MEXT

WHITEPAPER | Q3'25

Eliminating RenderMan Out-of-Memory Challenges with MEXT Predictive Memory™

An Evaluation of MEXT AI-Powered Predictive Memory™ Technology on Pixar's Open-Source Animation Rendering Software, RenderMan

---

Executive Summary

This whitepaper showcases how MEXT can transform the economics of running RenderMan, the open-source animation rendering application developed by Pixar. The user's original configuration was a workstation with 64GB of memory (all DRAM). The RenderMan workload being run, however, required a much larger amount of memory, resulting in repeated Out-of-Memory (OOM) error-related crashes.

MEXT software enables system flash to appear as DRAM-speed memory to the OS; this expands the apparent memory capacity of the system WITHOUT the user having to purchase additional physical DRAM for their existing system or having to upgrade to a new, higher-memory system. Upon installing MEXT software, which was a < 5-minute process, the user can now successfully run RenderMan, avoiding OOM crashes and instead achieving performance levels similar to what a 128GB all-DRAM system would have provided.

Introduction

Modern animation workloads running on RenderMan continue to stretch the limits of compute and memory resources, driven by ultra-detailed assets, advanced shading networks, and physically based rendering techniques. These demands place immense pressure on system memory, making it harder for studios to scale efficiently using traditional DRAM alone. As memory requirements grow, the cost, supply limitations, and lack of scalability inherent to DRAM create critical bottlenecks that hinder pipeline throughput and creative iteration. Eliminating these challenges calls for a more intelligent memory strategy—one that introduces elasticity, maximizes system utilization, and delivers leadership RenderMan performance without the overhead of memory overprovisioning.

Hardware Configuration

MEXT-Enabled System

- Lenovo ThinkStation P620 Workstation
- AMD Ryzen™ Threadripper™ Pro 3975WX 32-core, 64-Thread 3.5 GHz Processor
- Kioxia PJ1-KW1920 1.92T NVMe SSD
- 128GB Total Memory
    - 64GB DRAM
    - 64GB of MEXT Memory™

Original System

- Lenovo ThinkStation P620 Workstation
- AMD Ryzen™ Threadripper™ Pro 3975WX 32-core, 64-Thread 3.5 GHz Processor
- Kioxia PJ1-KW1920 1.92T NVMe SSD
- 64GB Total Memory
    - 64GB DRAM
    - No MEXT Memory™

Software Stack

- Pixar RenderMan v 26.3
- OpenUSD 1.39.3
- Rocky Linux 9.6
- MEXT Predictive Memory™ software
- MEXT View™ observability software leveraging Open Telemetry (OTel) and Grafana

---

MEXT AI-Powered Predictive Memory™

How It Works

System memory, or DRAM, is one of the costliest components involved in modern computing. However, across many business environments, its utilization regularly drops to 50% or below (demonstrated by various studies from leading cloud service providers and hyperscalers).

MEXT's software solution solves this utilization issue by 1) continuously monitoring which memory pages in DRAM actively being utilized, or "hot", and which have gone "cold", 2) offloading the cold memory from DRAM to flash, and 3) leveraging AI to mitigate the effects of flash latency and keep the system performant (via the MEXT Predictive Memory™ Engine).

The MEXT AI-Powered Predictive Memory™ Engine continually predicts which offloaded pages might be requested by the application soon (in other words, which pages are likely to soon go from cold to hot), and transparently moves them back into DRAM before the requests are even made. As a result, the application stays performant because from its perspective, the relevant memory pages are always already resident in DRAM.

In this way, MEXT gets system flash to appear as memory to the OS, extending the effective memory capacity of the system.

Value

Some customer applications tend to run out of memory and suffer performance issues or crashes as a result. In the past, they would have had to either add more DRAM to their existing system or purchase an entirely new system with more DRAM. With MEXT, these complex and expensive paths can be avoided; instead, existing system flash can be leveraged as a form of memory, extending the memory capacity of the system and enabling the application to keep running performantly instead of suffering crashes. This keeps infrastructure costs stable while solving application performance challenges.

Seamless Implementation

MEXT is a patent-pending, software-only solution that works with any configuration: on-premises or cloud, with any processor, in virtualized / bare-metal / containerized environments, with no changes to the OS or applications. Installation takes less than 5 minutes.

MEXT Solution Components

The MEXT Predictive Memory™ solution consists of 3 primary components: the MEXT Driver, the MEXT Predictive Memory™ Engine, and the MEXT View™ Observability Platform.

MEXT Driver

The MEXT Driver is a dynamically loadable kernel module (which does not alter the standard Linux kernel) that sends process and memory page telemetry data to the MEXT Predictive Memory™ Engine.

MEXT Predictive Memory™ Engine

The MEXT Predictive Memory™ Engine is a user-space process that feeds predictions of which memory pages should be pushed from flash to DRAM—making predictions / inferences in under a fraction of a second. It runs entirely on the local Linux operating system (on a single CPU core) and does not require a GPU.

It was inspired by modern AI techniques based on neural networks. Instead of using these techniques to predict words or natural language patterns (like ChatGPT does), it predicts sequences of future memory page accesses. It consists of a family of models that work together, combining extremely lightweight heuristic predictors with more powerful neural-network models. For any given workload, it automatically adjusts to use the model or group of models that performs best. Continuous observation of which predicted pages were actually used by the application also enables the engine to acquire real-time feedback regarding model accuracy, supporting ongoing adaptation and self-optimization.

MEXT View™ Observability Platform

MEXT also provides a user-space application called MEXT View™ that provides observability / visualization tools to help customers profile their workloads—illustrating how much memory their applications are using at any given time and what portion of this memory is hot / warm / cold. All cold memory pages are good candidates for optimization by MEXT Predictive Memory™ software. MEXT View™ also provides insight into the ongoing prediction accuracy of the MEXT Predictive Memory™ Engine.

Methodology

RenderMan with the OpenUSD Hydra Delegate was used to render the Moana Island scene definition in usda into a high resolution OpenEXR image format using the usdrender command line tool. Complete time to render the scene and create the OpenEXR image was recorded for 2 scenarios: the MEXT-enabled system (64GB DRAM, 64GB MEXT Memory™) and the original system (64GB all-DRAM).

Results

OOM Crashes Avoided

In the original system, repeated OOM crashes were observed because the RenderMan workload required more memory than the 64GB DRAM maximum provided by the system. Upon installation of MEXT, however, the effective memory capacity of the system expanded to 128GB—enabling RenderMan to keep running performantly.

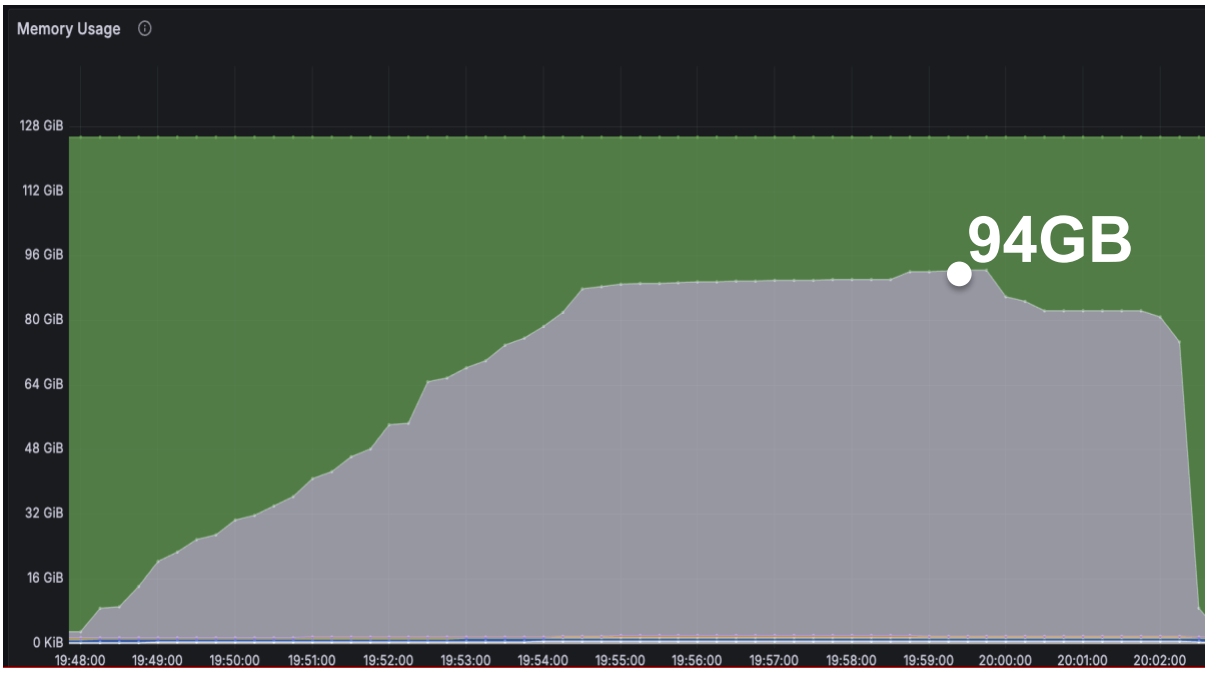
Circumvented Costs

Without MEXT, the user would have had to either add additional DRAM into the existing system, or purchase a new system altogether. This translates to an additional $10-20K per user. Now, multiply this

cost across each user at a medium-to-large animation studio: the costs could have racked up significantly.

Maximized DRAM Utilization

As seen in the graph below, the maximum amount of system memory used was 94GB out of 128GB of total system memory; in other words, the workload used up 73% of the total system memory which consists of 64GB of DRAM and 64GB of MEXT Memory™. Without MEXT, it is clear that crashes would abound because of the workload's true need for the full 94GB of memory.



Conclusion

For users running large RenderMan workloads that are suffering latency challenges or OOM crashes, leveraging MEXT is a cost-effective way to prevent such challenges and continue to run performantly instead (because MEXT helps avoid the high price tags associated with adding additional physical DRAM). MEXT software also integrates seamlessly into existing infrastructure, with a < 5-minute installation process, and scales effortlessly across studio workstations. In this way, MEXT is enabling a step-change in the economics of creative production—delivering a high-performance, cost-effective path for studios running RenderMan at scale.