METRUM AI

Know Your AI. Grow Your Business.

May 2025

HIGH-PERFORMANCE TCO-OPTIMIZED RAG WITH SSD OFFLOADING

Performance Analysis and Real-World Application of Solidigm SSDs

Table of Contents

Introduction 3

Key Highlights 3

Performance Results 4

- Queries Per Second 5
- Recall 6
- Indexing Time 7
- DRAM Usage 8
- Gen-on-Gen Performance Comparison 9

Testing Solidigm SSDs in a RAG Solution 10

How the Solution Works 11

Solution Architecture 13

Summary 14

Addendum

Appendix A 15

© 2025 Metrum AI Inc.

Introduction

Memory limitations pose a significant challenge in deploying AI workloads, especially in Retrieval-Augmented Generation (RAG) applications. Large-scale vector databases, AI models, and indexing structures consume substantial memory resources, making it difficult to load massive datasets, run cutting-edge large language models (LLMs), or perform efficient real-time inference. This forces organizations into an uncomfortable choice: either invest in high-memory GPUs or compromise on model performance and dataset size, both limiting their AI capabilities.

This is where offloading data storage to SSDs comes in. By leveraging high-speed NVMe storage instead of relying solely on DRAM, Solidigm SSDs offer a scalable, cost-effective alternative that allows enterprises to deploy larger models, expand dataset capacities, and optimize memory usage without imposing excessive infrastructure costs.

Metrum AI recently gained access to high-speed Solidigm SSDs and eagerly tested their effectiveness with the goal of improving performance on our own industry-specific AI agents. To accomplish this, we first benchmarked key retrieval metrics using VectorDBBench to assess Solidigm SSD performance in vector search workloads. Then, to evaluate SSD performance in a real-world AI application, we built an end-to-end RAG solution that processes videos and generates detailed summaries of their contents. Here, we highlight the key performance insights we uncovered through these evaluations:

Key Highlights

Up to
70%
Higher Queries Per Second

Using D7-PS1010, comparing DiskANN to HNSW on 10M Dataset

Up to
57%
Lower DRAM Usage

Using D7-PS1010, comparing DiskANN to HNSW on 100M Dataset

Maintained
>98%
Recall Accuracy

While reducing DRAM usage, compared DiskANN to HNSW on 100M dataset

Up to
29%
Gen-on-Gen Queries Per Second Performance Gains

Comparing D7-PS1010 to D7-P5520

© 2025 Metrum AI Inc.

Performance Results

To measure the benefits of offloading data storage to SSDs, we conducted a series of performance tests with **Solidigm SSDs*,** using **VectorDBBench**—a benchmark tool designed to compare both performance and cost-effectiveness across popular vector databases. To compare performance with and without SSD offloading, we compared two different indexing methods:

- **DiskANN[1](utilizes SSD offloading):** An approximate nearest neighbor search (ANNS) algorithm that achieves high accuracy and a low DRAM footprint by leveraging SSD storage, making it significantly more cost-effective than DRAM-intensive alternatives.
- **HNSW (in-memory, doesn't utilize SSD offloading):** Hierarchical Navigable Small World (HNSW) graphs are a leading in-memory index for vector similarity search, delivering ultra-fast search speeds and high recall. By keeping the entire index in DRAM, HNSW offers low-latency performance at the cost of higher memory usage, relying on expensive DRAM to maintain real-time responsiveness.

Key Observations

- **Significant Increase in Query Speed:** SSD offloading with DiskANN improves query throughput, achieving 10268 QPS on the Cohere 1M dataset (compared to 8405 QPS with HNSW) and maintaining an **up to 70% higher QPS advantage on the 10M dataset.**
- **Drastic Reduction in DRAM Usage: DiskANN reduces DRAM consumption by up to 57% compared to HNSW in large-scale datasets,** with DRAM usage dropping from 336GB to 145GB on the LAION 100M dataset, demonstrating the efficiency of SSD-based retrieval.
- **Scalability Without Compromising Accuracy:** Despite reducing DRAM dependency, DiskANN maintains a **high recall accuracy of 99.21% on the LAION 100M dataset.** This translates to higher quality retrievals in RAG workloads while ensuring scalability and cost efficiency.
- **Impressive Generational Performance Gains:** The D7-PS1010 delivers **29% better performance than the D7-P5520** on the Cohere 1M dataset, highlighting substantial improvements in Solidigm's newer SSD architectures.

* Refer to **Appendix A** for hardware and SSD specifications.

© 2025 Metrum AI Inc.

Queries Per Second (QPS)

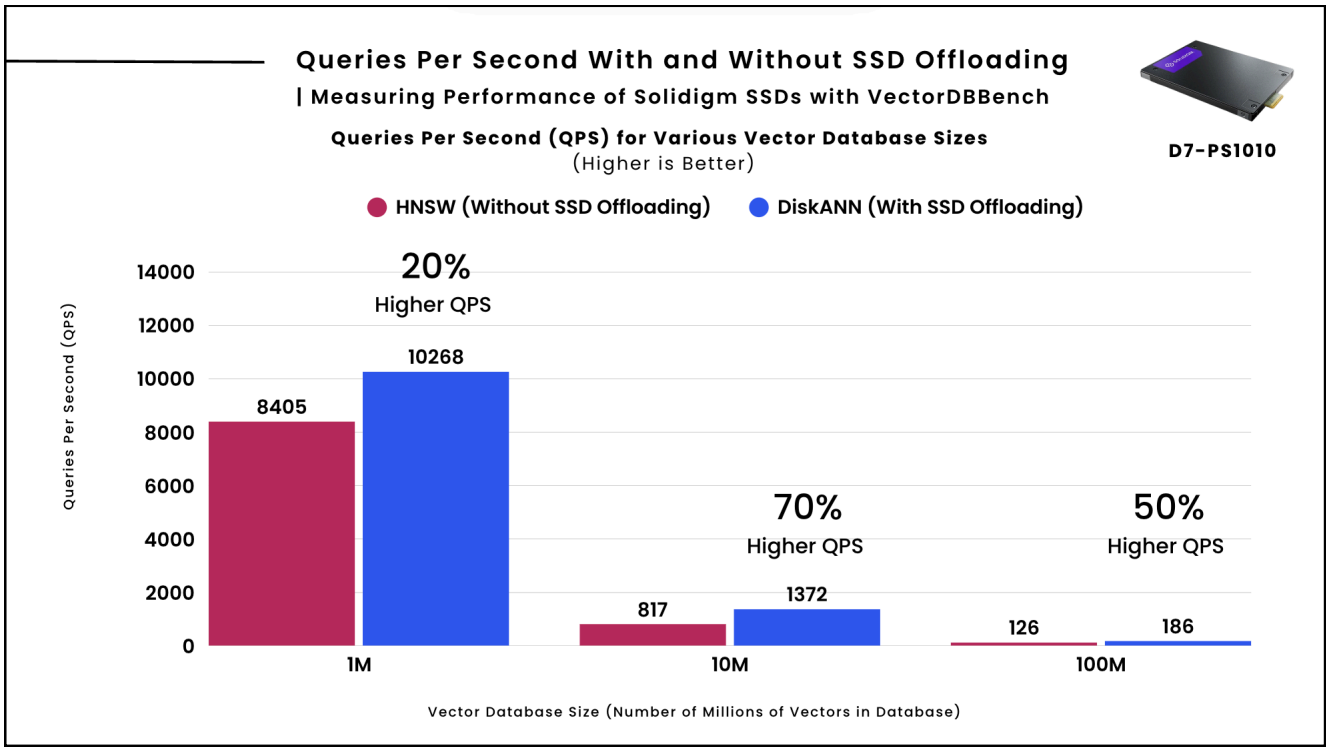

Figure 1: Queries per second performance for DiskANN and HNSW indexing methods over various vector database sizes.

The following test results are based on the default settings from the benchmark. These results are eye-opening: **despite the common assumption that SSDs are inherently slower than DRAM, DiskANN consistently outperforms the in-memory HNSW approach across both DRAM usage and query throughput metrics.** While this is counterintuitive as SSDs are usually slower, it highlights the fact that disk-friendly graph traversal algorithms such as Vamana (the core of DiskANN's highly optimized architecture) with plenty of pre-processing, can dramatically speeding up similarity searches by efficiently packing vectors into the SSD. This allows for sequential and batched I/O operations instead of random reads. Ultimately, the DiskANN architecture not only reduces the reliance on DRAM but also maximizes the high bandwidth and low latency capabilities of modern NVMe drives, such as Solidigm SSDs.

This specific metric measures how many search queries the system can process per second, indicating throughput capacity. DiskANN using Solidigm SSDs consistently outperforms HNSW across all dataset sizes. At 10M and 100M vector scales, **DiskANN achieves up to 1.7x and 1.5x higher query throughput**, respectively, compared to HNSW. This performance boost makes DiskANN a strong fit for large-scale AI workloads that demand high QPS to support real-time user experiences and large concurrent workloads. In contrast, HNSW's in-memory design relies on DRAM to deliver fast performance but struggles to maintain its throughput as data scales.

© 2025 Metrum AI Inc.

DRAM Usage

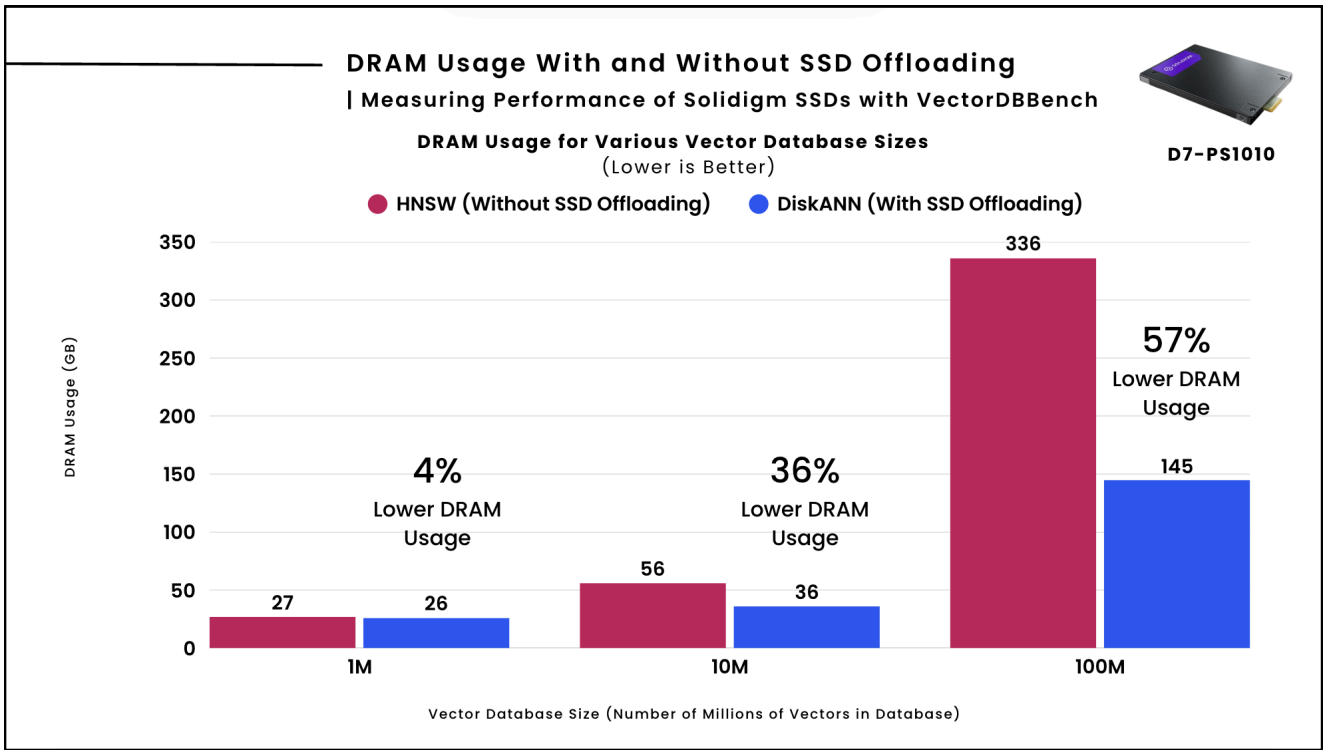

Figure 2: Indexing time performance for DiskANN and HNSW indexing methods over various vector database sizes.

This metric quantifies the system memory (DRAM) required during vector search operations. HNSW, as an in-memory algorithm, loads the entire index into DRAM, resulting in high memory usage of 336 GB at 100M vectors. DiskANN, by offloading the index to SSD, drastically **reduces memory requirements by up to 57% to just 145 GB** at the same scale, less than half the DRAM footprint. This efficiency allows organizations to scale vector databases more cost-effectively, reducing reliance on expensive memory while maximizing existing infrastructure.

In addition to reducing DRAM requirements, **DiskANN with Solidigm SSDs** also delivers superior cost-efficiency when **measuring queries per second per dollar (QPS/$)*** of memory. At 100M vectors, HNSW achieves 126 QPS using $3,560 worth of combined DRAM and SSD, yielding 0.035 QPS per dollar. In contrast, DiskANN delivers 186 QPS with a combined DRAM and SSD cost of $2,480, resulting in 0.075 QPS per dollar, **more than 2x higher efficiency**. This highlights how SSD offloading not only cuts down memory usage, but also significantly improves the return on infrastructure investment, making it a smarter choice for scalable vector search deployments.

* QPS/$ was measured by dividing QPS results on **Page 5** by the total cost of DRAM and SSDs used in the server configurations for testing DiskANN and HNSW. Prices reflect publicly sourced data available at the time this document was published.

© 2025 Metrum AI Inc.

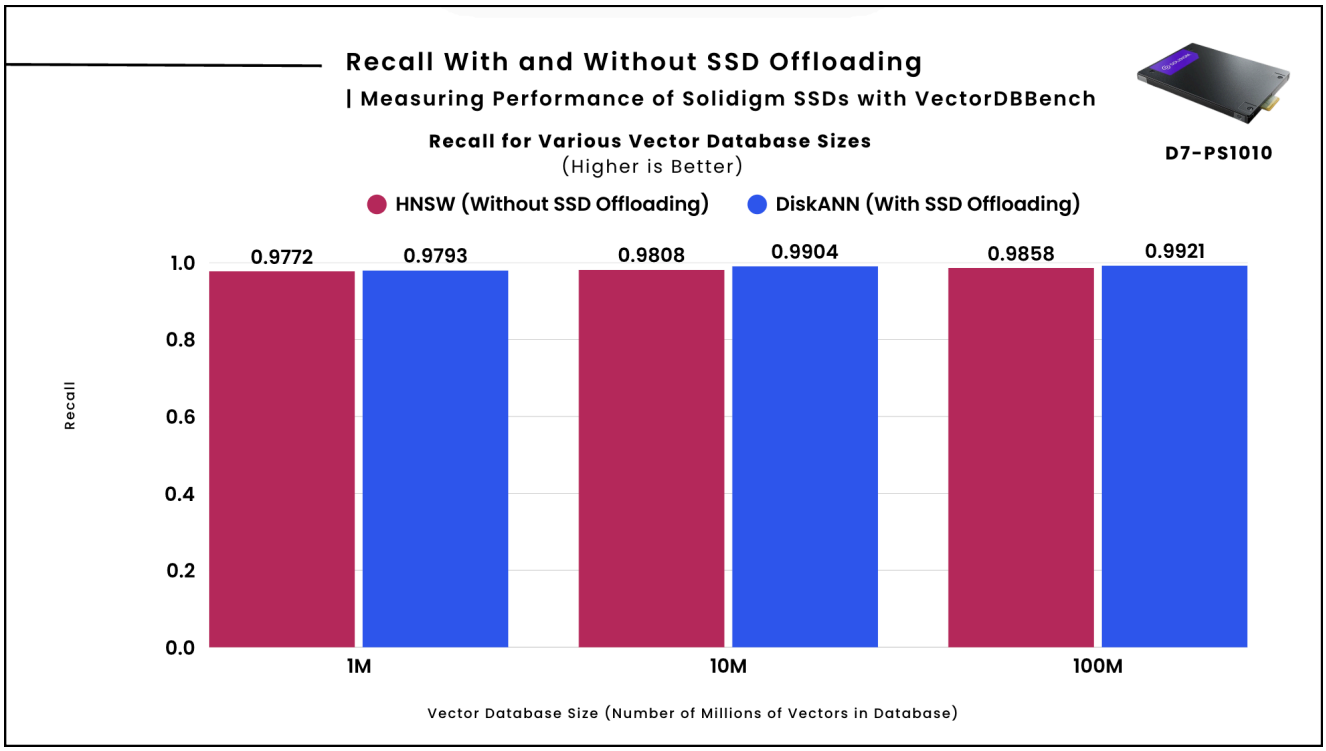

Figure 3: Recall performance for DiskANN and HNSW indexing methods over various vector database sizes.

This metric measures how accurately the system retrieves relevant vectors from the database. DiskANN achieves slightly higher recall values than HNSW across all dataset sizes. At 100M vectors, **DiskANN with Solidigm SSDs achieves 99.21% recall accuracy versus HNSW's 98.58% recall accuracy, maintaining high search quality despite relying on SSDs.** This demonstrates that SSD-offloading doesn't come at the expense of accuracy, making it suitable for production-grade AI applications where precision matters.

© 2025 Metrum AI Inc.

Indexing Time

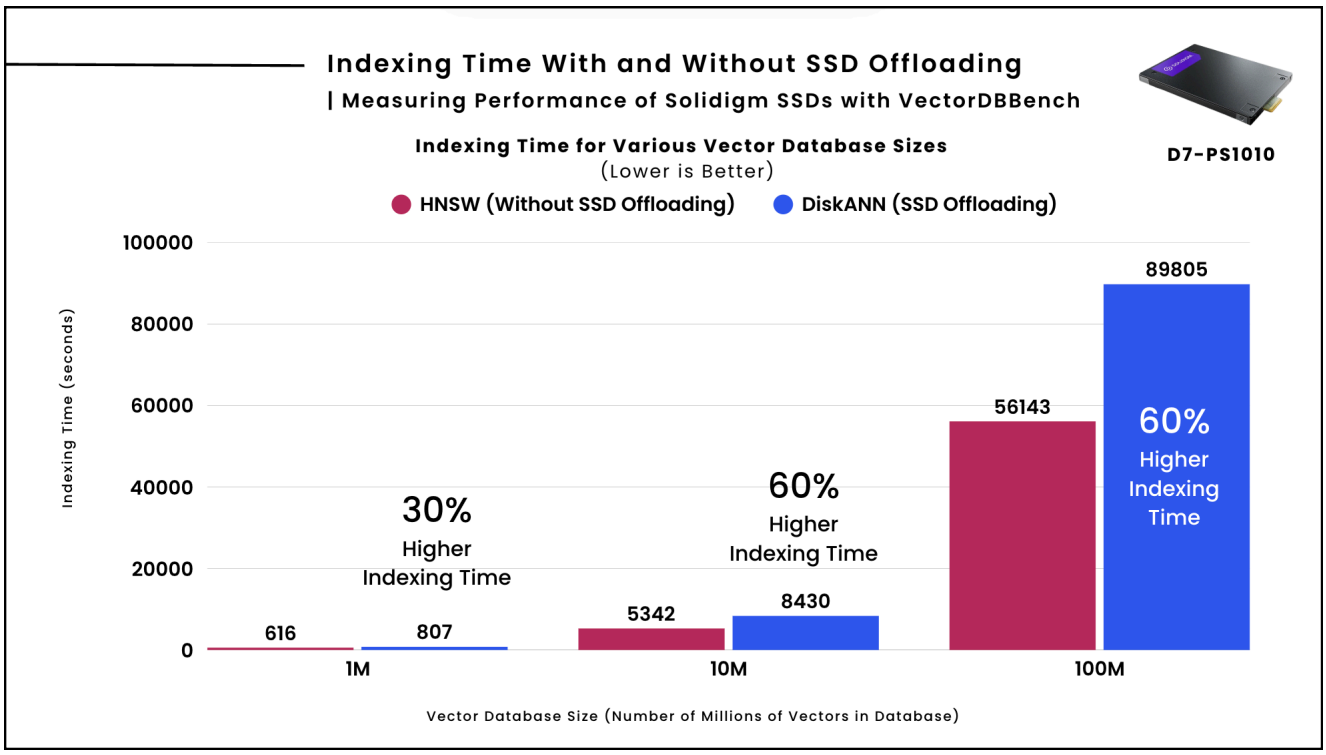

Figure 4: Indexing time performance for DiskANN and HNSW indexing methods over various vector database sizes.

This metric captures the time required to build the search index. **HNSW indexes data faster, especially at larger scales: 56,143 seconds vs. DiskANN's 89,805 seconds at 100M vectors.** While indexing time is higher, the benefit of faster retrieval and lower DRAM footprint makes **DiskANN using Solidigm SSDs a more viable solution for large-scale AI-driven applications where frequent indexing updates are not a bottleneck.** For most enterprise applications, this one-time cost is outweighed by the ongoing performance benefits during operation.

© 2025 Metrum AI Inc.

Gen-on-Gen Performance Comparison

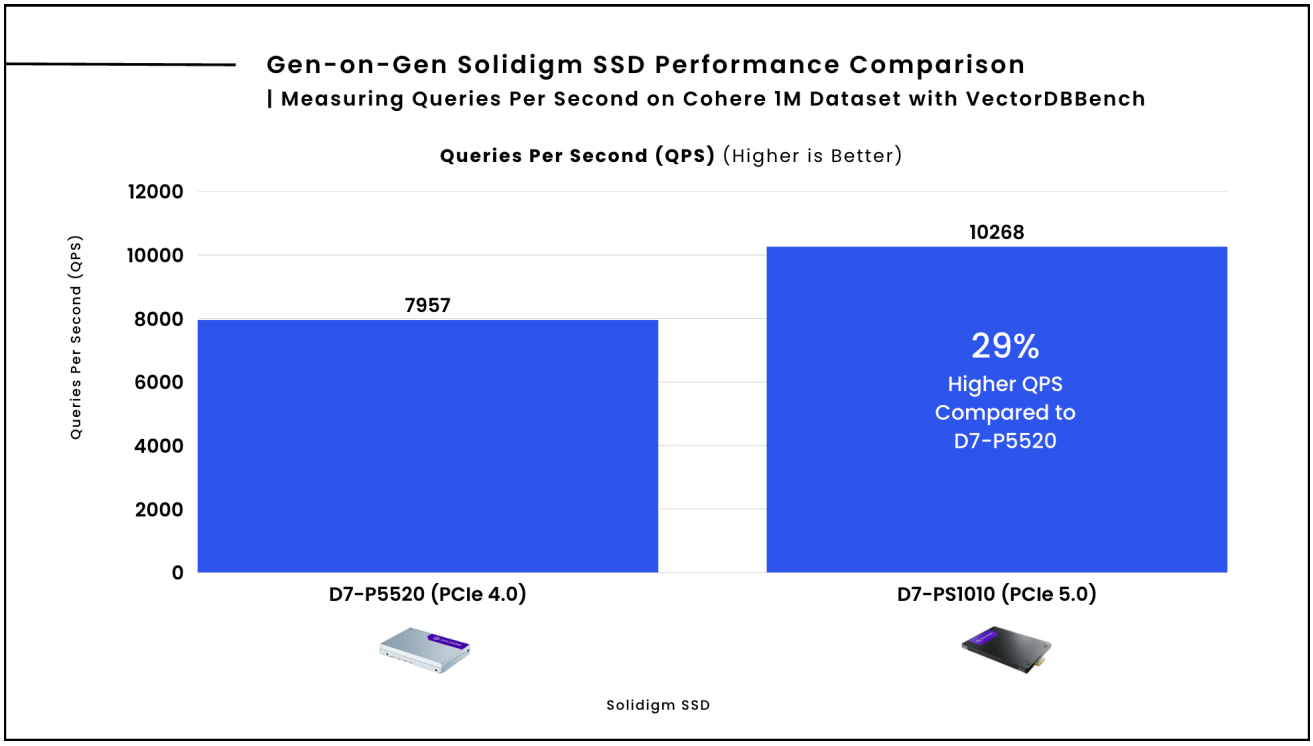

Figure 5: Queries per second performance using DiskANN on the Cohere 1M dataset with VectorDBBench, comparing generations of Solidgm SSDs.

Here, we conducted performance testing using VectorDBBench on the Cohere 1M dataset, highlighting significant gains in query throughput within Solidigm's D7 SSD lineup. As illustrated in the benchmark results, the D7-P5520 (TLC) achieves 7,957 queries per second (QPS), serving as a strong baseline within the PCIe 4.0 generation. Stepping up to the D7-PS1010 (TLC), performance increases sharply to 10,268 QPS, **marking a 29% improvement over the D7-P5520.**

These results reflect the architectural and interface advancements embedded in the D7-PS1010. With support for PCIe 5.0 and enhanced firmware-level optimization for random read workloads, the D7-PS1010 is engineered to meet the demands of AI inference and vector search at scale. Its performance boost is particularly relevant in high-throughput environments where real-time responsiveness and concurrent vector search requests are critical.

© 2025 Metrum AI Inc.

Testing Solidigm SSDs in a RAG Solution

After benchmarking with VectorDBBench for vector search performance, we evaluated the real-world effectiveness of Solidigm SSDs by building an end-to-end RAG solution for analyzing videos and generating comprehensive summaries. This system enables enterprises to process large-scale video data by storing and retrieving extensive metadata and embeddings directly from Solidigm SSDs. By leveraging high-speed NVMe storage, organizations can scale their retrieval databases well beyond DRAM limitations while still achieving real-time query speeds.

Our solution stands out through its **production-ready implementation** using DiskANN with **established ecosystem support**, enabling seamless integration with open-source RAG architectures and state-of-the-art models like Llama 3.3 70B. This approach allows organizations to deploy advanced AI capabilities without specialized technologies while leveraging existing hardware for larger models that would otherwise exceed available memory, delivering practical AI acceleration that solves real business challenges today.

In this solution, we leverage **Solidigm D7-PS1010 PCIe 5.0 NVMe SSDs** to efficiently handle data-intensive AI operations, supporting efficient video analysis and insight generation.

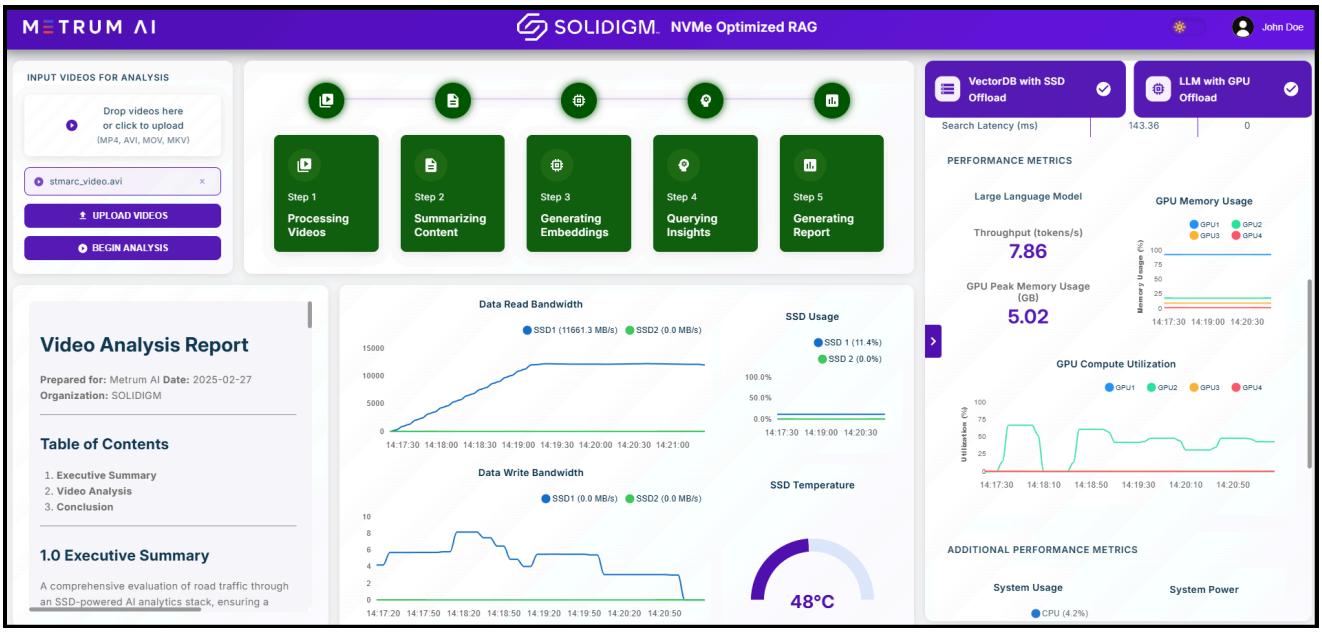

Figure 6. User Interface of Metrum AI's RAG solution leveraging Solidigm SSDs.

© 2025 Metrum AI Inc.

How the Solution Works

The solution operates through five interconnected steps:

1. **Processing Videos:** Video files are transferred to the Solidigm NVMe SSD and processed into individual frames at a configurable frame rate. This step involves concurrent read/write operations, with performance directly related to the input video characteristics.

2. **Summarizing Content:** The extracted video frames are processed using the Phi 3 Vision language model (deployed on NVIDIA GPUs) to generate descriptive text summaries. The solution reads frames from the NVMe SSD and performs optimized batch inference.

3. **Generating Embeddings:** Text summaries are converted to embeddings using the gte-Qwen2-1.5B model. These vector embeddings are stored in Milvus VectorDB, with indexes optionally offloaded to the NVMe SSD. This stage demonstrates substantial write performance, reaching up to 4,520 MB/s.

4. **Querying Insights:** The solution performs parallel queries against the vector database to extract meaningful insights from the video frame embeddings. When configured with SSD offload, this step involves significant read operations from the vector indexes stored on the NVMe SSD.

5. **Generating Reports:** The system transforms queried insights into structured reports containing both image frames and corresponding analysis. A Llama 3.3 70B language model, with weights offloaded to the NVMe SSD via DeepSpeed, generates safety recommendations and summaries. This stage showcases exceptional read performance, with observed bandwidth reaching 12,343 MB/s.

© 2025 Metrum AI Inc.

GPU Memory Optimization: A groundbreaking innovation in our solution is the offloaded inference mechanism for serving the Llama 3.3 70B model. While this model typically requires approximately 160GB of GPU memory (equivalent to four L40 GPUs with 48GB each), our implementation uses Ray Serve with DeepSpeed Inference and dynamic batching to reduce peak GPU memory usage to just 7-8GB. This represents a 20x reduction in memory requirements while maintaining model performance.

The system can handle up to 1,024 concurrent generation requests within this minimal memory footprint, establishing a novel approach to large model inference with no prior implementations of this kind. This breakthrough in efficient model deployment dramatically reduces infrastructure costs and enables deployment on more modest hardware configurations. The NVMe SSD acts as an extension of GPU memory, with the system intelligently loading only required model weights during inference.

Vector Database Performance: When comparing vector database implementations, the solution demonstrates that DiskANN (with SSD offload) outperforms in-memory HNSW, handling more queries per second (80.17 vs. 75.39) with lower search latency (187.11s vs. 198.97s).

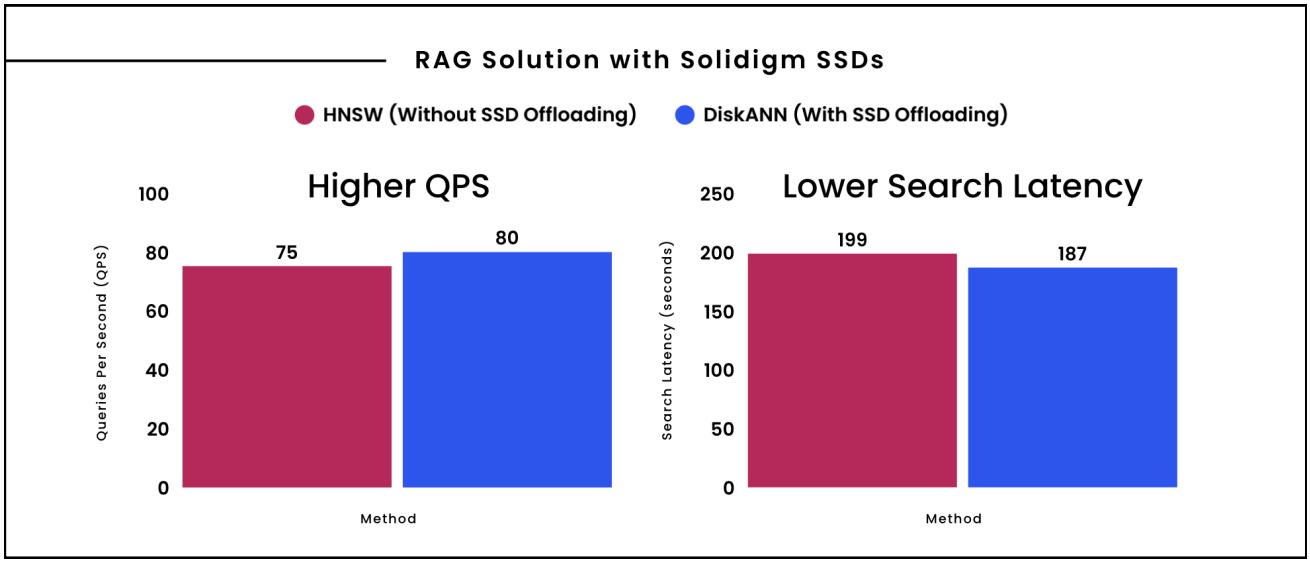

Figure 7. Looking at QPS and Search Latency Performance for Metrum AI's RAG solution leveraging Solidigm SSDs.

Additionally, the solution integrates a custom retriever built for parallel querying, specifically optimized to maximize throughput when used with a DiskANN index on SSD storage. This further accelerates RAG performance and showcases how properly configured NVMe storage can enhance even memory-intensive vector operations.

The Solidigm NVMe optimized RAG solution exemplifies how strategic integration of high-performance storage can enhance complex AI workflows, enabling efficient processing of high-dimensional data while optimizing computational resource utilization.

© 2025 Metrum AI Inc.

Solution Architecture

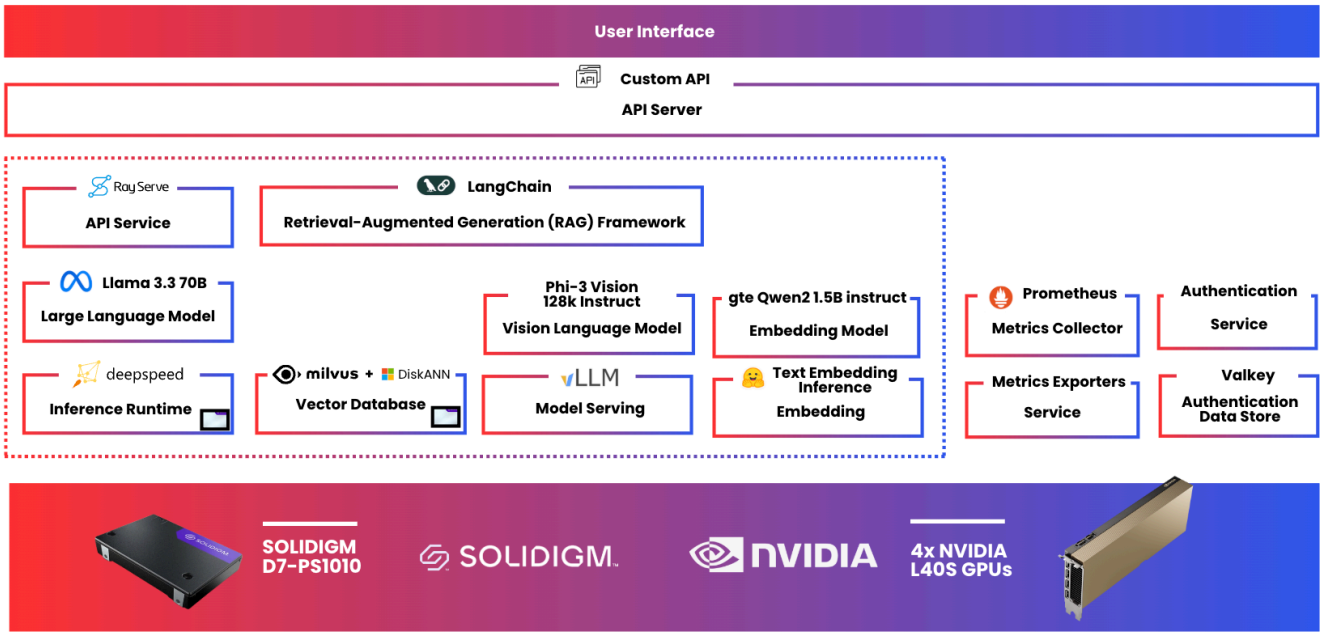

Figure 8. Solution Architecture.

Our retrieval-augmented generation (RAG) software stack incorporates the following key components to power this solution:

- **DeepSpeed Inference + Ray Serve (Llama 3.3 70B):** The solution's highlight is the deployment of the Llama 3.3 70B model, one of the most advanced open-weight LLMs, using DeepSpeed Inference with Ray Serve, enabling scalable, high-throughput, and low-latency inference for demanding enterprise workloads.
- **vLLM (Phi-3 Vision Language Model):** The Phi-3 vision-language model is served using vLLM, an industry-standard library for optimized model inference, ensuring efficient GPU utilization and responsive multi-modal generation.
- **Text Embedding Interface (GTE Qwen2 1.5B Instruct):** A top-ranked embedding model hosted locally, used to generate high-quality vector representations for document and query matching.
- **LangChain:** A widely-used open-source framework for building retrieval-augmented generation pipelines, orchestrating components like LLMs, retrievers, and tools.
- **MilvusDB:** A scalable, high-performance vector database used for fast similarity search across large embedding collections.

© 2025 Metrum AI Inc.

Summary

Solidigm SSDs represent a game-changing solution for overcoming the memory constraints that have been holding back AI workloads. By offloading vector database storage and model execution to high-speed NVMe SSDs, enterprises can dramatically expand their dataset capacities, run larger AI models, improve retrieval accuracy, and boost system efficiency—all without the costs of high-memory GPUs. This approach maintains rapid query speeds while significantly reducing dependence on expensive DRAM.

Our benchmarking and real-world testing results demonstrate that SSD-based retrieval not only enhances query efficiency but also scales AI workloads beyond traditional memory limitations. Now, enterprises can unlock the full potential of their RAG-based applications, supporting larger datasets, AI-driven analysis, and state-of-the-art language models without performance bottlenecks.

Looking ahead, several promising developments could further enhance NVMe-optimized AI systems. GPU-accelerated index construction, highlighted recently by NVIDIA and Microsoft at GTC, offers a solution to the challenge of lengthy index build times for large-scale vector datasets. Additionally, we anticipate computational storage devices with built-in AI acceleration capabilities, more sophisticated memory hierarchies that intelligently distribute workloads across tiered storage, and emerging industry standards for SSD-based AI acceleration. These developments point toward increasingly efficient AI systems capable of processing vastly larger datasets with modest hardware requirements.

To learn more about our performance benchmarking methodologies, reach out to us at **contact@metrum.ai**.

References

1. @misc{diskann-github; authors = Simhadri, Harsha Vardhan and Krishnaswamy, Ravishankar and Srinivasa, Gopal and Subramanya, Suhas Jayaram and Antonijevic, Andrija and Pryce, Dax and Kaczynski, David and Williams, Shane and Gollapudi, Siddarth and Sivashankar, Varun and Karia, Neel and Singh, Aditi and Jaiswal, Shikhar and Mahapatro, Neelam and Adams, Philip and Tower, Bryan and Patel, Yash; title = DiskANN: Graph-structured Indices for Scalable, Fast, Fresh and Filtered Approximate Nearest Neighbor Search; urls = https://github.com/Microsoft/DiskANN}; version = 0.6.1; year = 2023;

Solidigm images: solidigm.com
NVIDIA images: nvidia.com

© 2025 Metrum AI Inc.

Appendix A

Experiment Configuration Details

Configuration	Details
Benchmarking Toolkit	VectorDBBench (version 0.0.21)
Datasets	Cohere: 1M, 10M datasets LAION: 100M dataset
Vector Dimensions	768
Vector Database	Milvus (version 2.5.3)
HNSW Parameters	M: 30 efConstruction: 360 ef: 100
DiskANN Parameters	search_list: 100
NVMe Configuration	Mount type: xfs- rw, relatime, attr2, inode64, logbufs=8, logbsize=32k, noquota

Systems Under Test

Type	Details
Server	1x FarmGPU Giga Computing G293-Z23-AAM1-000
CPU	AMD EPYC 9554 64-Core Emb Processor
Memory	12 channel DDR5, 32GB at 4800MT/s, 384GB total (2GB per 1GB of VRAM), 436GB/s threaded memory bandwidth
Storage	2x 15.36TB PCIe 5.0 NVMe, D7-PS1010 (SOLIDIGM SB5PH27X153T) firmware G70YG100, D7-P5520 (INTEL SSDPF2KX153T1)
GPU	4x NVIDIA L40S GPUs

© 2025 Metrum AI Inc.

Solidigm SSDs

SSD	D7-PS1010	D7-P5520
Capacity	15.36TB	15.36TB
Form Factor	U.2 15mm	U.2 15mm
Interface	PCIe 5.0 x4, NVMe	PCIe 4.0 x4, NVMe
Weight	110g ± 10g	165g ± 5g
Lithography	176L TLC 3D NAND	144L TLC 3D NAND
Use Conditions	Server / Enterprise	Server / Enterprise
OS	Ubuntu 24.04 LTS	Ubuntu 24.04 LTS

© 2025 Metrum AI Inc.

Copyright Notice

Copyright © 2025 Metrum AI Inc. All Rights Reserved. Metrum AI, the Metrum AI logo, and other trademarks are trademarks of Metrum AI Inc.

Other trademarks may be the property of their respective owners.

Disclaimer

Performance varies by hardware and software configurations, including testing conditions, system settings, application complexity, the quantity of data, batch sizes, software versions, libraries used, and other factors. The results of performance testing provided are intended for informational purposes only and should not be considered as a guarantee of actual performance.

Metrum AI believes the information in this document is accurate as of its publication date. The information is subject to change without notice.