

The Emerging Memory-Centric Data Center

Workloads, Benchmarks, KV Cache and CXL Acceleration on NVIDIA and AMD Systems

Endless Memory with CXL | Episode 2 | October 28, 2026

Where We Left Off, and Where We're Going

Episode 1 introduced CXL and the new AI memory hierarchy. Today we look at the workloads and the data.

Episode 1 Recap

A new tier

CXL adds DRAM-class memory (~500–800 ns) between local DRAM (~100 ns) and NVMe (~100,000 ns).

Three ways to deploy

Expansion inside one server → pooling across servers → sharing the same memory between servers.

Shipping today

CXL 2.0 pooling on PCIe Gen5; CXL 3.1 support arriving in NVIDIA Vera and AMD EPYC 9006 hosts.

Today's Agenda

1. **Why the data center is becoming memory-centric**
2. **KV cache explained for infrastructure buyers**
3. **CXL results on NVIDIA systems**
4. **CXL on AMD systems**
5. **Industry benchmarks and a buyer's checklist**

© 2026 LIQID Inc. All rights reserved.

2026: Inference Turns AI Into a Memory Problem

Agentic AI is changing what servers need. Memory demand and prices are rising faster than compute.

+303%

DRAM market growth in 2026

TrendForce projects $619B in 2026 as agentic AI drives demand; server DRAM contract prices are surging.

>5x / yr

Growth in output tokens per query

Reasoning and agentic models now produce tens of thousands of tokens per answer, and each one adds to the KV cache.

1:8 → 1:4

CPU-to-GPU ratio in AI servers

Agentic deployments push toward 1:1, so more host memory is needed per GPU.

~9.6 PB

KV cache storage per rack (NVIDIA CMX)

Even the GPU leader now ships a dedicated tier just to hold inference context.

Bottom line: in 2026 the cost and speed of AI inference depend on how much memory you can bring to the GPU, and how well you use it.

Sources: TrendForce press release, May 29, 2026; TrendForce, “AI Inference Drives Memory Demand,” Jul 7, 2026; NVIDIA CMX announcement, Jan 2026.

© 2026 LIQID Inc. All rights reserved.

Workloads That Outgrow the Server

The common thread: a working set bigger than one server's DRAM, high reuse, and sensitivity to latency

LLM Inference (KV Cache)	Vector and Graph RAG	Large In-Memory Databases	Graph Analytics and HPC	Virtualization
Multi-turn chat, coding agents and agentic AI. Coding-agent prompts share 92–97% of their prefix, so reuse is very high.	Embedding indexes and knowledge graphs that answer best when the whole index is in memory.	Analytics and transactional data sets measured in tens to hundreds of TB, where NVMe latency is the bottleneck.	Scientific simulation and graph traversal with random access patterns across huge data sets.	VM density capped by memory, not cores. More memory per host means more VMs per server.

Today's focus: LLM Inference & KV-Cache, the fastest-growing memory consumer in the data center.

Source: Micron, "KV Cache Scaling using Dynamo and CXL Shared Memory," CXL Consortium webinar, Oct 21, 2026 (shared-prefix data); Abaco 3.0 target applications.

© 2026 LIQID Inc. All rights reserved.

KV Cache in 60 Seconds

The KV cache is the model's working memory of the conversation, and it lives in the fastest memory available

1 Prefill

The GPU reads the whole prompt and builds the KV cache. This is the expensive, compute-heavy step that sets time to first token (TTFT).

2 Decode

Each new token reads the KV cache instead of re-reading the prompt. A bigger cache means longer contexts and more users.

3 Next turn or next user

If the KV cache is still in memory it's a cache hit: skip prefill and answer immediately. If not, recompute from scratch.

KV cache size ≈ context length × concurrent users × model size

Think of it this way:

The KV cache is the model's notes on the conversation so far. Keep the notes and the next answer starts right away. Throw them out to free up space, and the model has to reread the whole conversation from page one, burning GPU time you already paid for.

© 2026 LIQID Inc. All rights reserved.

Reuse Sets the Ceiling; Capacity Unlocks It

LIQID

Analytical model: every point of KV reuse cuts prefill work and raises concurrency at the same time

Effective reuse	TTFT speedup	Concurrency	Typical workload
50%	**2x**	2x	System prompt reuse (chat)
80%	**5x**	5x	Measured DeepSeek serving
90%	**10x**	10x	Multi-turn sessions
95%	**20x**	20x	Agentic workloads
98.7%	**~77x**	~77x	Coding agents

Reuse sets the ceiling

Maximum speedup scales as 1 / (1 − reuse).

Capacity is the first gate

Reuse only pays off if the reusable working set fits in memory. Capacity → hit rate → performance.

Bandwidth sets how close you get

Large mixture-of-experts models are the most bandwidth-sensitive.

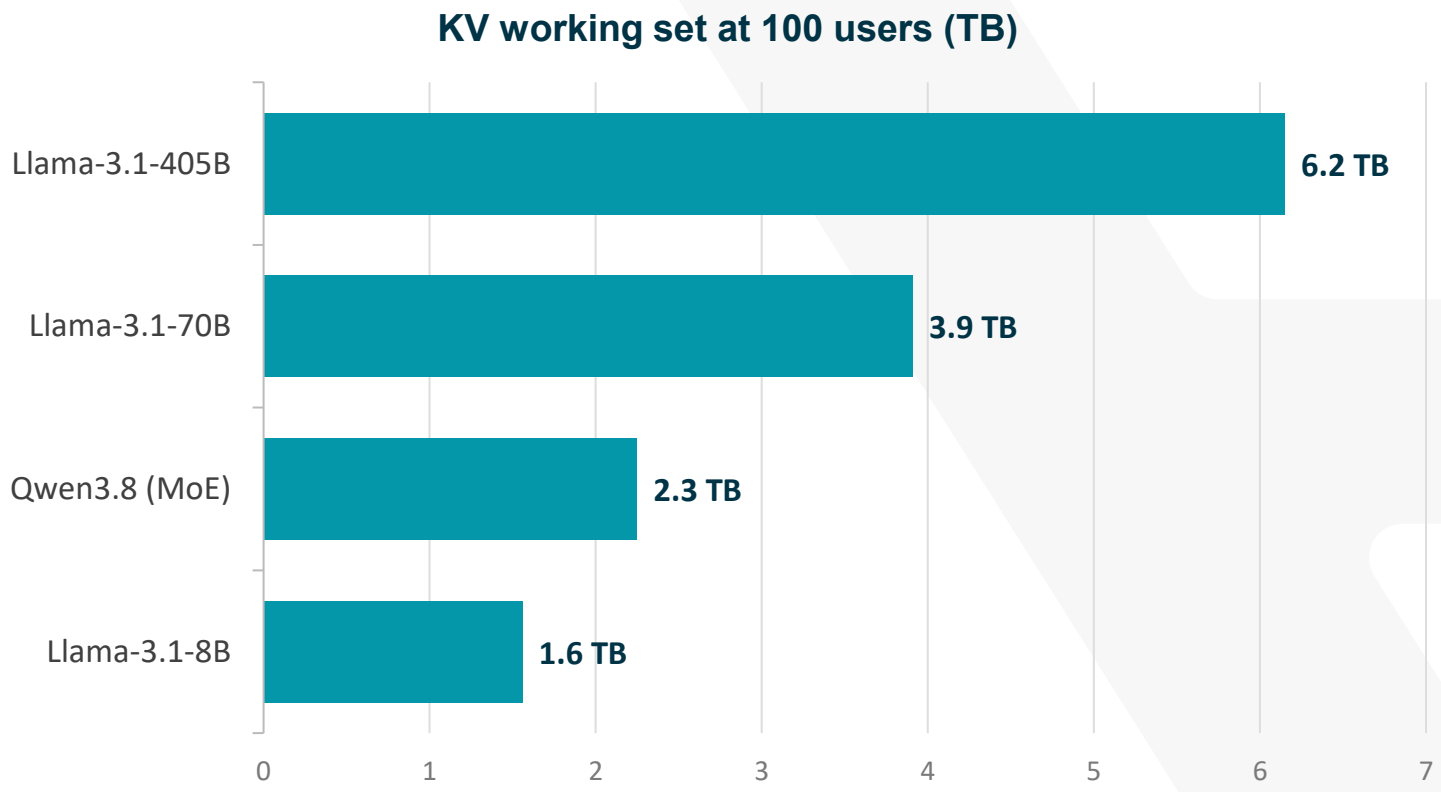
This is why memory capacity, not just GPU count, now decides inference economics.

Source: Micron (Dr. Skyler Windh), "KV Cache Scaling using Dynamo and CXL Shared Memory," CXL Consortium webinar, Oct 21, 2026. Theoretical model; results depend on hit rate. Full presentation and slides: https://computeexpresslink.org/event/kv-cache-scaling-using-dynamo-and-cxl-shared-memory/

© 2026 LIQID Inc. All rights reserved.

KV Working Sets Have Outgrown Host DRAM

KV cache needed at each model's full native context with 100 concurrent users, vs ~0.9 TB of host DRAM



Host DRAM wall: ~0.9 TB

Every model's full context window needs more KV cache than a well-equipped server holds.

Context that fits in 0.9 TB of DRAM

Llama-3.1-405B: 18K tokens

Llama-3.1-70B: 29K tokens

Llama-3.1-8B: 72K tokens

Qwen3.8 (MoE): 100K tokens

Native windows run to 128K–256K tokens, so the rest has to go to a pooled tier or be recomputed.

Per-token KV isn't what grows. Context length × concurrency is, and CXL pools scale to 100 TB+.

Source: Micron, "KV Cache Scaling using Dynamo and CXL Shared Memory," CXL Consortium webinar, Oct 21, 2026. Modeled at native context length, 100 concurrent users. Full presentation and slides: https://computeexpresslink.org/event/kv-cache-scaling-using-dynamo-and-cxl-shared-memory/

© 2026 LIQID Inc. All rights reserved.

Where KV Cache Lives on NVIDIA and AMD

Both platforms already tier KV cache beyond GPU memory. CXL adds a large, shareable DRAM-class tier between host DRAM and flash.

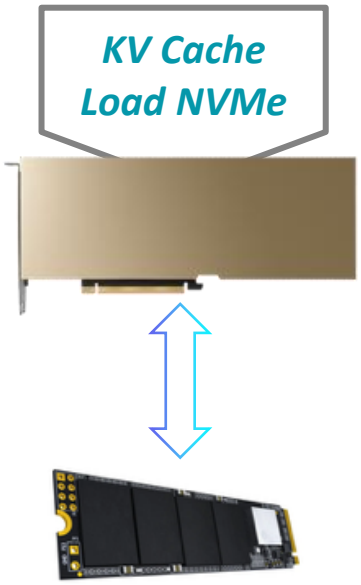
NVIDIA (Dynamo KV cache tiers)		AMD (ROCm + vLLM / LMCache)
G1 GPU memory HBM on Blackwell / Rubin GPUs	Fastest, smallest	**GPU memory** MI355X: 288 GB HBM3E
G2 Host DRAM Vera CPU: up to 1.5 TB LPDDR5X		**Host DRAM** EPYC 9006: 16 DDR5 channels, up to 1.6 TB/s
CXL memory pool (“G2.5”): ~500–800 ns • 10s–100s of TB • pooled and shared across servers • Vera and EPYC 9006 both support CXL 3.1		
G3 Local NVMe SSD offload tier		**Local NVMe** SSD offload tier
G4 Remote / CMX BlueField-4 context storage, ~9.6 PB per rack	Largest, slowest	**Networked storage** Infinity Context with storage partners

Sources: NVIDIA Dynamo KVBM design docs (G1–G4); NVIDIA CMX (Jan 2026); Tom’s Hardware, Vera specs (Jul 21, 2026); StorageReview, EPYC 9006 (Jul 23, 2026); AMD ROCm blog (Aug 28, 2026).

© 2026 LIQID Inc. All rights reserved.

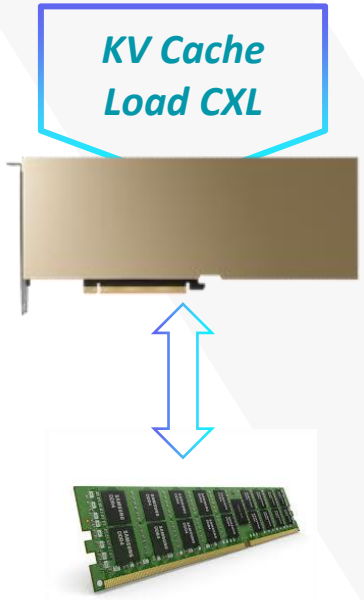
KV Cache Load Time | CXL vs. NVMe

Legacy Infrastructure

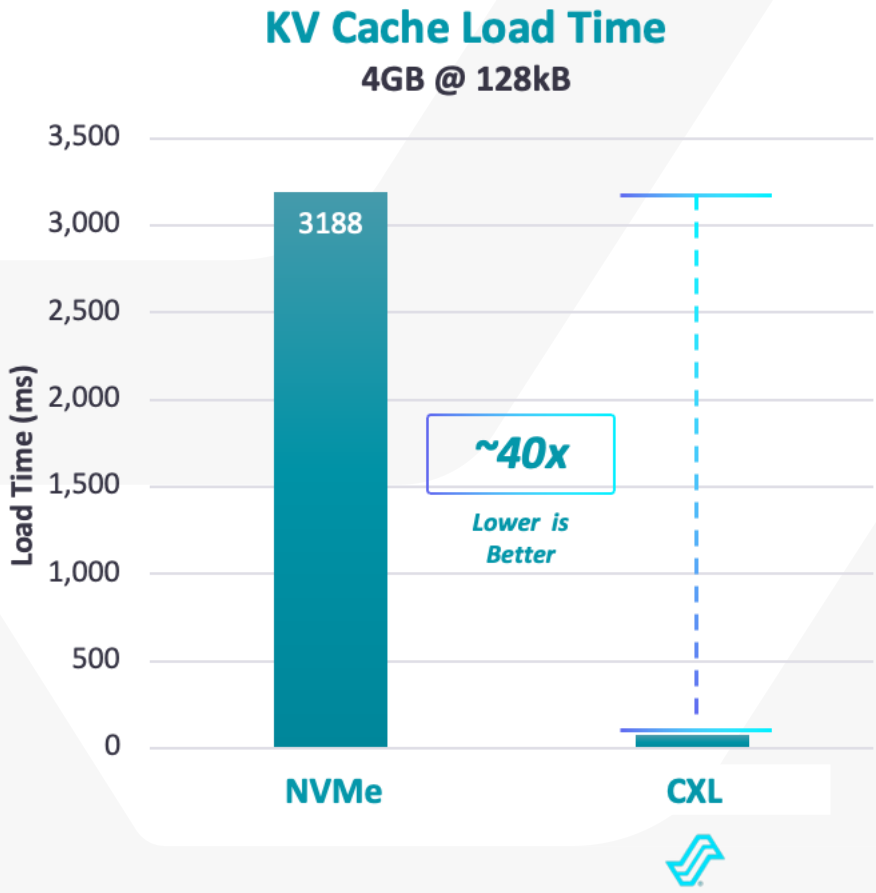


3,188ms Load Time
4GB @ 128kB

LIQID



78ms Load Time
4GB @ 128kB



© 2026 LIQID Inc. All rights reserved.

CXL for GPU Workloads

Why memory disaggregation is becoming core infrastructure for inference



Latency

CXL memory performance is comparable to DRAM, with virtually no measurable performance loss.

~0%
PERF. LOSS



Capacity

DRAM capacity at the G2.5 layer directly drives tokens/second — memory capacity is a key lever for inference performance.

160TB
DRAM POOL



Sharing

Sharing at the G2.5 layer delivered ~3x the throughput of a same-size DRAM cache, ~7x vs. stock Dynamo.

~3x
FASTER



Endurance

CXL substantially reduces NVMe writes, delivering a major endurance advantage for storage devices.

90%
FEWER WRITES



RDMA

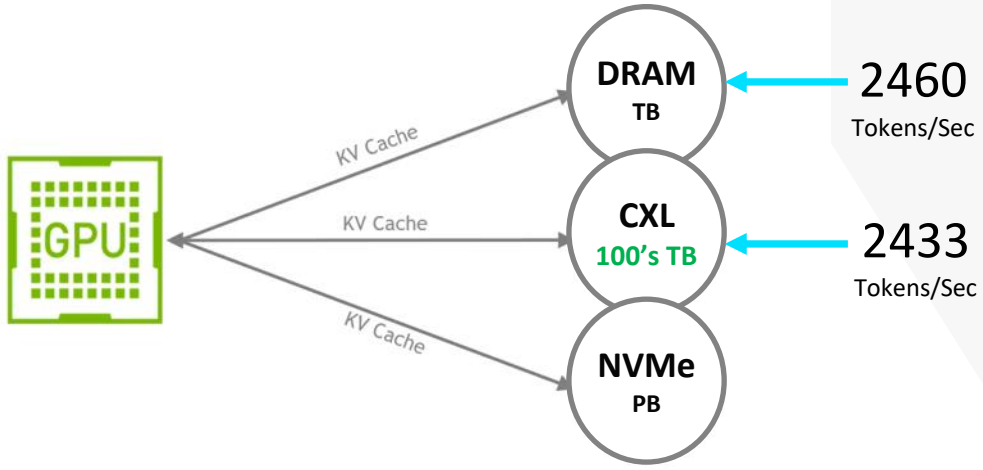
Initial G3.5 CXL-connected results validate functional operation; further optimization is needed to hit target performance.

G3.5
IN VALIDATION

© 2026 LIQID Inc. All rights reserved.

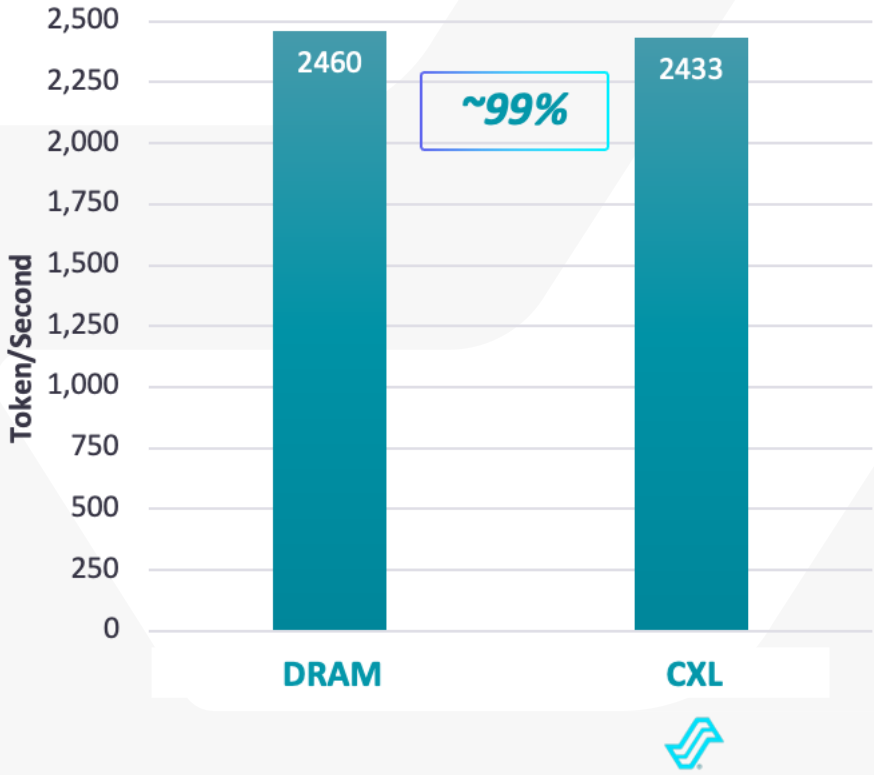
DRAM vs. CXL Performance | Nvidia AIPerf Benchmark

Performance Delta on KVCache to DRAM vs. CXL



Offload KV Cache from DRAM to CXL Memory Pools Without Performance Loss at 100x Larger Scale

NVIDIA ai-dynamo/aiperf

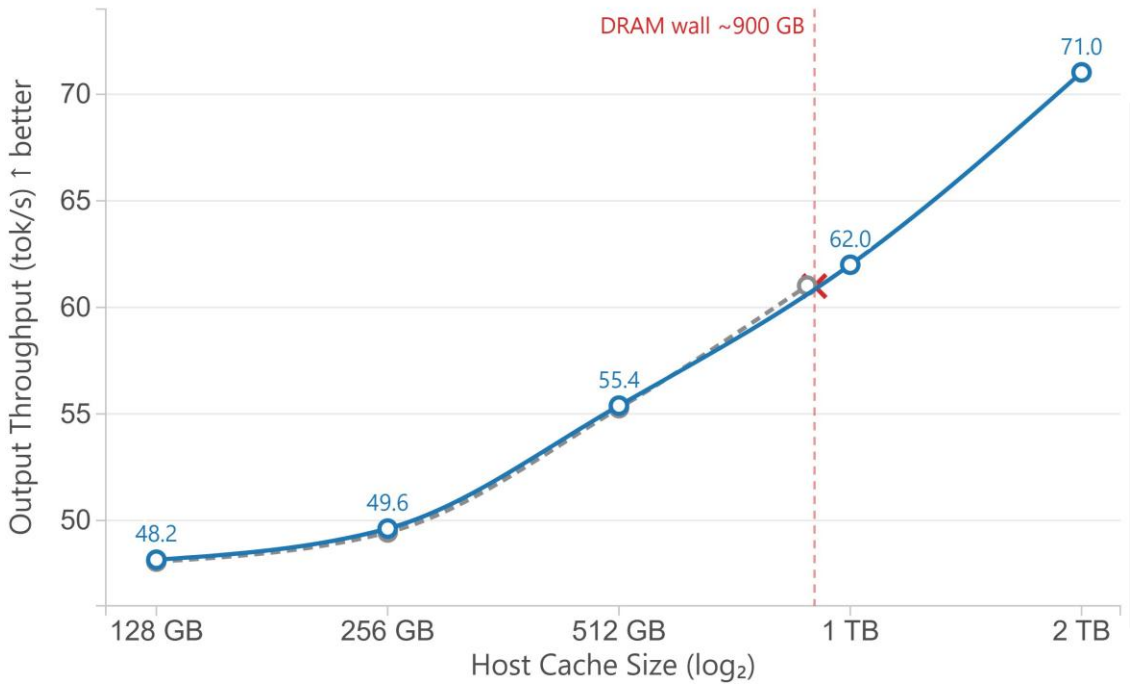


© 2026 LIQID Inc. All rights reserved.

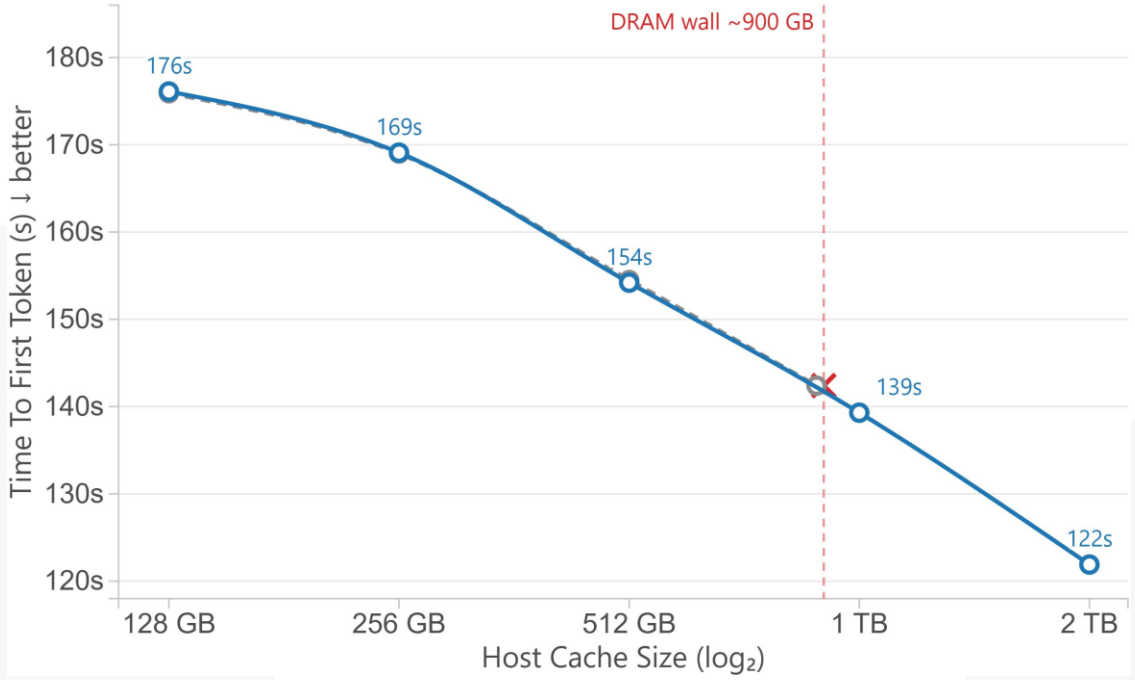
Memory Pooling Delivers DRAM-Class Performance

Larger offload pools improve performance, and CXL memory performs within 99% of host DRAM

Output Throughput vs Host Cache Size



Time To First Token vs Host Cache Size



DRAM host cache CXL host cache beyond DRAM capacity

Profile A, Llama3-70B, 256 unique groups, concurrency 64, 2,560 total requests, GPU cache on

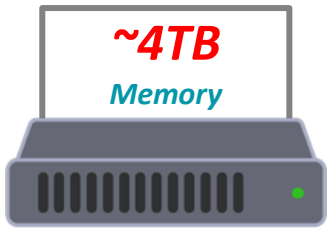
Source: Micron (Dr. Skyler Windh), "KV Cache Scaling using Dynamo and CXL Shared Memory," CXL Consortium webinar, Oct 21, 2026. Charts reproduced from the presentation. Watch the full presentation and download the slides: https://computeexpresslink.org/event/kv-cache-scaling-using-dynamo-and-cxl-shared-memory/

© 2026 LIQID Inc. All rights reserved.

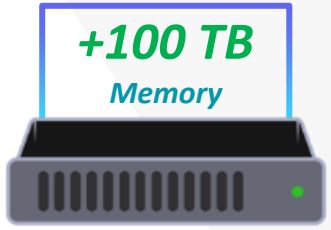
Larger Memory for Higher KV Cache Hit Rate

Memory pooling enables larger server memory for improved KV cache performance

Workload vs. KV Cache Size Requirement



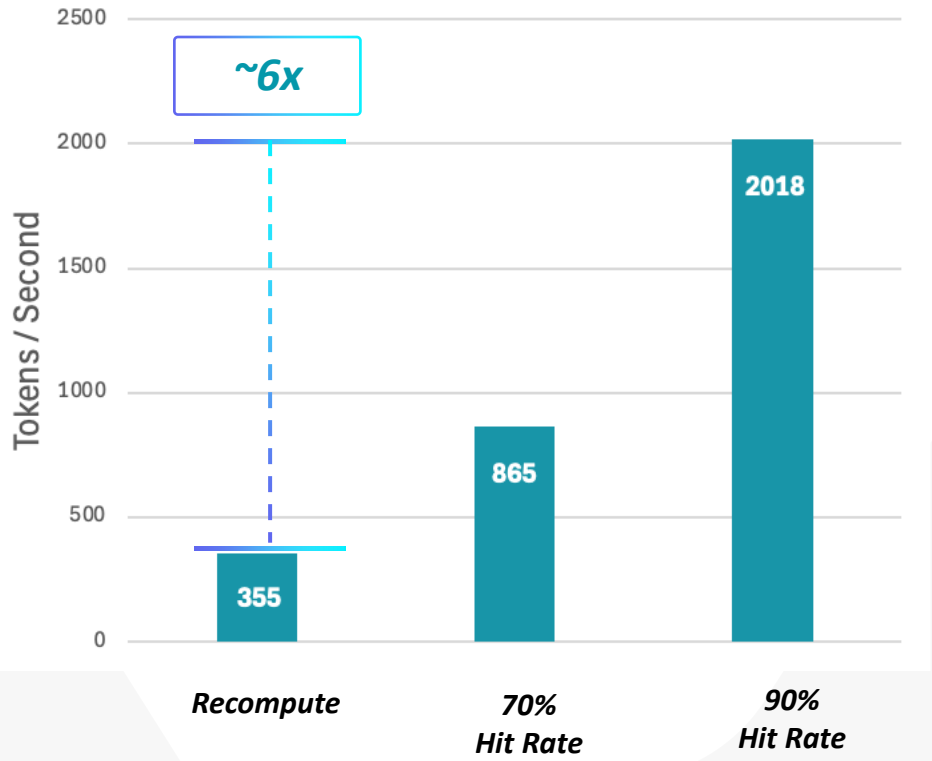
Standard Server



Memory Pooling

Scenario	Context	Blocks	Model: Llama 400B	
			KV / User	KV / 250 users
RAG / Chat	32K	2,048	15.8 GB	3.9 TB
Document Analysis	64K	4,096	31.5 GB	7.9 TB
Agentic	128K	8,192	63 GB	15.8 TB

Tokens/Second vs. KV Cache Hit Rate



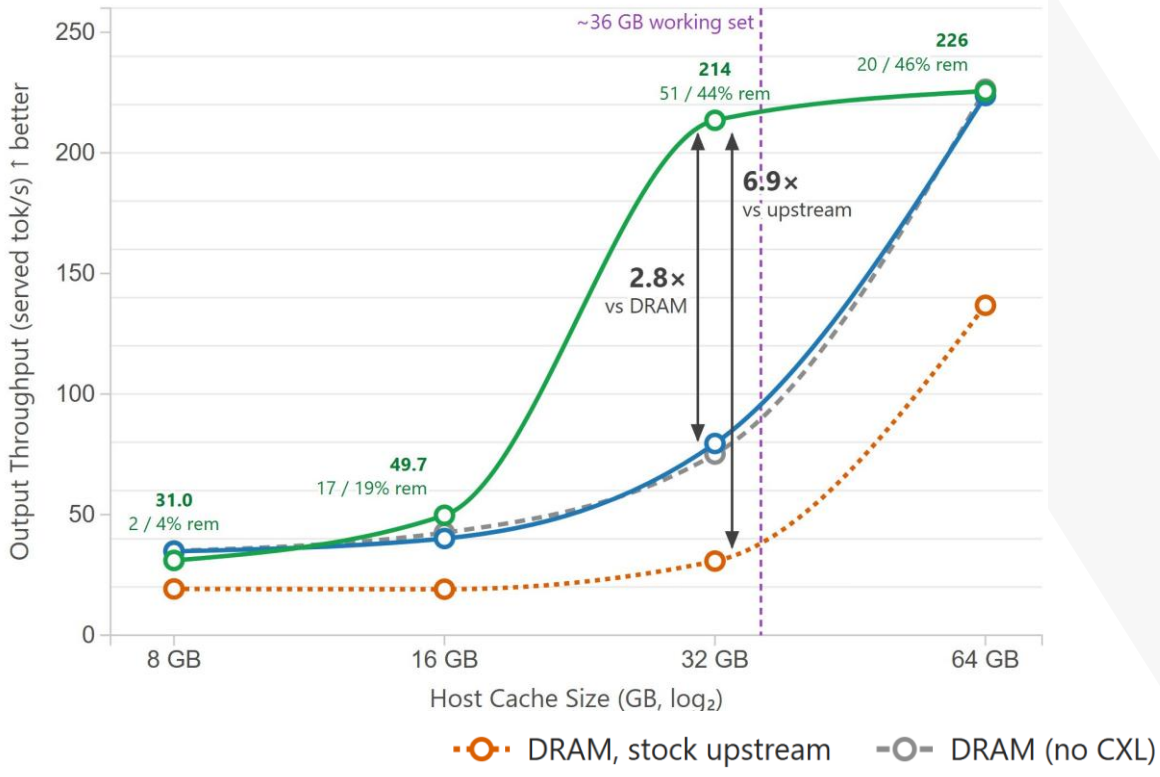
Memory Pooling Enables Higher Hit Rate for KV Cache

© 2026 LIQID Inc. All rights reserved.

Multi-Server Memory Sharing for KV Cache

Shared, reused prefixes increase effective cache capacity and improve GPU utilization - Time to first token (TTFT) approaches full cache performance early

Output Throughput vs Host Cache Size



Average TTFT vs Host Cache Size



Llama3-70B • 4 unique prefix groups • 90% shared • GPU cache off

Source: Micron (Dr. Skyler Windh), "KV Cache Scaling using Dynamo and CXL Shared Memory," CXL Consortium webinar, Oct 21, 2026. Charts reproduced from the presentation. Watch the full presentation and download the slides: https://computeexpresslink.org/event/kv-cache-scaling-using-dynamo-and-cxl-shared-memory/

© 2026 LIQID Inc. All rights reserved.

Pooling Is Plug-In. Sharing Needs Software.

Micron extended NVIDIA Dynamo to share KV blocks across servers through a Liqid CXL memory pool

4 TB
of KV cache served over CXL

3x
throughput when hosts reuse each other's blocks, with half the memory

0
vLLM patches. The Dynamo router and KV Block Manager are unchanged

86–90%
of pinned-DRAM bandwidth for a CXL tier in LMCache (Aug 2026)

Four levels of software enablement

1 Bigger private cache
No software change. Available today.

2 GPU-direct pinning
Kernel patches (Micron-built).

3 Blocks shared between hosts
Coherency + block registry. Built and measured.

4 General shared memory
Library support for all apps. CXL 3.x hardware coherency helps.

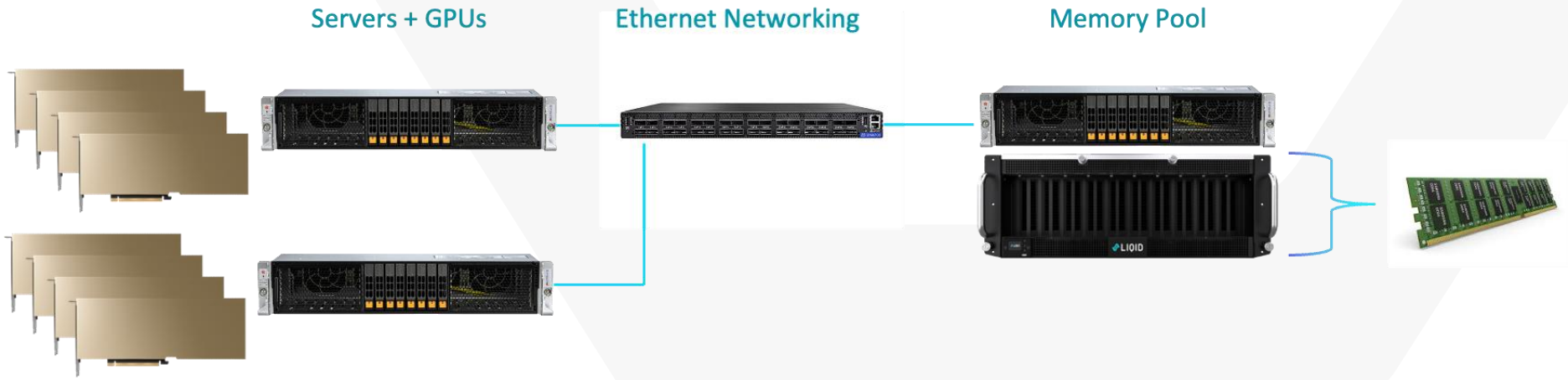
Levels 1–3 are enough to deploy the KV cache use case today. A cluster of N pools behaves like one N-times-larger pool.

Source: Micron, "KV Cache Scaling using Dynamo and CXL Shared Memory," CXL Consortium webinar, Oct 21, 2026 (test system: 2x Supermicro hosts, 4x H200 NVL, 4x Liqid CXL chassis); LMCache blog, Aug 4, 2026. Full presentation and slides: https://computeexpresslink.org/event/kv-cache-scaling-using-dynamo-and-cxl-shared-memory/

© 2026 LIQID Inc. All rights reserved.

Memory Pooling Over RDMA Networking

Initial G3.5 CXL-connected results validate functional operation; further optimization is needed to hit target performance



Basic Functionality Working

Initial RDMA-connected CXL pooling is functional end-to-end, validating the core memory-sharing path.

More Performance Tuning Required

Early results confirm the approach works; further optimization is needed to hit target throughput and latency.

Working with Micron and Nvidia

Development is a close collaboration with Micron and Nvidia, aligning the memory, compute, and networking stack.

Still in Development Phase

This capability is pre-production — architecture and results are expected to keep evolving as testing continues.

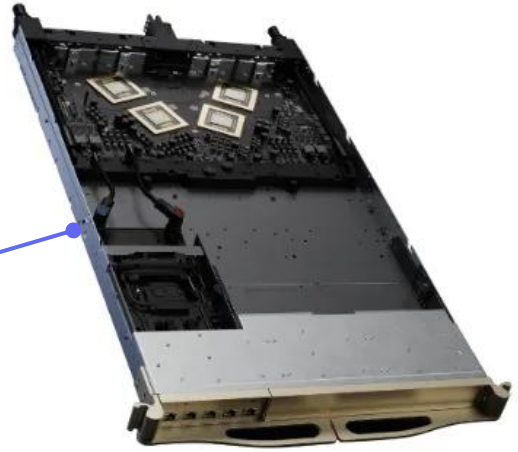
© 2026 LIQID Inc. All rights reserved.

NVIDIA Supports CXL for Vera Based Systems

NVIDIA DEVELOPER Home Blog Forums Docs Downloads Training

Technical Blog Search blog https://shorturl.at/4Kmrw

Feature	Grace CPU	Vera CPU
Cores	72 Neoverse V2 cores	88 NVIDIA Custom Olympus cores
PCIe/CXL	Gen5	Gen6/CXL 3.1



Vera Server

© 2026 LIQID Inc. All rights reserved.

AMD Systems: Ready for a CXL Memory Tier

AMD’s stack already offloads KV cache beyond HBM. CXL adds the large, shareable DRAM tier in between.

EPYC 9006 “Venice” host

- Up to 256 Zen 6 cores
- 16 DDR5 channels, up to 1.6 TB/s per socket
- First server CPU with PCIe Gen6
- CXL 3.1 for memory expansion
- Partner platforms from Q4 2026

Instinct GPUs + KV offload

- MI355X: 288 GB HBM3E per GPU
- LMCache on ROCm, 4-bit KV to host DRAM: ~2x goodput, ~2.6x lower P95 TTFT
- Infinity Context with VAST: up to 9x faster TTFT, 9.7x throughput (storage tier)

Liqid on AMD today

- UltraStack: 30x AMD MI350P GPUs on a single server
- Memory and GPU pooling over one fabric with Liqid Matrix
- Measured AIM scaling on Liqid: next slide

Sources: StorageReview, EPYC 9006 Venice (Jul 23, 2026); AMD ROCm blog, 4-bit KV caching in LMCache on MI355X (Aug 28, 2026); StorageReview, VAST Data and AMD (Jul 27, 2026).

© 2026 LIQID Inc. All rights reserved.

CXL for KV Cache: Industry Benchmarks

Memory makers, switch vendors, hyperscalers and Liqid all report the same direction of results

Source (date)	Setup	Reported result
Samsung (Jun 2026)	CXL switch pool, 8x NVIDIA RTX PRO 6000, vLLM + LMCache	**~92% of DRAM throughput; at ~700 GB of KV, 354K vs 61K tokens/s once DRAM ran out**
Micron / CXL Consortium (Oct 2026)	Dynamo, 4x H200 NVL, Liqid CXL chassis	**Within 99% of host DRAM; 3x throughput with cross-host sharing at half the memory**
Liqid + Micron (Aug 2026)	EX-5410C pool, KV cache workloads	**Up to 7x more tokens/s**
Marvell Structera (Aug 2026)	GPU memory-pooling configurations	**4.8x inference throughput; 82.7% lower TTFT**
XConn + MemVerge (Nov 2025)	CXL switch pool with NVIDIA Dynamo / NIXL	**>5x vs SSD- or RDMA-based KV offload**
Alibaba “Beluga” (Nov 2025)	CXL vs RDMA memory pool for KV cache	**7.35x throughput; ~10x lower TTFT**
SK hynix ITME (Jun 2026)	CXL-hybrid memory, NVIDIA A100, vLLM	**1.8x throughput vs NVMe-oF baseline**

Vendor- and author-reported results; configurations differ, so results are not directly comparable. Sources: Samsung white paper; Micron CXL Consortium webinar; Liqid FMS 2026 release; Futurum (Aug 5, 2026); XConn release (Nov 17, 2025); arXiv 2511.20172; arXiv 2606.12556.

© 2026 LIQID Inc. All rights reserved.

CXL Cuts NVMe Writes and Extends Drive Life

Multiple mechanisms work together to cut the total volume of writes that ever reach flash and reduce costs

Larger Cache, Less Write-Through

A DRAM-class capacity tier in front of NVMe means far more data stays resident before eviction, so fewer writes ever pass through to disk.

Write Filtering, Fewer Unnecessary Writes

Class-aware placement keeps ephemeral, low-reuse blocks in cache instead of writing them out to NVMe at all.

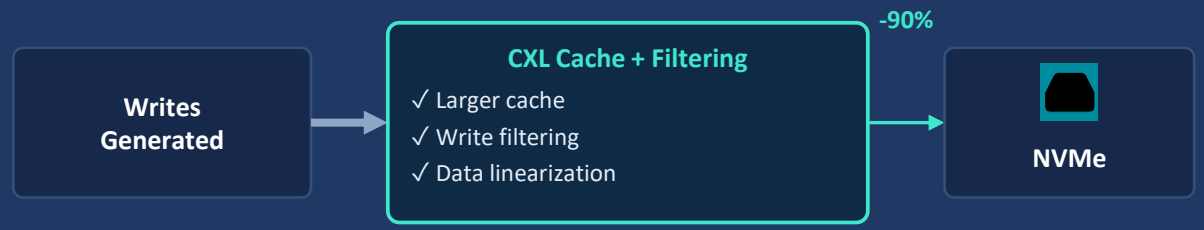
COMBINED EFFECT

Up to 90%

fewer writes reach NVMe

Bigger cache, smarter filtering, and data linearization

HOW IT WORKS



Lower Endurance Requirement

5 DWPD → 1 DWPD

Absorbing and filtering writes before they reach NVMe drops the sustained write pressure the media must tolerate — from 5 DWPD down to 1 DWPD.

Lower Media Cost

~65% lower media cost

1 DWPD NAND is dramatically cheaper than 5 DWPD NAND. The same workload can now run on lower-endurance, lower-cost media — cutting drive cost by up to ~65%.

© 2026 LIQID Inc. All rights reserved.

Buyer’s Checklist: CXL for Inference

Six questions to bring to your next infrastructure review

1 What is your KV reuse rate?

System prompts, multi-turn sessions and agents often exceed 80–90% reuse.

2 How big is your working set?

Context length × concurrent users × model size. Compare it with your host DRAM.

3 Where does KV spill today?

HBM → DRAM → NVMe → recompute. Every step down costs latency and GPU time.

4 Does your stack support a host tier?

NVIDIA Dynamo and vLLM + LMCache support DRAM-class offload without app changes.

5 Is your CPU platform CXL-ready?

Intel Xeon 6, AMD EPYC 9006 and NVIDIA Vera all support CXL.

6 How will you measure success?

Tokens per dollar, tokens per watt and P99 TTFT, not just GB added.

© 2026 LIQID Inc. All rights reserved.

Liqid on AMD: 1 to 30 MI350P GPU Scaling with AMD AIMs

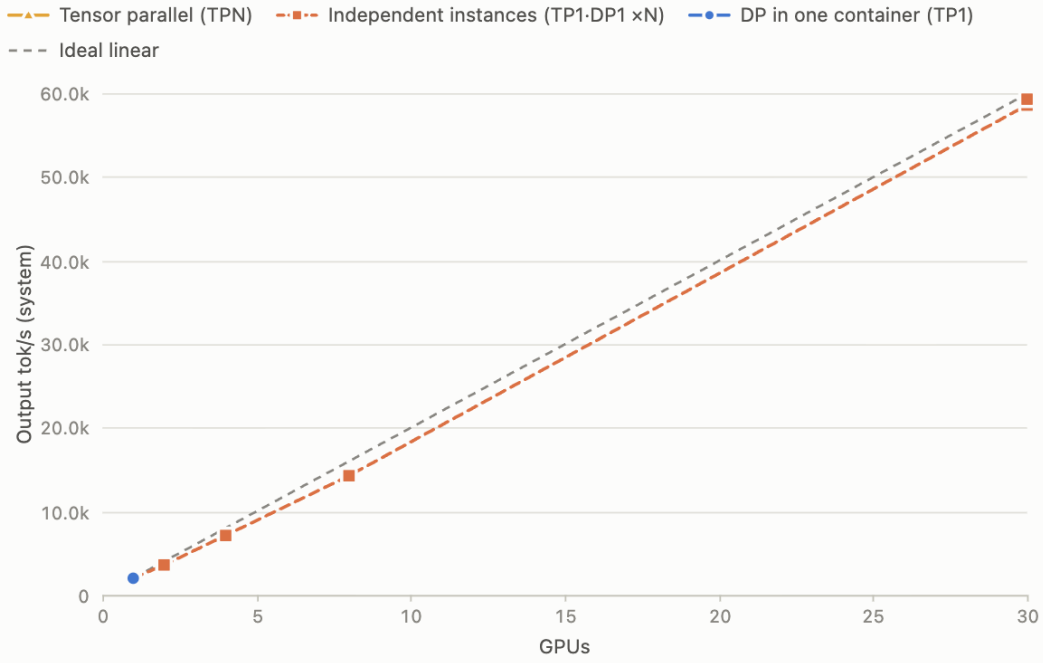
Llama 3.3 70B Instruct (FP8) — Dense, 70B params (70B active) · FP8 · max context 128k tokens · image liqid-aim-instinct-meta-llama-llama-3-3-70b-instruct:0.12.0-rc6 · 6 configurations · 45 measurements

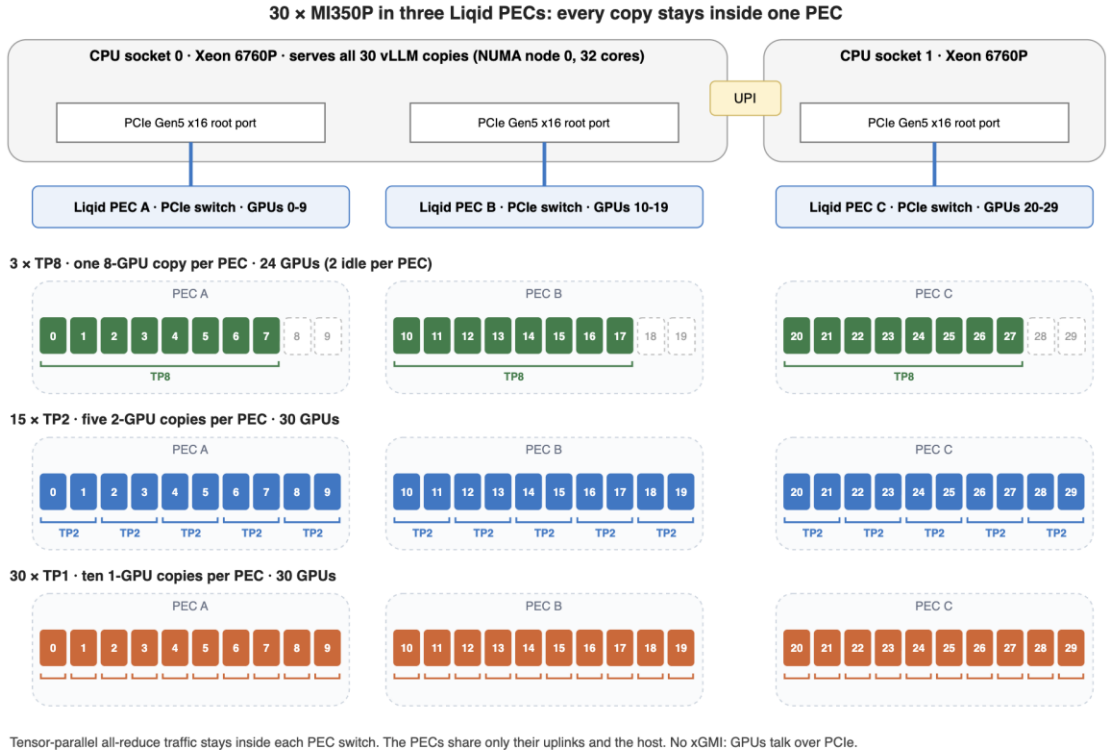
Max users at "Interactive chat" SLO	Best tok/s per GPU within SLO	Peak system output	Fastest single-user stream	Best energy efficiency
960 users	**949** tok/s/GPU	**59.3k** tok/s	**40.8** tok/s/user	**3.44** tok/J
TP1·DP1 ×30 (30 GPUs) · P99 TTFT ≤ 1,000 ms, TPOT ≤ 40 ms	TP1·DP1 (1 GPU) @ 32 users	TP1·DP1 ×30 (30 GPUs) · cpuloc, no SLO	TP1·DP1 (1 GPU) @ 1 user	TP1·DP1 (1 GPU)

Scaling with GPU count

LOAD: Peak (any load)

Total system output tokens/s as GPUs are added, per strategy. Dashed gray = perfect linear scaling from the 1-GPU result. On the line = every GPU as productive as one alone; below it = GPUs lose efficiency as they're added (per-user speed can't rise from DP; only TP or speculative decoding raise it).





SUT 2 x Intel(R) Xeon(R) 6760P, 512GB DDR5, TP=1, DP=1, vLLM instances 1-30

Liqid aimbench: : P99 Results, Interactive Chat (TPOT <= 40ms), 1K input/1K output workload

© 2026 LIQID Inc. All rights reserved.

Thank You



Contact

11400 Westmoor Circle, Suite 225

Westminster, CO 80021

Phone: 303.500.1551

Website: liqid.com

General Info: info@liqid.com

Partner Info: partner@liqid.com