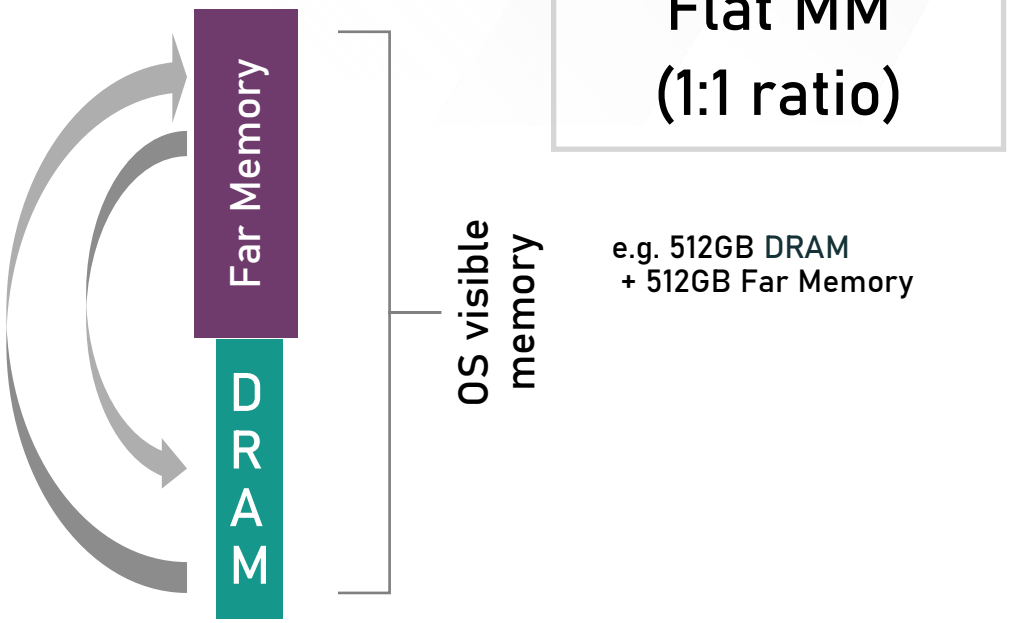
Memory per socket	Socket-attached DIMMs only	With CXL	Memory TCO Savings*
4TB	16x 256GB	16x 128GB + 16x 128GB on CXL	16%

*TCO savings based on Intel modeling using projected DIMM and CXL pricing for 2025

Use CXL-attached DIMMs to achieve high system memory capacity and avoid expensive high-capacity DIMMs

Memory Tiering: H/W Based Example, Intel FMM





- Both DRAM and far memory exposed to OS as combined physical memory – One memory tier
- Data is 'Tiered': Resides in either DRAM or FM - no replication
- Hot data is swapped into DRAM – one cacheline at a time, not a whole 4KB page
- Performance very good due to 1:1 Near/Far memory ratios
- No software modification needed

Summary and Conclusion

- Throughput Analysis
  - CXL FMM Setup shows performance almost equivalent to Native Setup in terms of throughput
  - Across all tested workloads, the CXL FMM setup consistently delivers performance within ~95–100%

- Latency Analysis
  - Read Latency: CXL FMM Setup tends to be 5–10ns higher than Native, a relative increase of 3–5%
  - Update Latency: Generally slightly higher on CXL FMM (10–20ns)
  - Latency results are consistent across repetitions
  - Latency with CXL FMM Setup is slightly higher, especially for update operations, but the increase is small

- Stability
  - Each workload repeated 3 times, and results were highly consistent, indicating stable system behavior.
  - No performance degradation or instability was observed due to CXL usage.

Conclusion: CXL FMM Setup demonstrates excellent usability and stability in MongoDB performance testing

Note: CXL Memory Module only add additional hardware latency less than 100ns. However, Overall CXL FMM latency addition is 5-10ns. This hints much of the latency savings could come from software side improvement

Next Step – Get involved now…

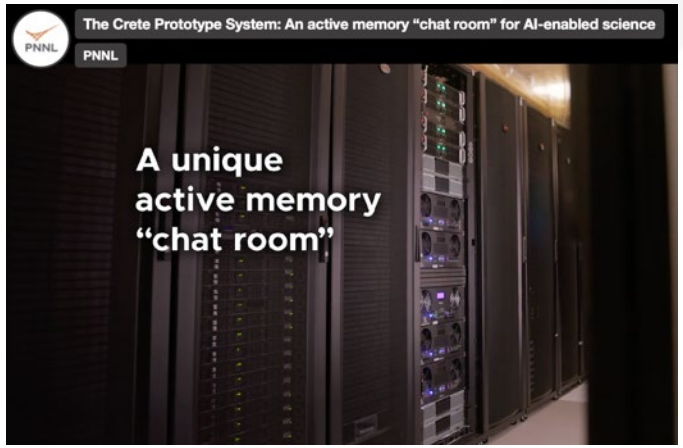
- Both Intel and AMD support CXL 2.x today on their new server platforms
- Multiple CXL E3 solutions are available from all the major DRAM suppliers
- Multiple CXL AIC solutions are available from suppliers in all GEOs
- If you are interested in CXL controllers, contact Montage (globalsales@montage-tech.com)
- Suppliers will start rolling out E3 and AIC CXL 3.1 solutions later this year and be in mass production mid next year
- Work with your favorite DRAM or DIMM supplier to obtain CXL products or contact Montage directly for EVBs(globalsales@montage-tech.com)

- We need to start working on SW optimization NOW so when Intel and AMD roll out next year their 3.1 support we are ready not only with HW but with optimized SW

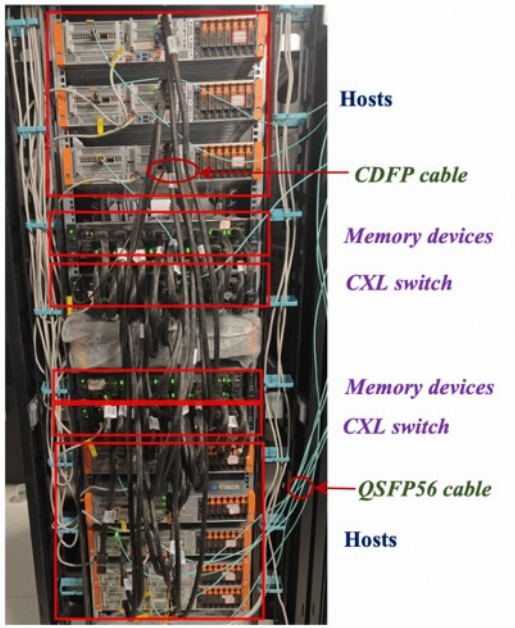
XConn Technologies

Presented by: JP Jiang

CXL Memory Pooling Chassis, Racks



PNNL Crete: CXL memory pool for scientific and AI computing



Alibaba Cloud:
CXL memory pool for PolarDB, a cloud native database

Boost Performance of Scientific and AI Computing

GraphDB performance increases 20X with CXL JBOM

What took 7hrs (an international flight) drops to 12mins (a coffee break)

Server (no CXL attached)

→ Run time 25,920s[1] (7hr12m)

CXL JBOM
(5.5TB via 22 CXL modules)

Up to 4 ports of PCIe x16 lanes

Server connected to CXL JBOM

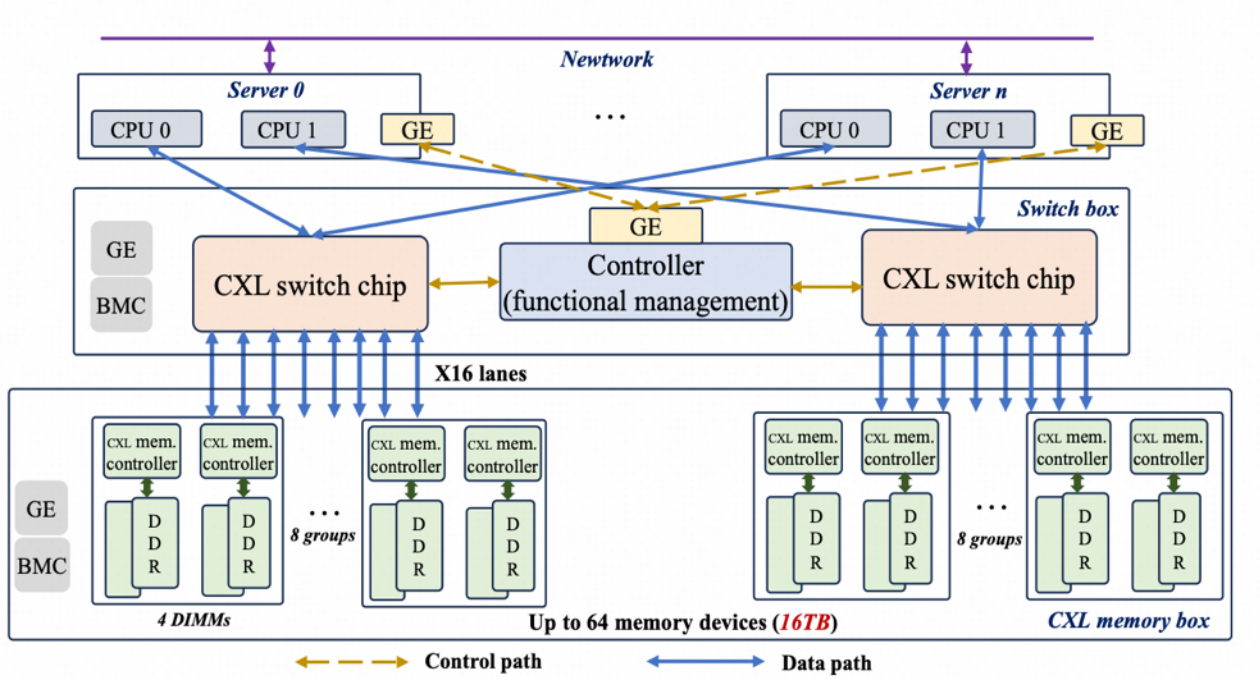
→ Run time 761s[1] (12.6m)

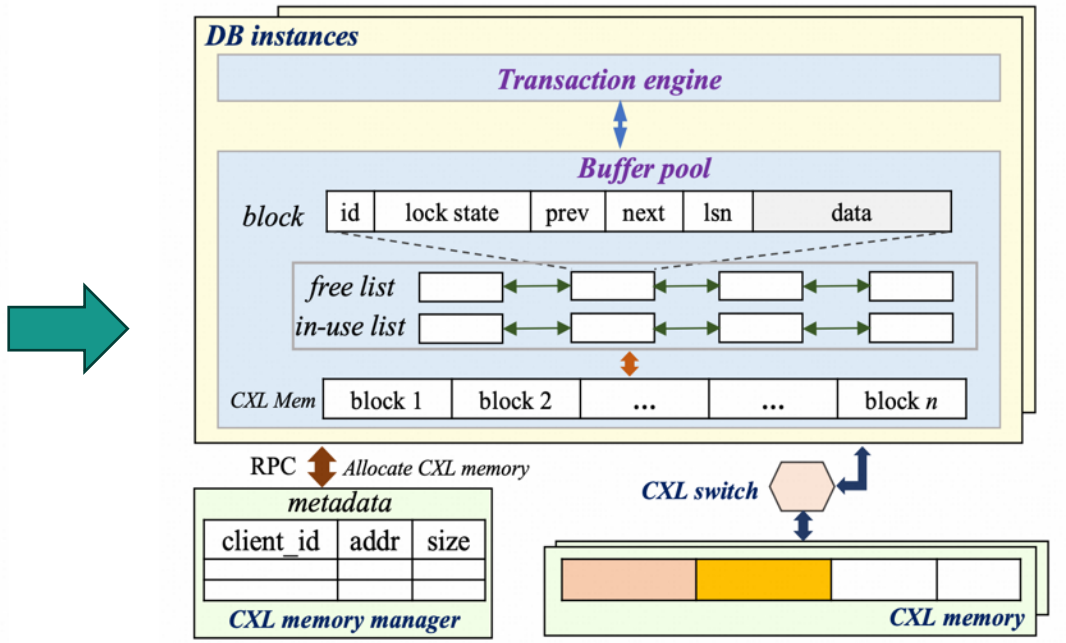
20X
performance leap
Measured speed-up in GraphDB workloads with CXL

5TB → 100TB+
scale up of DB sizes
Order of magnitude more DRAM per server enabled by CXL

7h → 12 min
runtime reduction
Graph500 analytics benchmark completion time

CXL-based Memory Pool in PolarDB



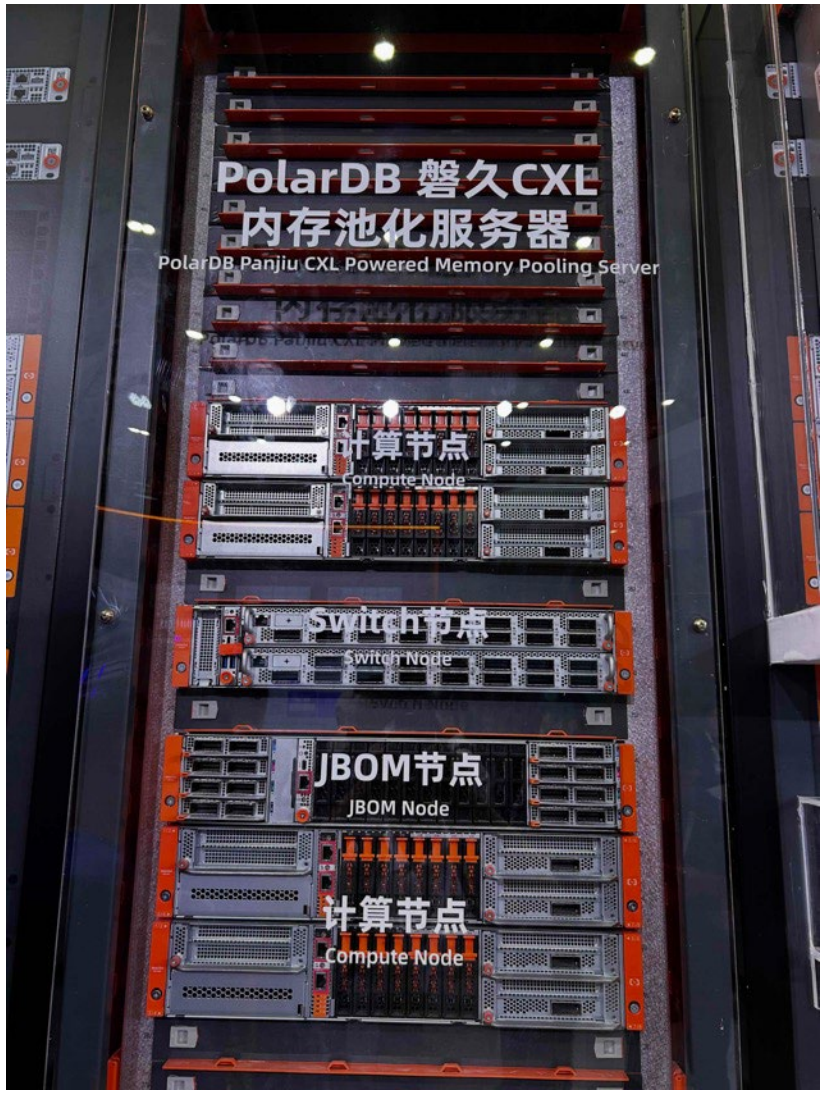


- Servers send control message via Ethernet
- CXL switch is connected via CXL x16 lanes
- Up to 16 TB memory

- Avoid tiered memory, deploying BP directly on CXL memory
- A metadata server is dedicated for the CXL memory pool management
- Compute node allocates CXL memory via RPC

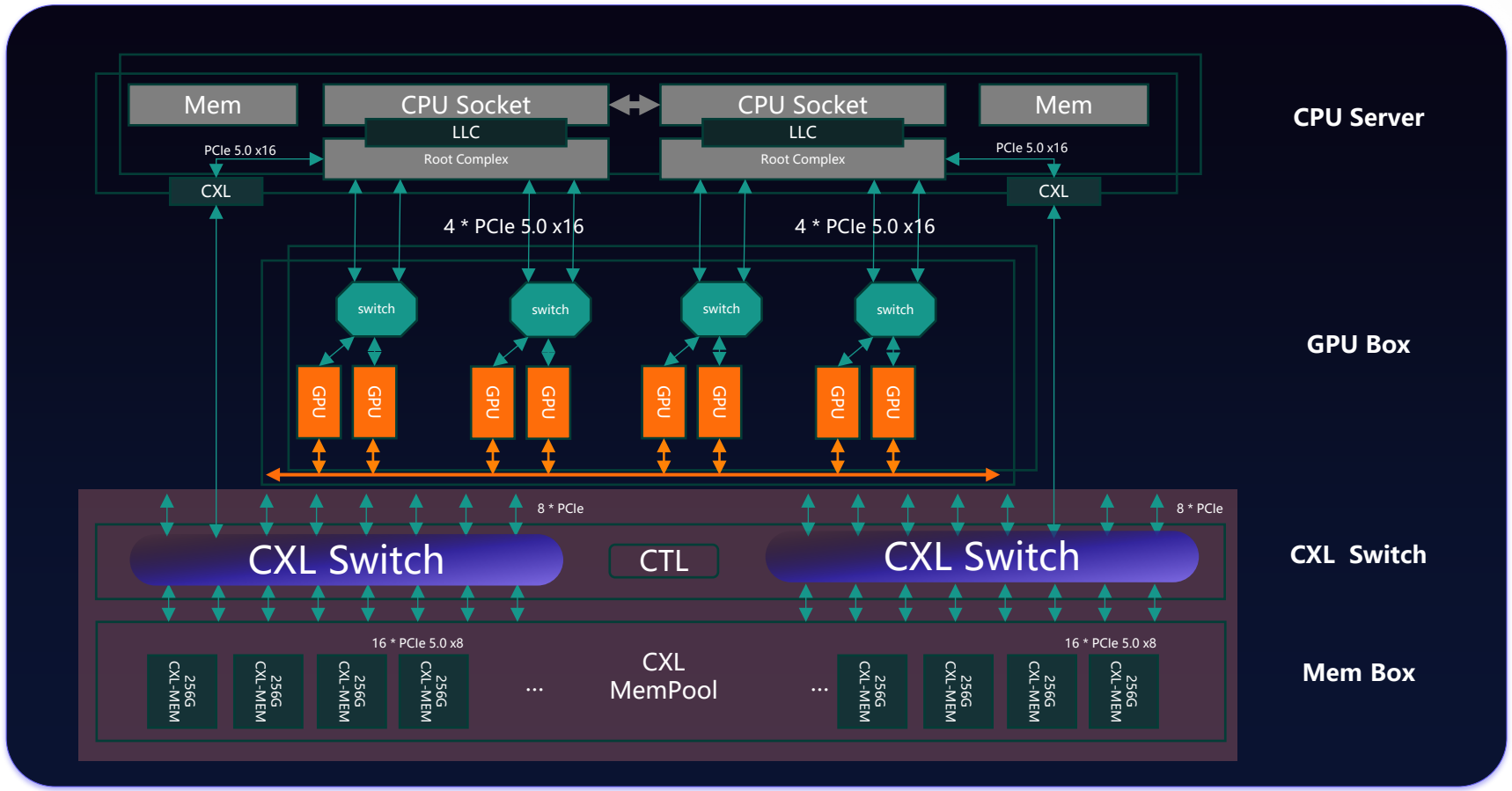
ACM 2025 SIGMOD Industrial best paper award
Unlocking the Potential of CXL for Disaggregated Memory in Cloud-Native Databases
https://lnkd.in/gwwB_4Ph

Alibaba YunQi Event 2025



PolarDB CXL: CPU + GPU

Bridging Heterogenous AI Compute



4.8x

Improved inference throughput;

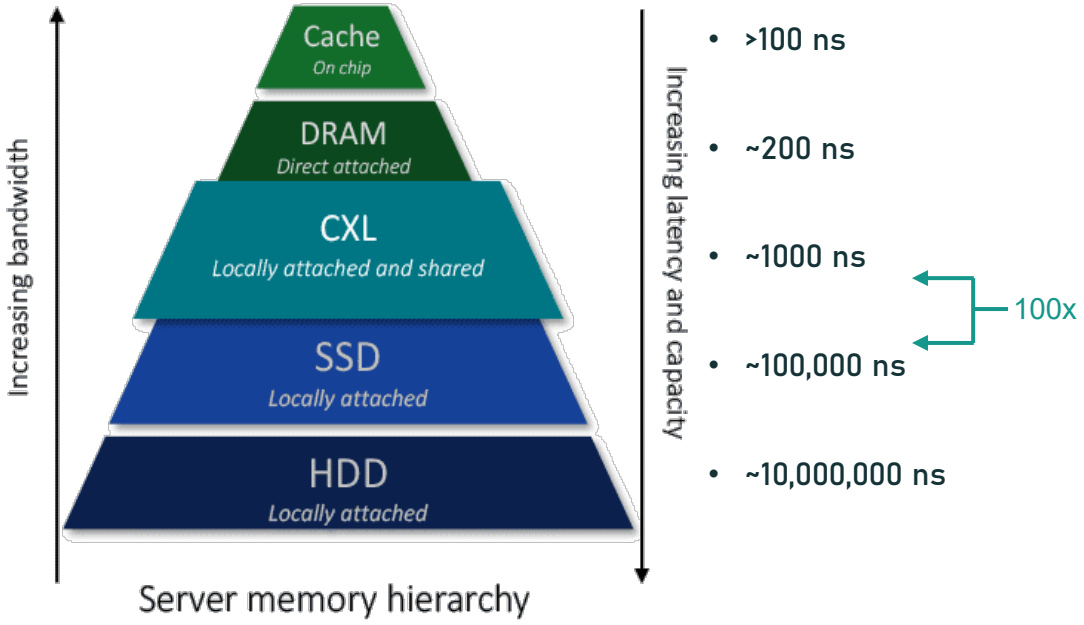
Fully improved GPU power utilization

82.7%

Time to First Token (TTFT) decreases

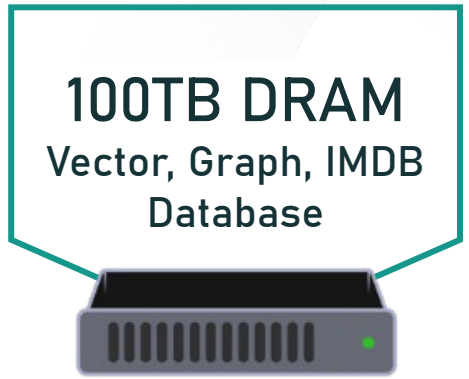
Higher Performance Computing Acceleration

Latency Tiers



CXL Provides 100x Lower Latency Over NVMe

Data Base Acceleration



- Large database shifts from NVMe to CXL
- GraphDB performance boost 20x with CXL JBOM, 7 hours work completed in 12 minutes

CXL Memory Pool:
The Critical Scaleup for AI Workload Memory

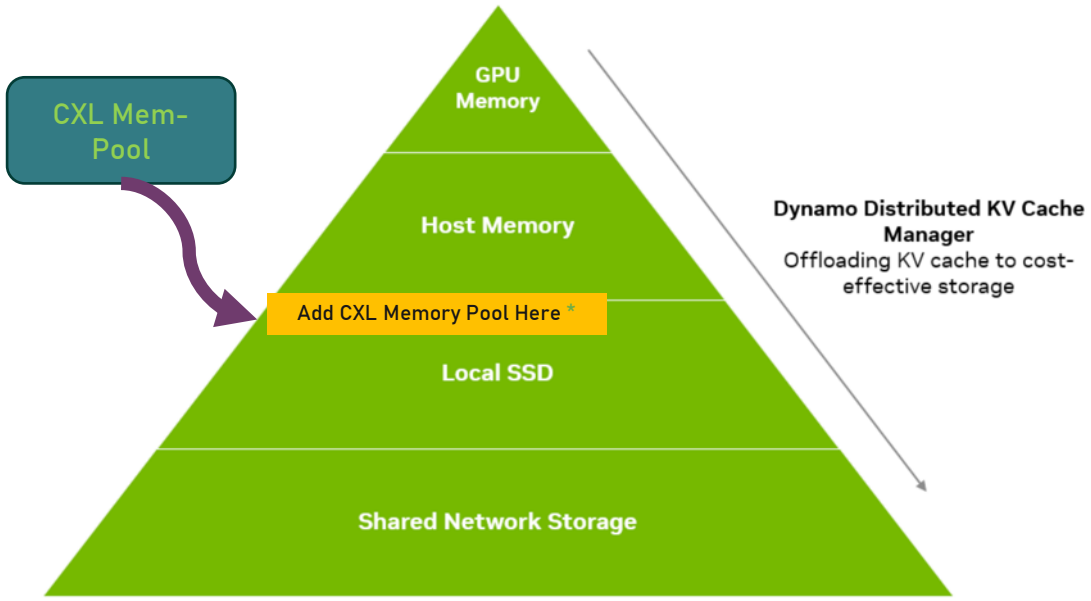


Figure 6. NVIDIA Dynamo Distributed KV Cache Manager offloads less frequently accessed KV cache to more economical memory hierarchies

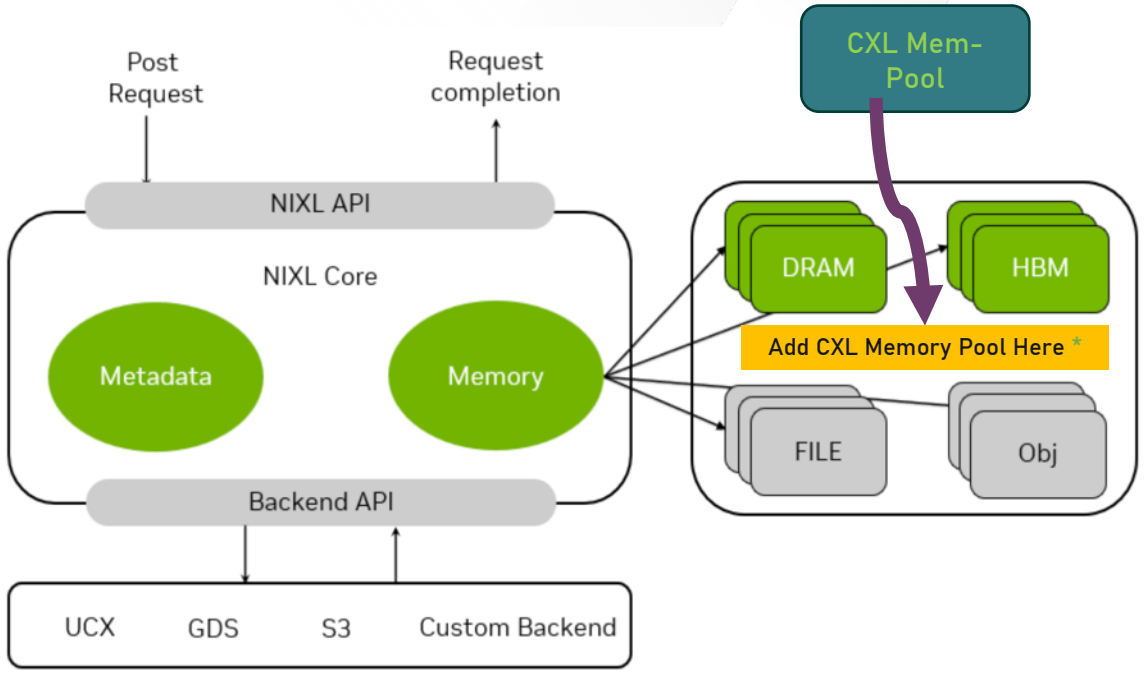


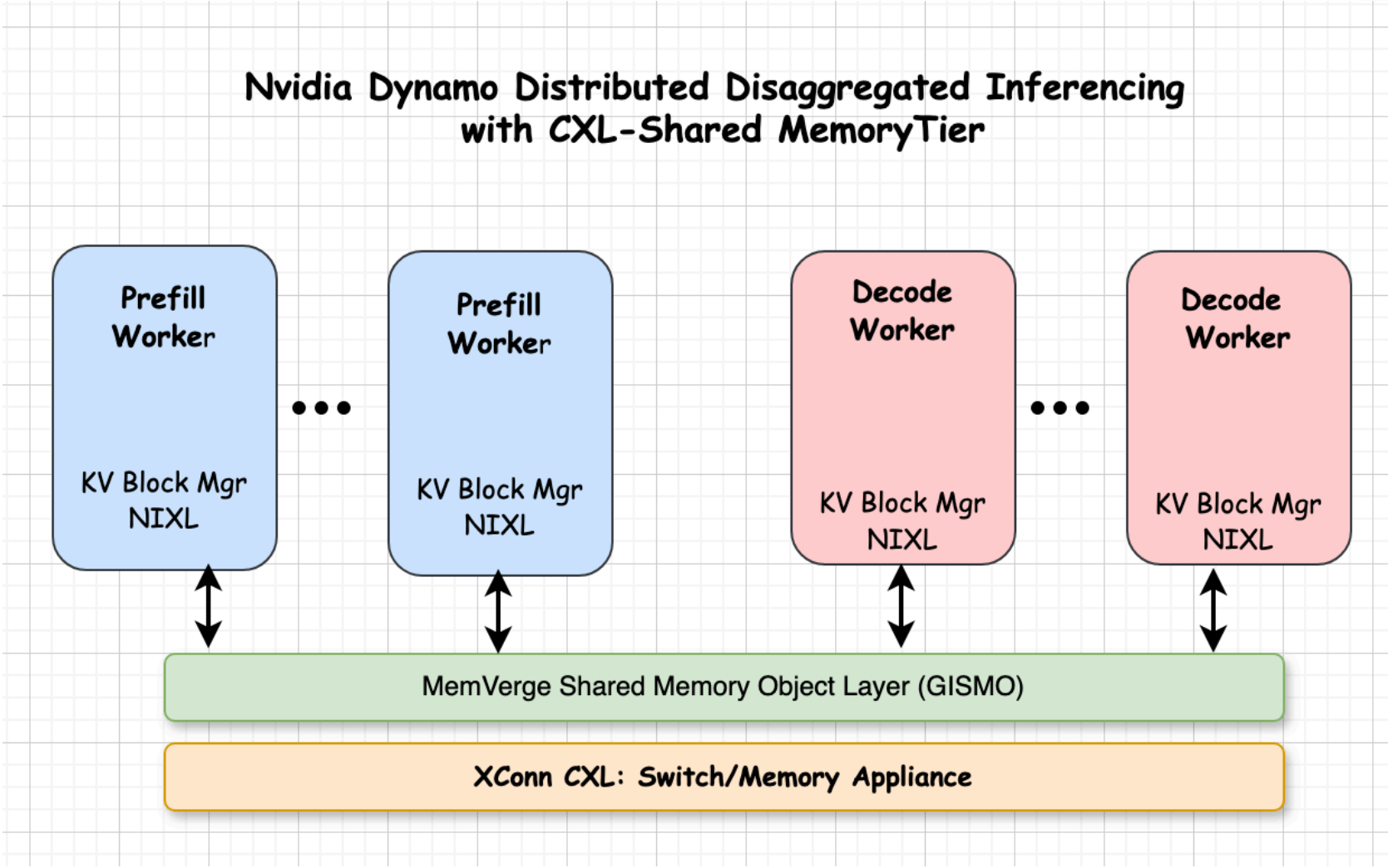
Figure 7. NVIDIA Inference Transfer Library (NIXL) abstracts the complexity of data movement across heterogeneous memory and storage devices

* CXL Memory pool is the critically missing component in the systems for AI workloads:

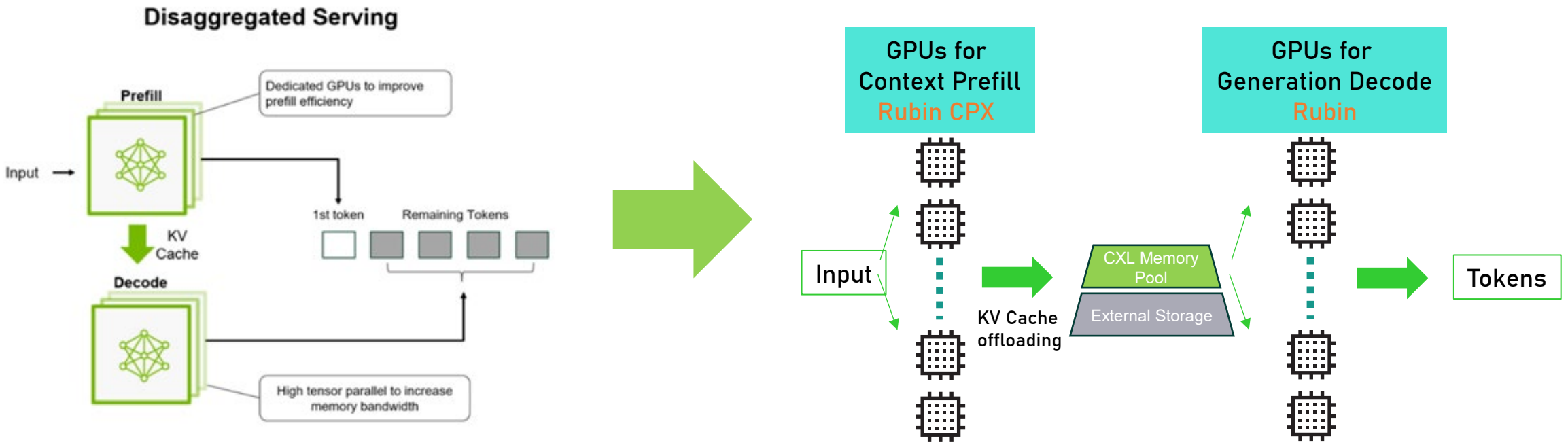
- Dynamic, flexible, cost-effective scaling of memory
- Low-latency, high-bandwidth shared memory access
- Energy-efficient infrastructure with optimized resource use
- Boosted performance of AI workloads, reduced TCOs

https://developer.nvidia.com/blog/introducing-nvidia-dynamo-a-low-latency-distributed-inference-framework-for-scaling-reasoning-ai-models/

Augmenting Nvidia Dynamo KV Cache Hierarchy



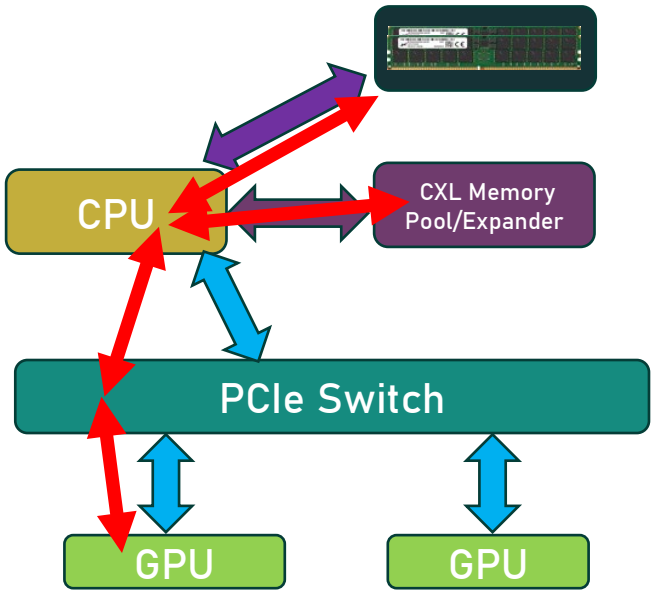
Memory Scale up for AI Inference Workload



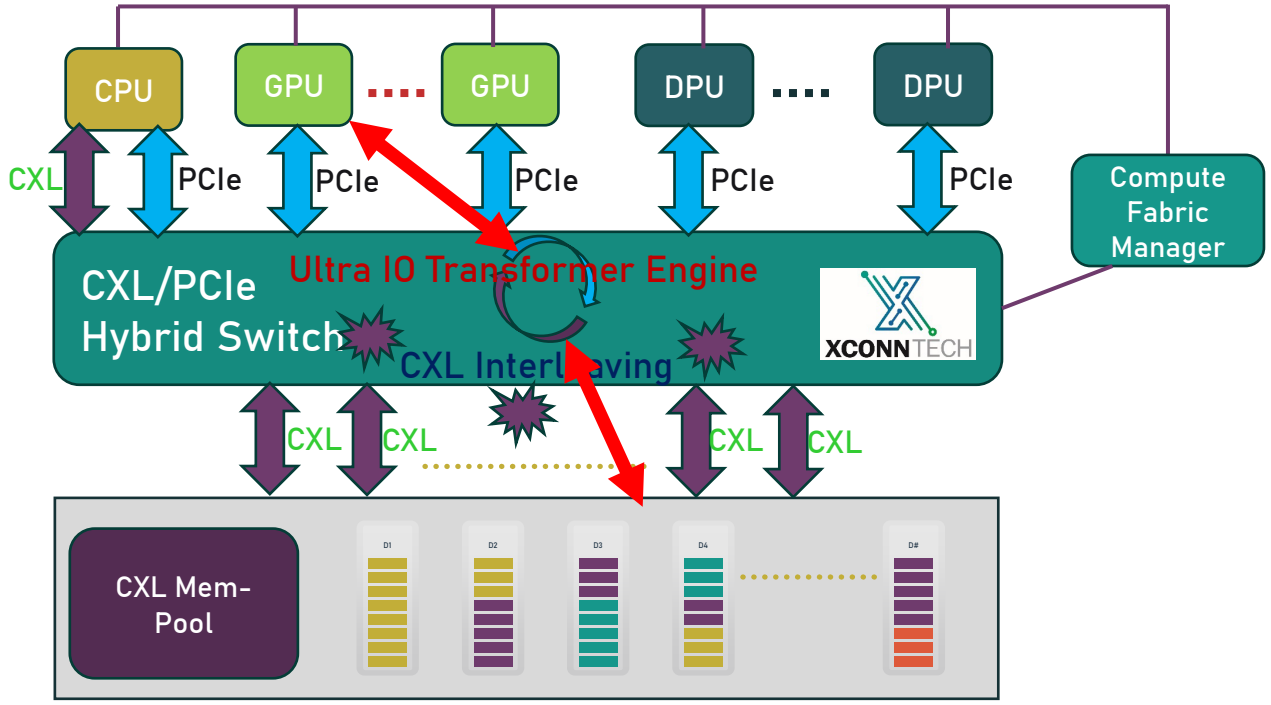
Ultra IO Transformer For GPU Direct CXL Access

Break the Memory Wall

- Majority of GPUs/Accelerators do not support CXL, PCIe is available
- AI “Memory Wall” --Large AI models require multi-TBs or more memory (Tokens, KV caching, etc.)
- XConn’s “Ultra IO Transformer” enables GPUs/DPUs (PCIe devices) to directly access CXL memory pool



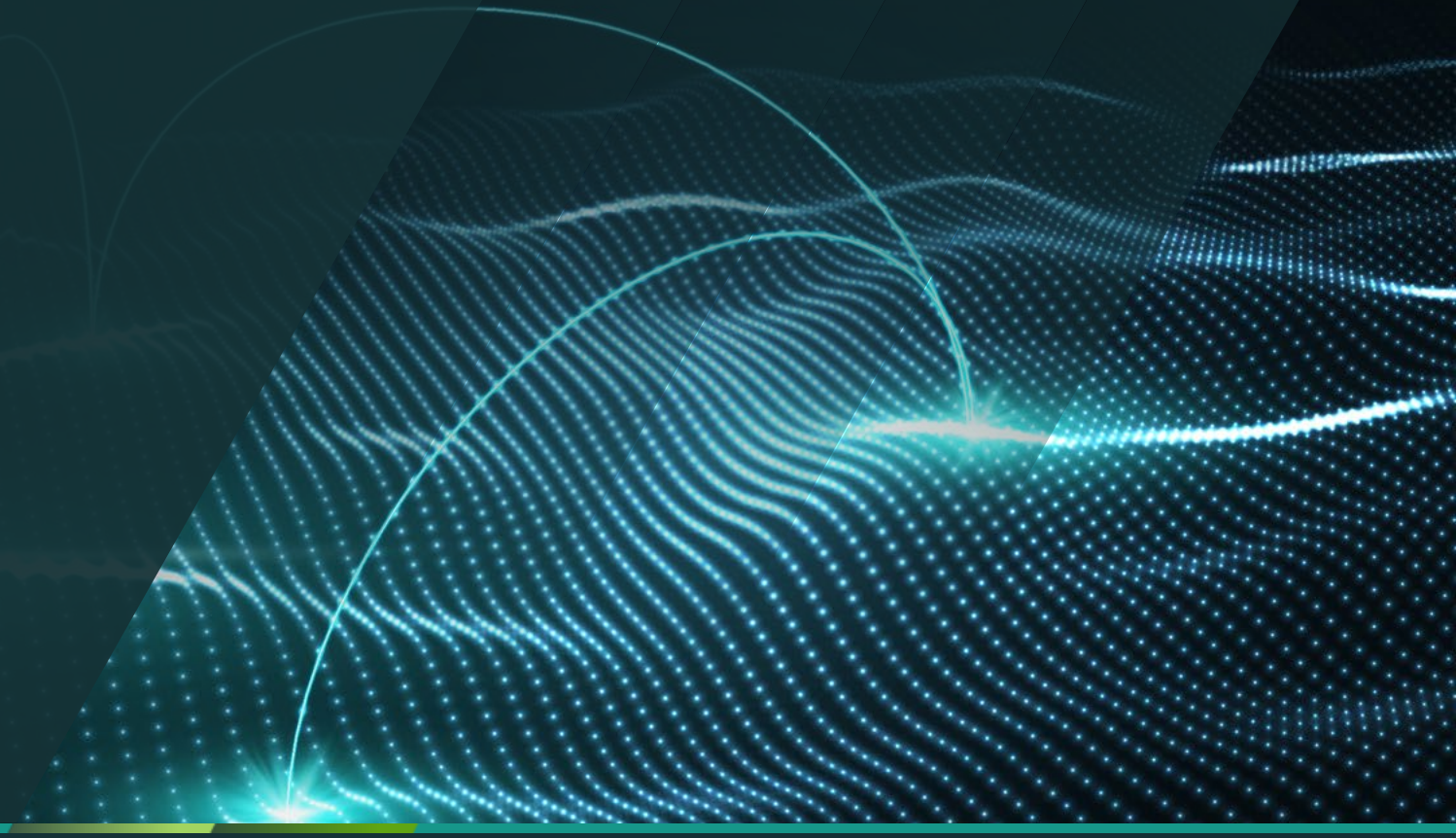
GPU access CXL through CPU



GPU-Centric Computing With Large Memory Pool Enabled by “Ultra IO Transformer” Technology

Q&A

Please share your questions in the Question Box



Thank You

www.ComputeExpressLink.org