Technical Report

Cross-Vendor Disaggregated Inference: GPT-OSS 120B across NVIDIA H100 and AMD MI300X

Moreh, Inc.

March 2026

MOREH

Contents

Cross-Vendor Disaggregated Inference: GPT-OSS 120B across NVIDIA H100 and AMD MI300X

The End of One-GPU-Does-Everything

AI data centers have traditionally been built around a single GPU model — buy as many of the latest NVIDIA GPUs as budget allows, deploy them identically, and let a load balancer distribute requests evenly. This approach is simple, but it is increasingly at odds with the economic reality of large-scale AI inference. No single accelerator architecture is optimal for every workload, and deploying only one type means some hardware is always over-provisioned or under-utilized.

The industry is moving toward heterogeneous systems that combine different accelerator types, each assigned to the workload it handles best. NVIDIA made this direction explicit at GTC 2026, presenting a system that pairs Vera Rubin GPUs with NVIDIA Groq 3 LPX — a rack of 256 LPU accelerators (500 MB SRAM per chip, 150 TB/s bandwidth per accelerator) — to perform inference jointly. NVIDIA states that previous inference architectures forced a choice between interactivity and throughput — “you couldn’t have all three” (interactivity, intelligence, and throughput). LPX addresses this by combining the GPU’s compute density with the LPU’s ultra-fast SRAM access in a single system, claiming up to 35× higher throughput per megawatt for trillion-parameter models.

Figure 1. NVIDIA Groq 3 LPX rack (256 LPU accelerators). In a Vera Rubin system, this rack works alongside a Rubin GPU rack — GPUs and LPUs jointly perform inference, each contributing its architectural strength. (Source: NVIDIA)

The same logic applies beyond a single vendor's product line. Real-world data centers are already mixing GPU generations (B300 alongside H200), GPU vendors (NVIDIA and AMD), and entirely different accelerator classes (GPUs and AI accelerators such as Tenstorrent processors). Each combination opens a different efficiency frontier depending on the workload.

Prefill-Decode Disaggregation

Among the various techniques for leveraging heterogeneous accelerators, the most representative is prefill-decode disaggregation (PD disaggregation). LLM inference consists of two computationally distinct phases. The prefill phase processes the entire input prompt in parallel through dense matrix multiplications and is compute-bound. The decode phase generates output tokens one at a time in an autoregressive manner, reading model parameters and the KV cache from GPU memory at each step, and is memory-bandwidth-bound. These two phases have fundamentally different hardware requirements.

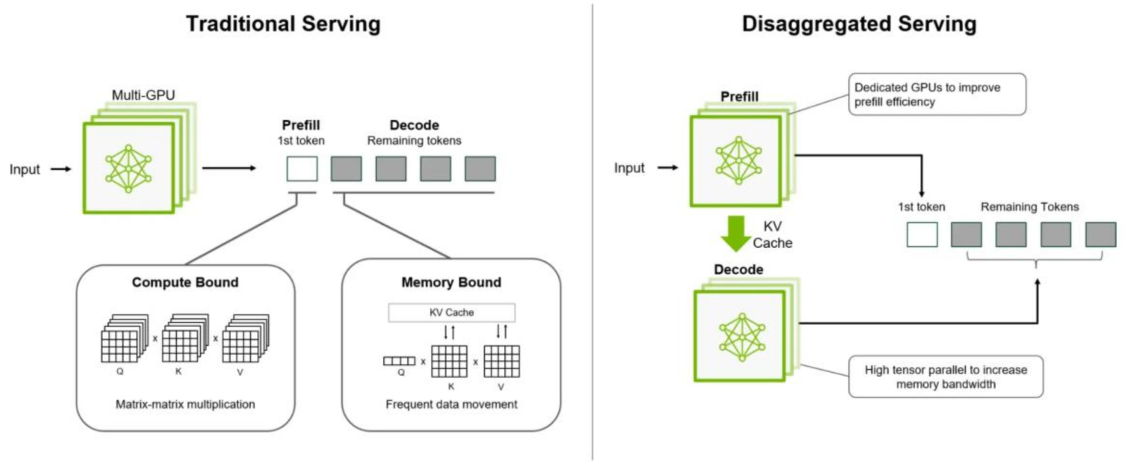

Figure 2. Traditional serving (left) vs. disaggregated serving (right). Separating compute-bound prefill from memory-bound decode onto dedicated GPUs eliminates interference between the two phases. (Source: NVIDIA)

PD disaggregation separates these phases onto dedicated server nodes, assigning each GPU to the role that best matches its hardware characteristics. A prefill node processes input prompts at high throughput and transfers the resulting KV cache to a decode node over a high-speed and low-latency network. The decode node handles only token generation, maintaining low and predictable inter-token latency. Without this separation, mixing different GPUs in a cluster offers limited benefit — each node still runs both phases, and the weaker phase becomes the bottleneck.

The Challenge of Cross-Vendor Disaggregation

PD disaggregation within a single vendor's ecosystem is already supported by frameworks such as vLLM, SGLang, and NVIDIA Dynamo. However, extending it across vendor boundaries — for example, running prefill on NVIDIA GPUs and decode on AMD GPUs — introduces a distinct set of engineering challenges that no existing open-source framework has solved.

The most fundamental barrier is the absence of a unified cross-vendor communication layer. NVIDIA GPUs rely on NCCL and AMD GPUs on RCCL, and these libraries do not interoperate. KV cache transfer between a prefill node and a decode node from different vendors cannot use GPUDirect RDMA for direct GPU-to-GPU memory access; instead, data must be staged through host CPU memory, adding latency on the critical path. Furthermore, the two vendor ecosystems use entirely different software stacks (CUDA vs. ROCm), requiring separate kernel implementations, separate model compilation, and careful management of KV cache memory formats to ensure compatibility across architectures. Finally, because different GPU architectures have different compute-to-bandwidth ratios, the framework must understand the performance characteristics of each architecture to optimally balance prefill and decode workloads across the cluster.

MoAI Inference Framework: Cross-Vendor Disaggregation

The MoAI Inference Framework is the only production-grade inference framework that supports cross-vendor PD disaggregation — routing prefill and decode to GPUs from different vendors within a single serving cluster. While NVIDIA Dynamo, vLLM, and SGLang each support PD disaggregation on their respective platforms, none extend this capability across vendor boundaries.

MoAI addresses the cross-vendor challenges through a vendor-neutral abstraction layer that handles the differences in software stacks and architecture-specific performance characteristics, along with a communication library that enables RDMA-based KV cache transfer between GPUs from different vendors. Data center operators can break free from single-vendor lock-in, allocate hardware budgets across multiple suppliers, and assign each vendor's GPUs to the workload phase where they deliver the best performance.

In this report, we benchmark one specific cross-vendor combination: NVIDIA H100 for prefill and AMD Instinct MI300X for decode, serving the GPT-OSS-120B model. Across four ISL/OSL scenarios, the cross-vendor configuration achieves 8–9% better geomean end-to-end latency and throughput compared to a single-vendor MI300X cluster, with gains of up to 43% in latency and 67% in throughput under the most demanding workloads.

System Architecture

NVIDIA H100 and AMD Instinct MI300X have fundamentally different hardware profiles, which make them suited to different phases of inference.

	NVIDIA H100 SXM	AMD Instinct MI300X	MI300X / H100
HBM Capacity	80 GB (HBM3)	192 GB (HBM3)	2.4×
Memory Bandwidth	3.35 TB/s	5.3 TB/s	1.58×
FP8 TFLOPS	1979	2615	1.32×
L1 + Scratchpad	256 KB per SM	32 KB L1D + 64 KB LDS per CU	0.38×

The MI300X provides 2.4× the HBM capacity and 1.58× the memory bandwidth of the H100, giving it a structural advantage for the memory-bandwidth-bound decode phase. Conversely, the prefill phase is dominated by large GEMMs where the H100's larger L1 and scratchpad memory (256 KB per SM vs. 96 KB per CU) allow it to sustain higher compute utilization despite lower peak TFLOPS. Based on these characteristics, we assign H100 to prefill and MI300X to decode.

Two configurations were tested, each with one dedicated prefill node and one dedicated decode node:

- Cross-vendor (heterogeneous): H100 node for prefill, MI300X node for decode. Moreh implemented a custom cross-vendor communication layer that transfers KV cache over RDMA. The current implementation stages data through host memory; a future release will add GPUDirect RDMA support to enable direct GPU-to-GPU transfer across vendor boundaries.
- Single-vendor (homogeneous): MI300X node for prefill, MI300X node for decode. KV cache transfer uses the NIXL connector with GPUDirect RDMA enabled, allowing direct GPU-to-GPU memory transfer via the NIC without CPU staging.

The backend inference engine is Moreh vLLM on AMD MI300X nodes and vLLM on the NVIDIA H100 node. Despite the cross-vendor configuration using a host-staged transfer path rather than hardware-accelerated GPUDirect RDMA, computation-communication overlapping techniques effectively hide the transfer latency, especially under heavy workloads.

Experimental Setup

Three independent server nodes were used: one NVIDIA H100 node (prefill) with 8× H100 80 GB SXM GPUs, and two AMD MI300X nodes (one for decode, one for prefill in single-vendor tests) with 8× MI300X 192 GB OAM GPUs each. All nodes were connected via 200 Gbps ConnectX-6 NICs.

Category	H100 Node	MI300X Node
CPU	2× AMD EPYC 9654 (96-core, 2.4 GHz)	2× AMD EPYC 9474F (48-core, 3.6 GHz)
Memory	1,536 GB	2,304 GB
GPU	8× NVIDIA H100 80 GB SXM	8× AMD Instinct MI300X 192 GB OAM
NIC	ConnectX-6 (200 Gbps)	ConnectX-6 (200 Gbps)
OS	Ubuntu 22.04.3 LTS	Ubuntu 22.04.4 LTS
Model	GPT-OSS-120B	GPT-OSS-120B
Precision	MXFP4	MXFP4
Parallelism	Tensor Parallelism (TP=8)	Tensor Parallelism (TP=8)
Backend Engine	vLLM 0.15.0	Moreh vLLM

The target model is OpenAI's GPT-OSS-120B, a sparse Mixture-of-Experts (MoE) model with approximately 116.8 billion total parameters and roughly 5.1 billion active parameters per token. Inference was run with MXFP4 quantization and tensor parallelism TP=8, so each node's 8 GPUs form a single model pipeline. Prefix caching was disabled to isolate the raw compute efficiency of each hardware configuration.

Four ISL/OSL (input sequence length / output sequence length) scenarios were tested: 1K/1K, 1K/8K, 8K/1K, and 8K/8K. Concurrency levels ranged from 1 to 32 for most scenarios, with the 8K/1K scenario extended up to 256 to observe behavior under heavy load. The request rate was fixed at REQ_RATE=8 across all experiments. Two warmup iterations preceded each measurement to eliminate cold-start effects from memory allocation, GPU kernel initialization, and KV cache connector handshakes.

Results

The tables below compare cross-vendor (H100 prefill + MI300X decode) and single-vendor (MI300X prefill + MI300X decode) performance. E2EL is the median end-to-end latency in seconds; TPS is the total throughput in tokens per second. E2EL Ratio and TPS Ratio are cross-vendor / single-vendor — E2EL ratio below 1.0 and TPS ratio above 1.0 indicate cross-vendor advantage.

Highlight: Cross-Vendor Advantage Under Heavy Workloads

Under demanding workloads with long sequences and high concurrency, cross-vendor disaggregation delivers substantial improvements over the single-vendor baseline.

ISL/OSL	CON	Cross E2EL (s)	Single E2EL (s)	E2EL Ratio	Cross TPS	Single TPS	TPS Ratio
8K/1K	256	190.52	256.62	0.74×	12,107	9,030	1.34×
8K/8K	16	119.24	207.67	0.57×	2,190	1,312	1.67×
8K/8K	32	214.75	324.80	0.66×	2,417	1,540	1.57×
Geomean				**0.65×**			**1.52×**

At ISL 8K / OSL 8K, the cross-vendor configuration reduces E2EL by 34–43% and increases throughput by 57–67% at concurrency 16–32. These gains come from assigning each GPU to the inference phase that best matches its hardware characteristics, combined with a communication layer that manages KV cache transfer across vendor boundaries. The following sections present full results across four ISL/OSL scenarios and a range of concurrency levels, showing when and why cross-vendor disaggregation provides an advantage.

ISL 1024 / OSL 1024

With both input and output at 1,024 tokens — the lightest workload — the two configurations show virtually identical performance across all concurrency levels.

CON	Cross E2EL (s)	Single E2EL (s)	E2EL Ratio	Cross TPS	Single TPS	TPS Ratio
1	5.32	5.32	1.00×	381	378	1.01×
4	6.42	6.39	1.00×	1,165	1,234	0.94×
8	7.28	7.30	1.00×	2,139	2,175	0.98×
16	9.09	9.24	0.98×	3,402	3,333	1.02×
32	11.66	11.39	1.02×	5,198	5,086	1.02×
Geomean			**1.00×**			**1.00×**

With only 1,024 input tokens, prefill completes quickly on either GPU and the per-request KV cache is only tens of megabytes. Since neither prefill computation nor KV cache transfer is a bottleneck, performance is determined almost entirely by decode speed — which is identical across configurations. This confirms that cross-vendor operation introduces no inherent penalty.

ISL 1024 / OSL 8192

With short input but long output (8,192 tokens), the decode phase dominates wall-clock time. The cross-vendor advantage emerges from concurrency 4 and widens steadily, reaching a 29% E2EL reduction and 46% throughput improvement at concurrency 32. The long 8K output sequence amplifies the decode-side congestion effect: under the single-vendor configuration, unmoderated KV cache bursts via the NIXL connector inflate inter-token latency across the entire output generation.

CON	Cross E2EL (s)	Single E2EL (s)	E2EL Ratio	Cross TPS	Single TPS	TPS Ratio
1	45.05	44.13	1.02×	204	208	0.98×
4	53.67	58.51	0.92×	686	626	1.10×
8	67.06	74.49	0.90×	1,093	986	1.11×
16	93.82	92.41	1.02×	1,639	1,489	1.10×
32	108.74	152.61	0.71×	2,508	1,722	1.46×
Geomean			**0.91×**			**1.14×**

ISL 8192 / OSL 1024

With long input (8,192 tokens) and short output, prefill accounts for a larger fraction of total latency. At low concurrency (1–16), prefill is more memory-bandwidth-bound, and the MI300X's 5.3 TB/s bandwidth gives the single-vendor configuration an edge. A clear crossover occurs at concurrency 32: as prefill becomes compute-bound, the H100's advantage takes effect, and the cross-vendor configuration maintains its lead through concurrency 256 (26% E2EL improvement, throughput sustaining over 12,000 tok/s vs. below 10,000 tok/s for single-vendor).

CON	Cross E2EL (s)	Single E2EL (s)	E2EL Ratio	Cross TPS	Single TPS	TPS Ratio
1	6.52	6.02	1.08×	1,409	1,509	0.93×
4	9.00	7.60	1.18×	3,854	4,727	0.82×
8	11.72	9.65	1.21×	6,077	7,095	0.86×
16	17.33	14.62	1.19×	8,794	9,597	0.92×
32	24.93	36.54	0.68×	11,486	8,105	1.42×
64	47.76	57.00	0.84×	11,870	9,978	1.19×
128	101.58	119.45	0.85×	11,224	9,598	1.17×
256	190.52	256.62	0.74×	12,107	9,030	1.34×
Geomean			**0.95×**			**1.06×**

ISL 8192 / OSL 8192

The heaviest workload combines the effects of the previous two scenarios. At concurrency 1, the single-vendor configuration is slightly faster due to the MI300X's bandwidth advantage at low concurrency. From concurrency 8 onward, the cross-vendor configuration pulls decisively ahead — at concurrency 16–32, it is 34–43% faster in E2EL and delivers 57–67% higher throughput. The dual pressure of long input and long output amplifies both the H100's compute-bound prefill advantage and the decode-side congestion effect.

CON	Cross E2EL (s)	Single E2EL (s)	E2EL Ratio	Cross TPS	Single TPS	TPS Ratio
1	52.67	47.88	1.10×	311	342	0.91×
4	73.47	74.77	0.98×	890	927	0.96×
8	87.13	110.40	0.79×	1,595	1,183	1.35×
16	119.24	207.67	0.57×	2,190	1,312	1.67×
32	214.75	324.80	0.66×	2,417	1,540	1.57×
Geomean			**0.80×**			**1.25×**

Key Findings

Across all four ISL/OSL scenarios, the cross-vendor setup (H100 prefill + MI300X decode) achieved a geomean E2EL ratio of 0.92× and a geomean TPS ratio of 1.10× relative to the single-vendor MI300X cluster (E2EL ratio below 1.0 and TPS ratio above 1.0 indicate cross-vendor advantage). The key takeaways are:

- **Feasibility confirmed:** Cross-vendor PD disaggregation between NVIDIA and AMD GPUs works reliably across all tested workloads. The MoAI Inference Framework abstracts away the hardware differences, allowing operators to mix GPU vendors in a single serving cluster without compatibility constraints.
- **Performance parity at low load:** For lightweight workloads (1K/1K) and low concurrency, the two configurations deliver virtually identical performance, confirming that cross-vendor operation introduces no inherent penalty.
- **Cross-vendor advantage at high concurrency:** For workloads with long output sequences (1K/8K, 8K/8K) or high concurrency (8K/1K at CON $\geq$ 32), the cross-vendor configuration outperforms the single-vendor setup by up to 43% in latency and 67% in throughput. Two factors contribute: (1) as concurrency rises, prefill becomes compute-bound, and the H100's larger on-chip memory (L1 + shared memory) handle dense GEMMs more efficiently; (2) the cross-vendor communication layer's software-based transfer buffering prevents decode-node queue saturation that occurs with the NIXL connector's direct GPU memory path.
- **Complementary hardware strengths:** At low concurrency, prefill is more memory-bandwidth-bound, and the MI300X's 5.3 TB/s bandwidth gives it an edge. At higher concurrency, prefill shifts to compute-bound, where the H100's compute density provides the advantage. The MI300X's 192 GB HBM3 and high bandwidth make it consistently well-suited for the memory-bandwidth-bound decode phase.

- **Workload-dependent optimization is critical:** Performance characteristics vary dramatically across sequence lengths and concurrency levels — a configuration that trails at low concurrency can lead by up to 43–67% at high concurrency. Extracting the full benefit of heterogeneous hardware requires dynamic tuning of prefill/decode assignment, KV cache transfer strategy, and request routing based on real-time workload patterns. The MoAI Inference Framework aims to automate this optimization, freeing operators from manual tuning that would otherwise require deep knowledge of each GPU architecture and workload interaction.

Conclusion

This study demonstrates that heterogeneous GPU clusters can serve LLMs as effectively as — and in many scenarios more effectively than — single-vendor configurations. By assigning NVIDIA H100 GPUs to the compute-intensive prefill phase and AMD MI300X GPUs to the memory-bandwidth-intensive decode phase, the MoAI Inference Framework enables data center operators to move beyond single-vendor lock-in and design GPU infrastructure based on workload characteristics rather than vendor constraints.

Under the most demanding workloads, the cross-vendor configuration reduces end-to-end latency by up to 43% and increases throughput by up to 67% compared to a single-vendor MI300X cluster. At the same time, the results show that performance characteristics shift substantially depending on sequence length and concurrency — what works best under light load may not be optimal under heavy load, and vice versa. Navigating this complexity manually is impractical at scale. The MoAI Inference Framework addresses this by automating the workload-aware assignment of GPU resources, KV cache transfer strategies, and request routing across heterogeneous clusters, enabling operators to capture the performance benefits of cross-vendor disaggregation without the operational burden of continuous manual tuning.

MOREH

To learn more, please visit our website (https://moreh.io) or contact us (contact@moreh.io).

The information contained herein is for informational purposes only and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions, and typographical errors, and Moreh, Inc. is under no obligation to update or otherwise correct this information. Moreh, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and assumes no liability of any kind for the consequences or use of such information or for any infringement of patents. Moreh, Inc. reserves the right to make corrections, modifications, enhancements, improvements, and other changes to this information, at any time and/or to discontinue any service without notice.