LPDDR at Scale: Enabling Efficient LLM Inference Through High-Capacity Memory

Authors: Khayam Anjam, Sujit Somandepalli, Sudharshan S. Vazhkudai, Praveen Vaidyanathan

Introduction

Capacity is a bottleneck
Large context and highly concurrent LLM inference are significantly constrained by memory capacity. Having sufficient capacity can help in the reuse of KV pairs obviating the need to recompute them.

Unified System Memory Architecture enables scalability
Combining LPDDR with HBM in a cache-coherent design expands total memory, reduces latency, and improves throughput while maintaining software simplicity.

Scaling context and concurrency with high-capacity LPDDR
Offloading large KV caches to 1.5TB of CPU attached memory enabled using 192GB LPDDR5X SOCAMM2 modules reduces time-to-first-token by up to **98%** for long-context real-time inference and supports **2X** larger concurrent requests (batch size) for batched inference.

Continued LPDDR scaling will be critical for supporting even larger context windows and batch sizes in future LLM deployments.

The rapid evolution of large language models (LLMs) has made memory capacity a significant bottleneck in datacenters, alongside compute and bandwidth limitations. Inference workloads, particularly those with large context windows and high concurrency, require extensive key-value (KV) caches, placing significant pressure on memory systems. KV cache is a memory optimization strategy used in LLMs to enhance inference efficiency by storing previously computed keys and values, thereby reducing redundant computations. Systems optimized for computation and memory bandwidth now struggle to meet these capacity demands introduced by large context windows.

A wide array of memory solutions are available to scale capacity in systems such as higher capacity DDR5 RDIMMs, DDR5 MRDIMMs, and what Micron has been focused on over the last 5 years with LPDRAM-based memory for datacenter applications. LPDRAM offers the most power and space efficiency with improved bandwidth when compared to mainstream DDR5 RDIMMs. LPDRAM offers a scalable, energy-efficient solution for KV cache storage, enabling higher throughput and lower latency by complementing high-bandwidth memory (HBM). Unified system memory architectures that integrate HBM and LPDRAM address the dual challenge of bandwidth and capacity. In such a setup, while HBM delivers exceptional bandwidth, LPDRAM offers low-power and low-latency capacity expansion, making it ideal for KV cache offload. Micron's blog and associated white paper on the role of low-power (LP) memory in data center workloads offers more on this topic.[1]

By increasing LPDRAM capacity, organizations can achieve substantial improvements in throughput and latency for their LLM workloads, especially for large-context and high-concurrency workloads. This strategy enables better resource utilization, reduces energy consumption, and optimizes overall TCO, making it an attractive option for enterprises seeking to balance performance with sustainable growth.

NVIDIA's current and next-generation systems exemplify this approach, combining HBM and LPDRAM in a cache-coherent architecture that expands total memory. While DDR5 can also provide capacity expansion in systems, LPDRAM is engineered to deliver high capacity and performance while consuming significantly less power than DDR5 memory, making it an ideal choice for scaling capacity in energy-conscious data center environments.

[1] Micron white paper: The role of low-power (LP) memory in data center workloads (2025)

This paper presents empirical evidence and architectural recommendations for deploying LPDRAM as a second memory tier in LLM inference, achieving a balanced trade-off across bandwidth, capacity, power, and TCO.

A critical insight emerging from our analysis is that KV cache growth, not the model's parameter size, is now the dominant driver of memory demand in LLM inference. As context lengths expand into millions of tokens, KV cache requirements grow to the order of hundreds of gigabytes or terabytes, overwhelming traditional system memory configurations. This shift fundamentally changes how capacity must be provisioned and underscores the need for significantly larger, on-node LPDRAM tiers.

KV Cache Offload for Scalable LLM Inference

KV cache is a memory optimization strategy commonly employed in LLMs to improve the efficiency of inference processes. By storing previously computed attention keys and values, the KV cache helps prevent redundant computations for tokens that have already been processed. This enables the model to quickly retrieve relevant information during inference, thereby accelerating token generation and reducing overall computational workload.

The efficient use of KV cache is crucial in LLMs, as it allows faster response times and supports longer context lengths without overwhelming system resources. By minimizing unnecessary recalculation, KV cache not only boosts performance but also contributes to better resource utilization and reduced energy consumption in data center environments.

Offloading this KV cache means moving the KV cache data blocks from high-bandwidth GPU memory to lower-cost tiers like LPDRAM which have increased capacity to support longer contexts and larger batches. KV cache reuse minimizes redundant computations by reapplying previously stored attention data, reducing re-computation, and improving performance.

KV cache can be conceptually categorized as hot, warm, and cold based on its recency and likelihood of reuse.

Hot KV cache: This is the active working set generated during the current inference session. It is stored in HBM to provide maximum bandwidth and lowest latency for ongoing token generation.

Warm KV cache: This refers to the recently generated KV cache - typically few minutes to few hours earlier. It can be offloaded to LPDDR5x memory which offers significantly lower power and sufficient bandwidth/latency to enable efficient swapping back to HBM when the model requires long context recall.

Cold KV cache: Cache that is more than several hours to days old and is less likely to reused can be offloaded to non-volatile storage. This tier prioritizes capacity over latency and is typically used for long-term history.

The demand for increased LPDRAM is driven by the growth in context length and batch/concurrency which unlocks new use cases for generative AI, including personalized AI assistants, analysis of large data sets and many multimodal applications. As context and batch size grows, the KV cache footprint increases so provisioning a low-power, high-capacity LPDRAM tier allows frameworks to store more KV cache, minimize evictions, and reduce re-computation. LPDRAM can offer much larger capacity to sustain KV-cache reuse for long contexts and many concurrent sessions—reflecting workload demands.

Experiment Setup

Experiments were conducted to quantify the effect of LPDRAM memory capacity on LLM inference. Scenarios included inference with extended context lengths and large batch sizes using the Llama3 70 Billion model. These represent current LLM deployment challenges, where memory saturation restricts scalability and responsiveness. Results show that increasing memory capacity via LPDRAM offload enables efficient scaling, higher throughput, larger context retention and lower latency.

The NVIDIA GH200 NVL2 configuration used for these experiments demonstrates a unified system memory architecture for LLM inference. The NVIDIA Grace™ CPU with LPDDR5x and the NVIDIA Hopper GPU with HBM3e are connected by NVIDIA NVLink™-C2C, providing cache-coherent access between HBM and LPDDR5X. Further, in the system, there are two of these Grace-Hopper Superchips that are connected by the NVLink-C2C that provides cache-coherent access between the two superchips. The unified memory model allows frameworks to treat the LPDRAM as an extension of GPU memory, simplifying software and improving scalability.

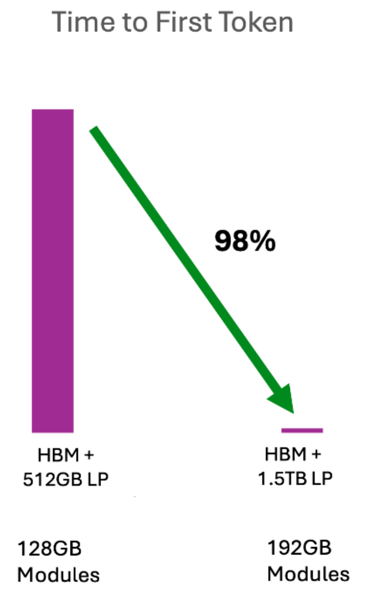

Figure 1: Real-time Inference with Llama3 70B (FP16) using 500K context length and 16 concurrent users

We use the system in two configurations: one with a single Grace-Hopper superchip and 512GB of LPDDR5X memory, and another with two superchips, each with 512GB LPDDR5X, emulating 1TB total. These setups enable measurement of LLM KV cache offloading for various LPDRAM capacities, supporting projections up to 1.5TB per CPU (enabled using eight 192GB LPDDR5X SOCAMM2 modules). This approach demonstrates the advantages of higher LPDRAM capacity, especially for long-context LLM inference and large KV caches.

Our performance analysis used optimized inference frameworks, including NVIDIA TensorRT™-LLM and vLLM, which support advanced memory management and batching for efficient LLM inference.

Results

Our experiments focused on two primary scenarios: large batch offline inference and real-time inference with extended context lengths. We used the Llama3-70B model with FP16 quantization.

Real-Time Inference

Figure 1 illustrates that moving the KV cache to larger LPDRAM dramatically improves time to first token (TTFT) latency. Expanding LPDRAM capacity from 512GB per CPU to 1.5TB reduces TTFT by up to 98%, which benefits latency-sensitive tasks like answering long documents, legal analysis, and interactive retrieval-augmented generation (RAG) in enterprise knowledge bases. Larger memory helps avoid re-computation as context length increases, speeding up initial responses and improving user experience.

LPDRAM capacity is essential for scaling large language model inference, ensuring high throughput and low latency for varied enterprise uses. Significant TTFT gains enable a single system to support up to 16 users, but as demand grows—requiring lower latencies and higher concurrency—higher LPDRAM capacities and processing power will be needed to accommodate larger KV caches and user loads.

Offline Inference

For offline batch inference operations such as transcribing conference calls, Figure 2 illustrates that increased LPDRAM capacity enables the system to handle a greater number of concurrent requests. Specifically, a 1.5TB LPDDR5X memory configuration doubled the batch size—defined as concurrent request capacity—in comparison with 512GB LPDDR5X. Lower memory capacities slow down request processing and necessitate more key-value recomputation. The optimal batch size in each case was established based on peak throughput.

Additionally, throughout these experiments, both memory types (LPDRAM and HBM) were utilized to their full potential, and GPU utilization reached its maximum, indicating efficient resource allocation. Therefore, further increasing LPDRAM capacity can deliver additional performance improvements, especially when combined with higher-speed accelerators.

These memory optimization strategies are particularly advantageous for enterprise-scale applications such as document processing, batch summarization, embedding creation, and synthetic data generation, where maximizing throughput remains critical.

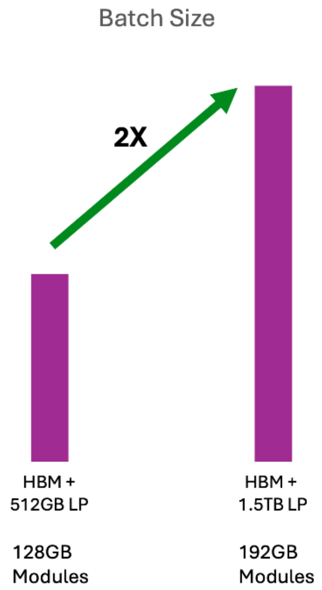

Figure 2: Offline inference with Llama3 70B (FP16) with Batch Size Scaling

Conclusion

LLMs need substantial memory that provides high bandwidth, low latency, and large capacity. Unified memory systems combining LPDRAM and HBM help meet these requirements, allowing LLMs to handle growing context sizes. LPDRAM supports this high-capacity demand efficiently, using less energy than traditional DRAM, which helps reduce the total cost of ownership (TCO) for data centers deploying LLMs.

By utilizing additional on-node LPDRAM capacity, LLMs can expand their KV cache and accommodate more concurrent users with lower latency, further optimizing TCO for AI in data centers. Our findings indicate that a 1.5TB LPDDR5X capacity per CPU can cut TTFT for real-time inference by 98% and double the batch size during batched inference compared to 512GB LPDDR5X, making it an appealing option.

Software improvements in models and algorithms are continually enhancing KV cache efficiency, ensuring optimal use. Meanwhile, developments in complex multi-modal AI, physical AI, and advancements in content memory storage underscore the growing challenge of managing larger KV caches. As these technologies and trends advance, careful management of the memory hierarchy becomes increasingly important, with LPDRAM playing a key role in maintaining consistently high-quality user experiences.

References

Micron white paper: The role of low-power (LP) memory in data center workloads (2025)

TensorRT: https://github.com/NVIDIA/TensorRT-LLM

vLLM: https://github.com/vllm-project/vllm

Acknowledgements

The authors acknowledge contributions from Kiran Palli (Product Management Director, AI Solutions Group), Steve Janzen (Sr. Manager, Strategic Marketing), and Sri Manasa Santoshi Penmetsa (Staff Product Manager, LPDRAM) and various groups in the Micron Cloud Memory Business Unit supporting HBM and LPDRAM products.

Author Bio

Khayam Anjam is a Staff Engineer at Micron Technology. As part of Micron's Data Center Workload Engineering team, he works on LPDDR and GDDR memory—characterizing and optimizing modern AI workloads across GPU-accelerated and data center platforms. He also contributes to technical papers, internal reports, and customer-facing collateral. Prior to Micron, Khayam designed AI systems for computer vision and anomaly detection. At Dell Technologies, he authored six patents while contributing to ML, analytics, and reliability initiatives. He brings 10+ years of industry experience spanning performance engineering, generative AI, and distributed systems. Khayam holds an M.S. in Computer Engineering from Clemson University.

Sujit Somandepalli is a Senior Engineering Manager in the Data Center Workload Engineering group at Micron Technology, Inc., where he leads performance characterization and workload-driven optimization of next-generation memory and storage solutions. His work focuses on bridging application behavior with system-level architecture to drive differentiated value in data center deployments. Sujit brings prior experience from Dell Inc. and Qualcomm Inc., with a strong background in systems design, performance modeling, and workload analysis and is particularly interested in emerging memory technologies, memory hierarchy tuning, and application-aware system design.

Dr. Sudharshan S. Vazhkudai is a Fellow of systems design engineering at Micron Technology. Here he established the Data Center & Client Workload Engineering team, which brings an end-to-end systems perspective in understanding how deep-memory hierarchy is used to create modern system architectures optimized for workloads. Prior to this, for over two decades, he worked at the US Department of Energy's Oak Ridge National Lab as a Distinguished Scientist and a Program Director, building HW/SW codesign solutions for generations of world's fastest production supercomputers, serving a large user base in AI and HPC. Dr. Vazhkudai holds a Ph.D. in computer science from the University of Mississippi and has also served as a joint faculty at the University of Tennessee. He has published over a hundred peer-reviewed papers on the fundamental underpinnings of data center systems design, system architecture, deep-memory/storage hierarchy, heterogeneous CPU/GPU architectures, data management, and distributed systems.

Praveen Vaidyanathan is Vice President and General Manager of Cloud Memory Products in the Cloud Memory Business Unit at Micron Technology. In this role, Praveen leads Micron's cloud memory strategy and execution, working closely with global hyperscale customers. He oversees product line management, business development, and customer qualification teams to deliver next-generation memory solutions optimized for cloud, AI, and data center workloads.

micron.com