SNIA DEVELOPER CONFERENCE

SDC 25

By Developers FOR Developers

Hyatt Regency Santa Clara, CA
September 15-17, 2025

Data-Intensive Inference Done Better

Scaling Models and RAG in Limited Memory with SSD Offload

www.sniadeveloper.org

Speaker Intro

Ace Stryker, Director of Market Development at Solidigm

ace.stryker@solidigm.com

- Day 1 team member (~3.5 years)
- 5 years at Intel – solution architect, product & technical marketing
- BS - U. of North Florida, MBA - Cornell U.
- Based in Sacramento area
- Favorite AI movie: *Ex Machina*

 ©2025 Solidigm. All Rights Reserved.

Solidigm Intro

SOLIDIGM™

Pioneer in Solid-State Storage Solutions

Sole Focus on Enterprise, Aiming for #1

Global Organization, HQ in California

Built on 40 years of solid-state innovation

intel® + SK hynix

From the highest capacities to the highest performance. Core data center to edge. Hardware, firmware and software. And everything in between.

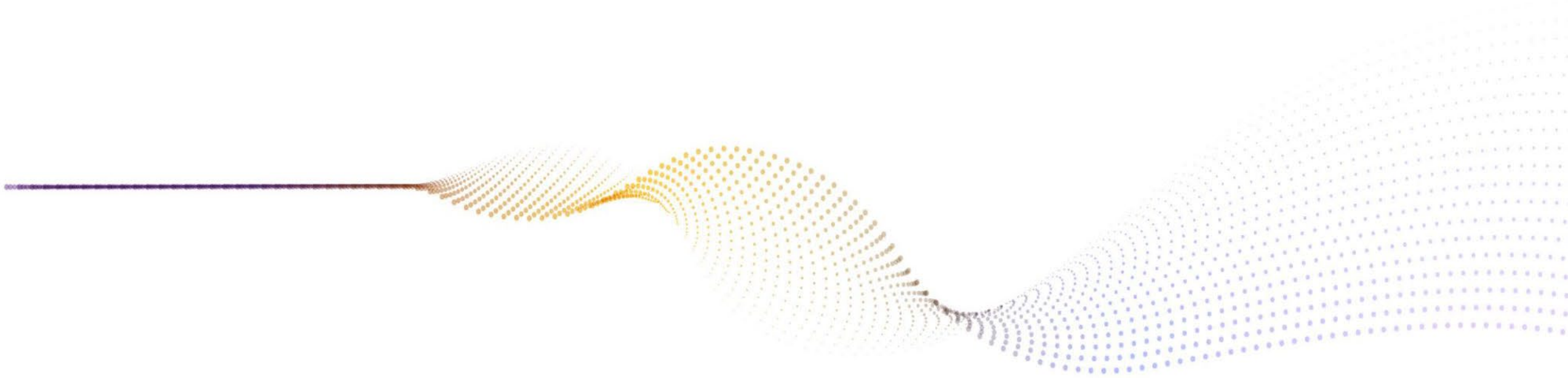
 ©2025 Solidigm. All Rights Reserved.

SDC 25

Agenda

- The Problem with Retrieval-Augmented Generation (RAG)
- Experiment and Methodology
- Example Workload: Traffic Safety Video
- Key Findings
- Next Steps

 ©2025 Solidigm. All Rights Reserved.

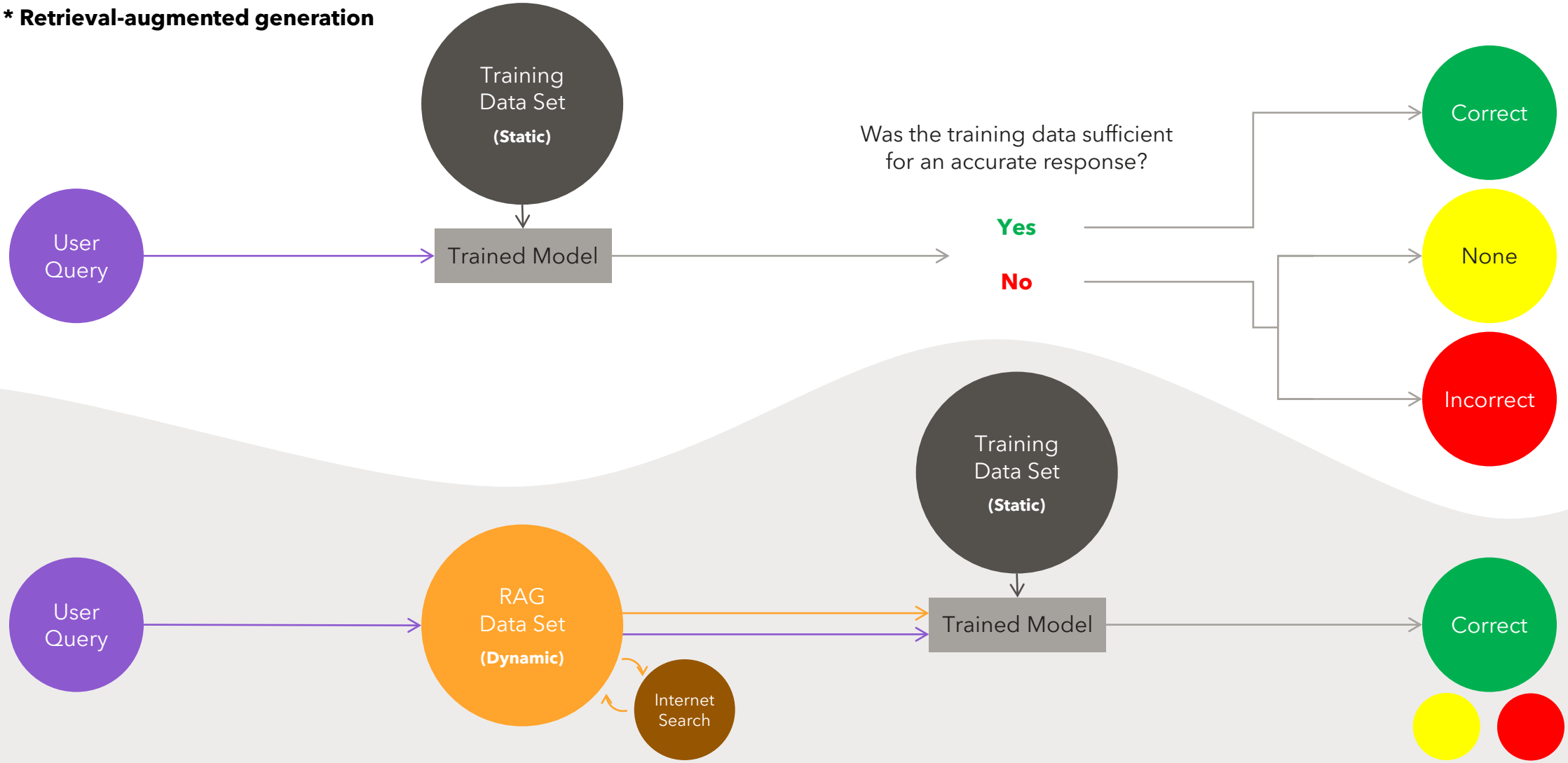


The Problem

 ©2025 Solidigm. All Rights Reserved.

RAG* Refresher: Why It's Useful

*** Retrieval-augmented generation**



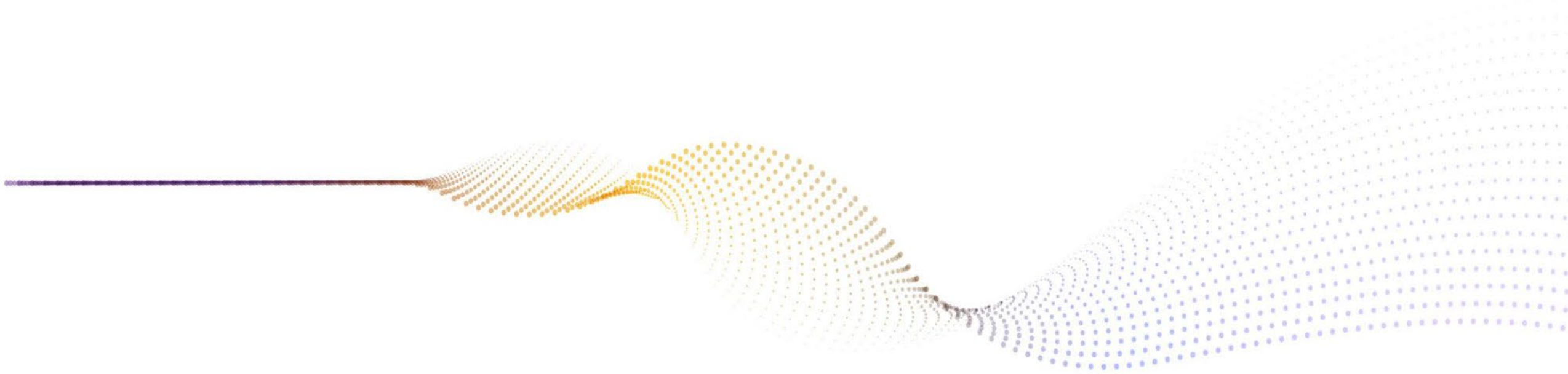
 ©2025 Solidigm. All Rights Reserved.

The Problem

In the pursuit of more valuable insights from AI, enterprises want **bigger RAG data sets** and **more complex models.**

This is very expensive.

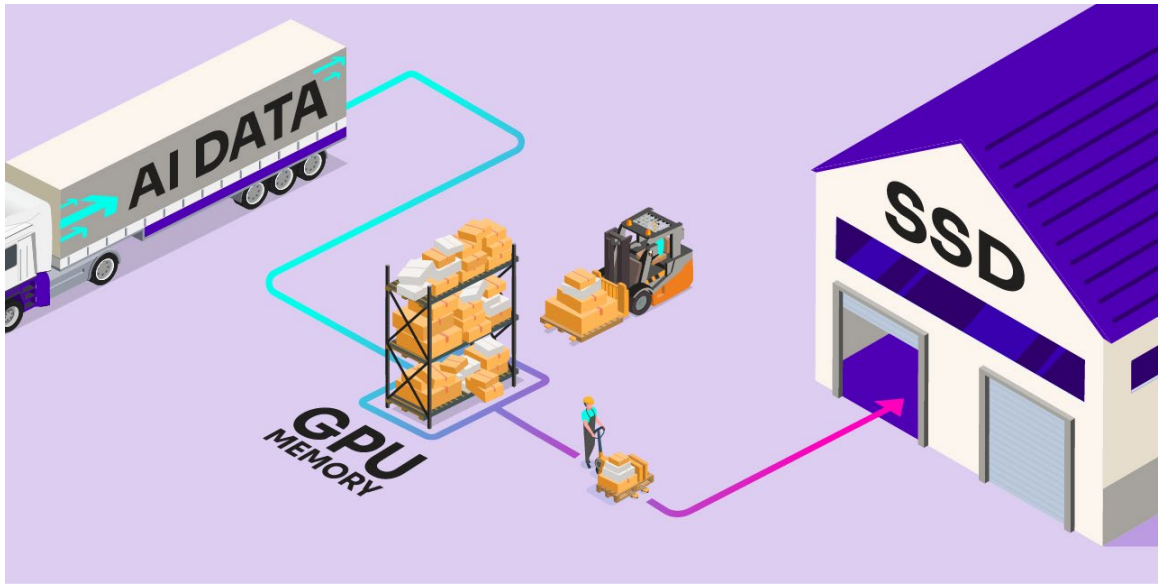
 ©2025 Solidigm. All Rights Reserved.



Experiment and Methodology

 ©2025 Solidigm. All Rights Reserved.

The Central Question



What if we could move significant amounts of AI inference data from GPU memory to lower-cost NAND SSDs?

- What would we gain?
- What would we lose?

SOLIDIGM™

METRUM AI

 ©2025 Solidigm. All Rights Reserved.

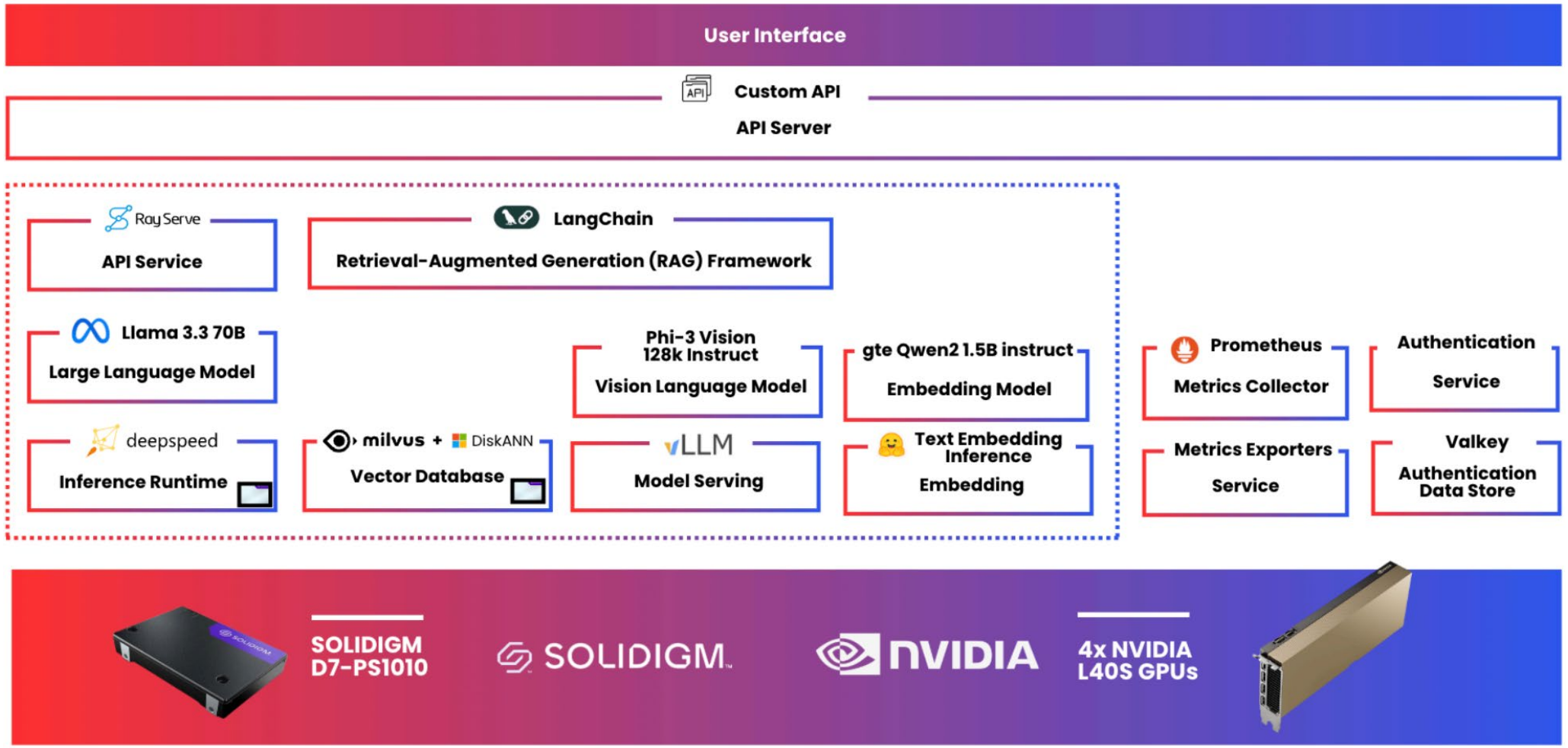
SDC 25

Key Technologies

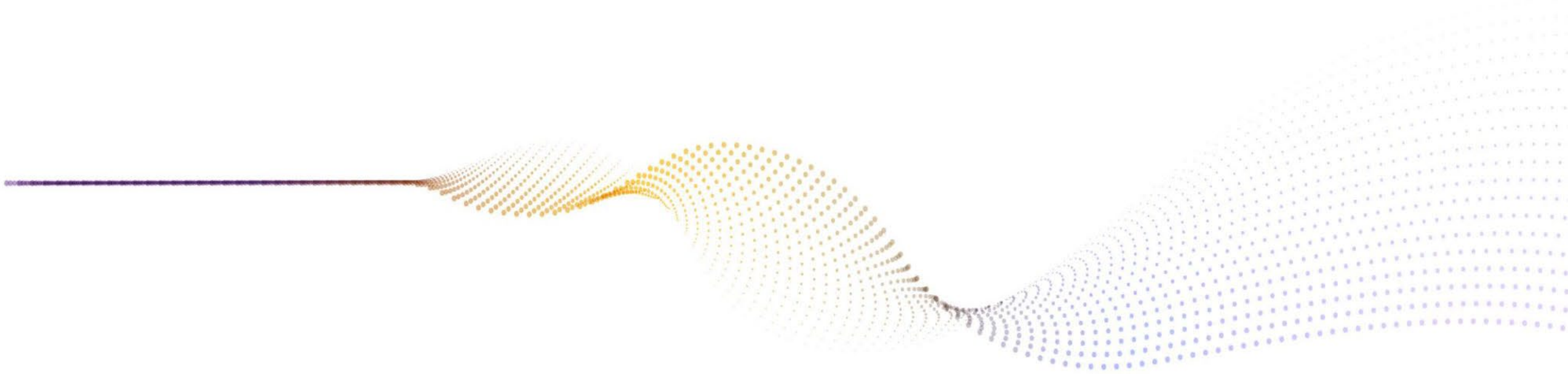
- **DiskANN:** A suite of algorithms for large-scale vector data searches (ANN = "approximate nearest neighbor"). Allows us to relocate part of the RAG data set (index) to SSDs.
- **Ray Serve + DeepSpeed**
  - **Ray Serve:** Scalable model serving library.
  - **DeepSpeed:** Includes DeepNVME component for data transfers between persistent storage and memory.
  - Combination allows us to relocate some model weights to SSDs.

 ©2025 Solidigm. All Rights Reserved.

Architecture: Inference + RAG With SSD Data Offload



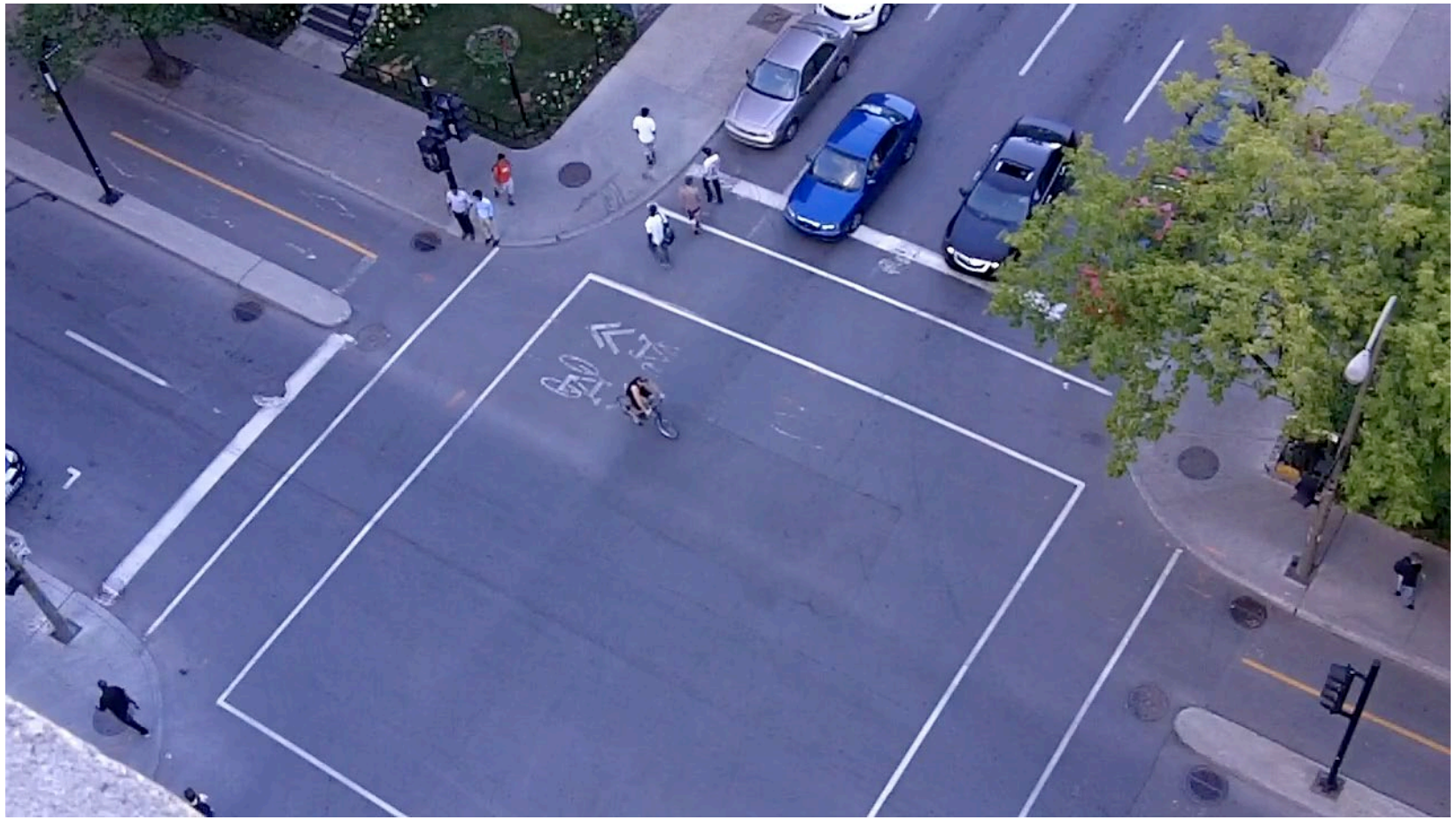
 ©2025 Solidigm. All Rights Reserved.



Example Workload

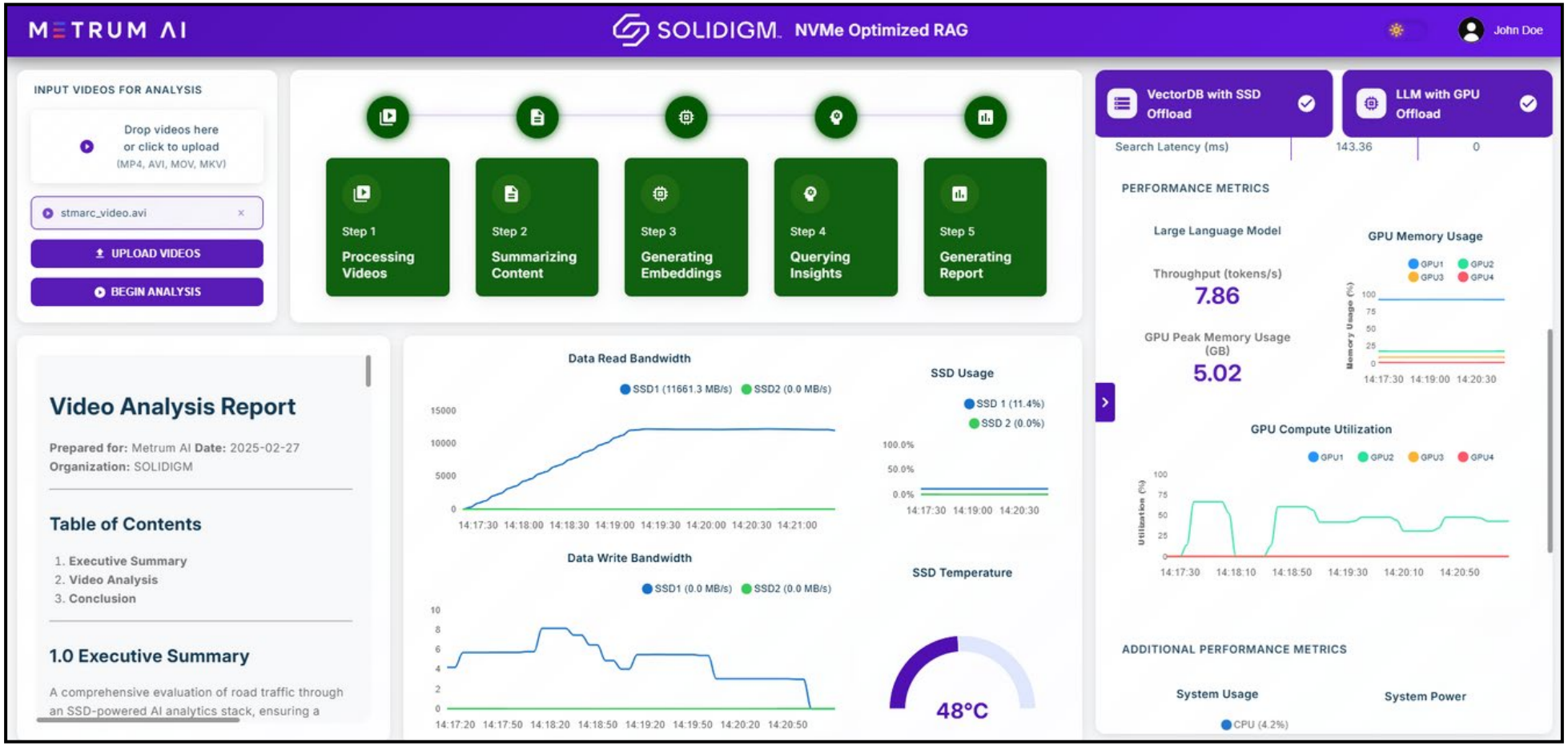
 ©2025 Solidigm. All Rights Reserved.

Source Video: Traffic Intersection



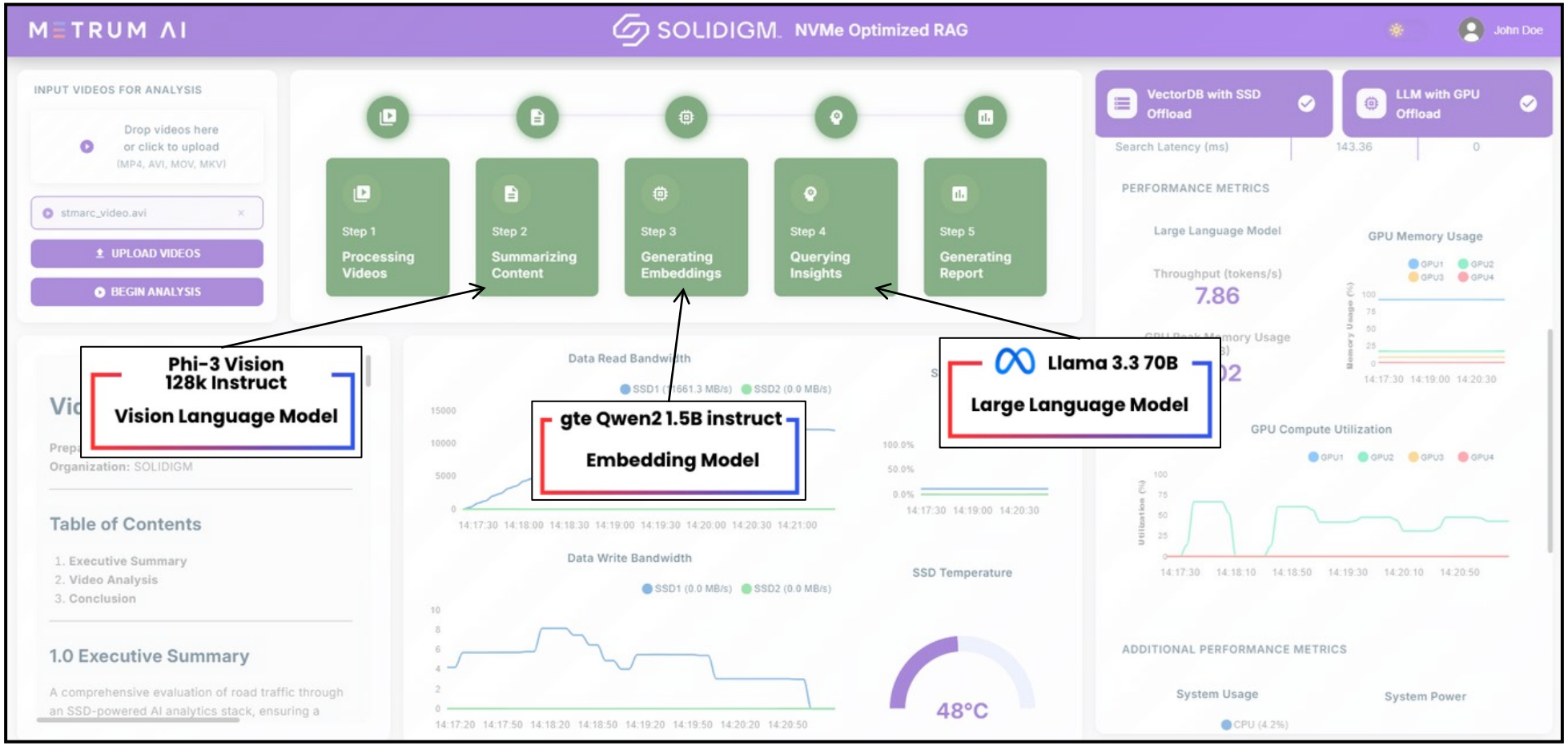
 ©2025 Solidigm. All Rights Reserved.

Dashboard with GPU + I/O Metrics

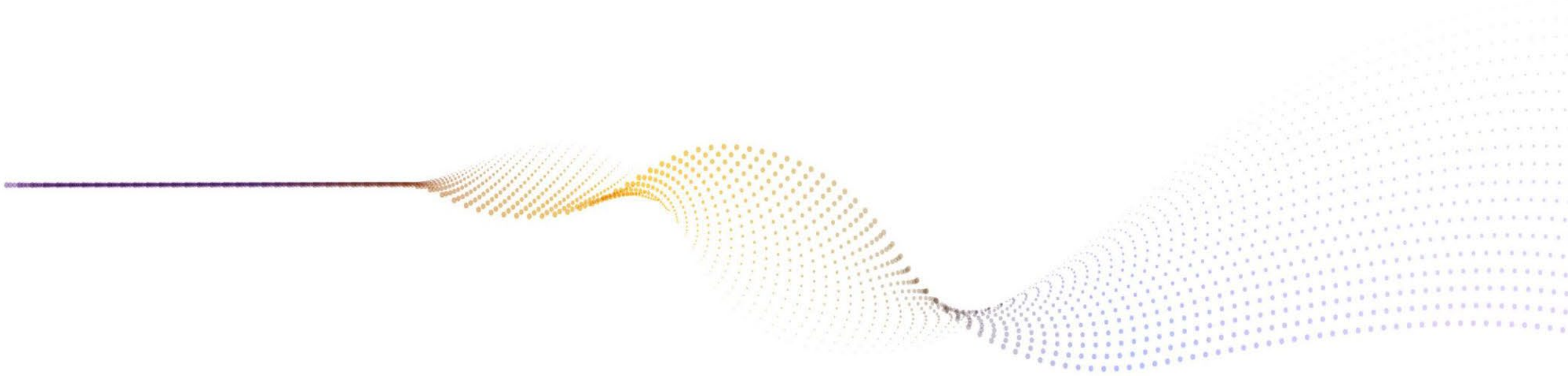


 ©2025 Solidigm. All Rights Reserved.

Dashboard with GPU + I/O Metrics



 ©2025 Solidigm. All Rights Reserved.



Key Findings

 ©2025 Solidigm. All Rights Reserved.

1. 70B Model on L40S

- NVIDIA L40S onboard VRAM: **48GB**
- GPU VRAM requirement for Llama 3.3 70B (FP16): **161GB**
- Peak observed VRAM usage with Ray Serve + DeepSpeed: **7-8GB**

Complex model

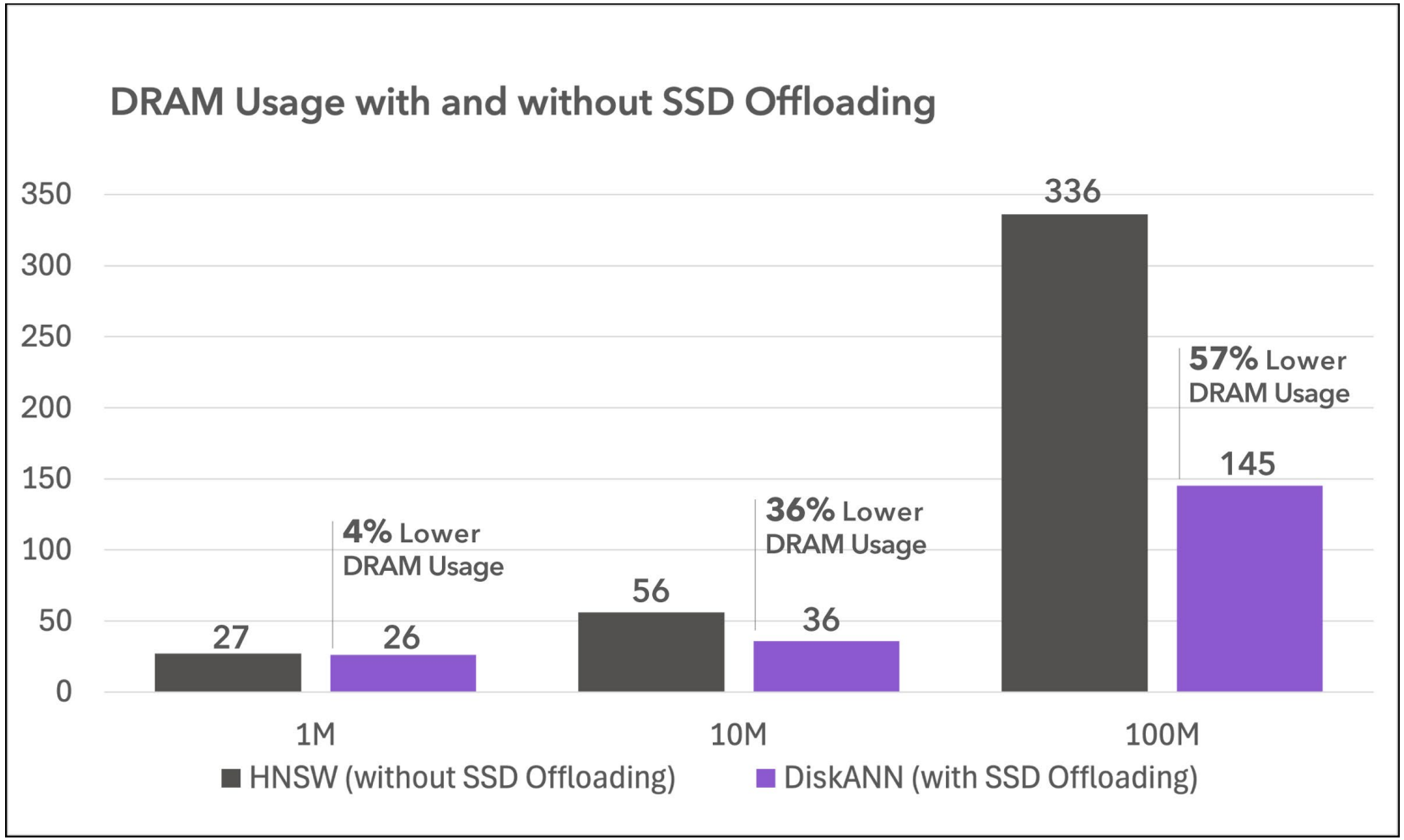
Cost-efficient GPU

20X

reduction in memory requirements

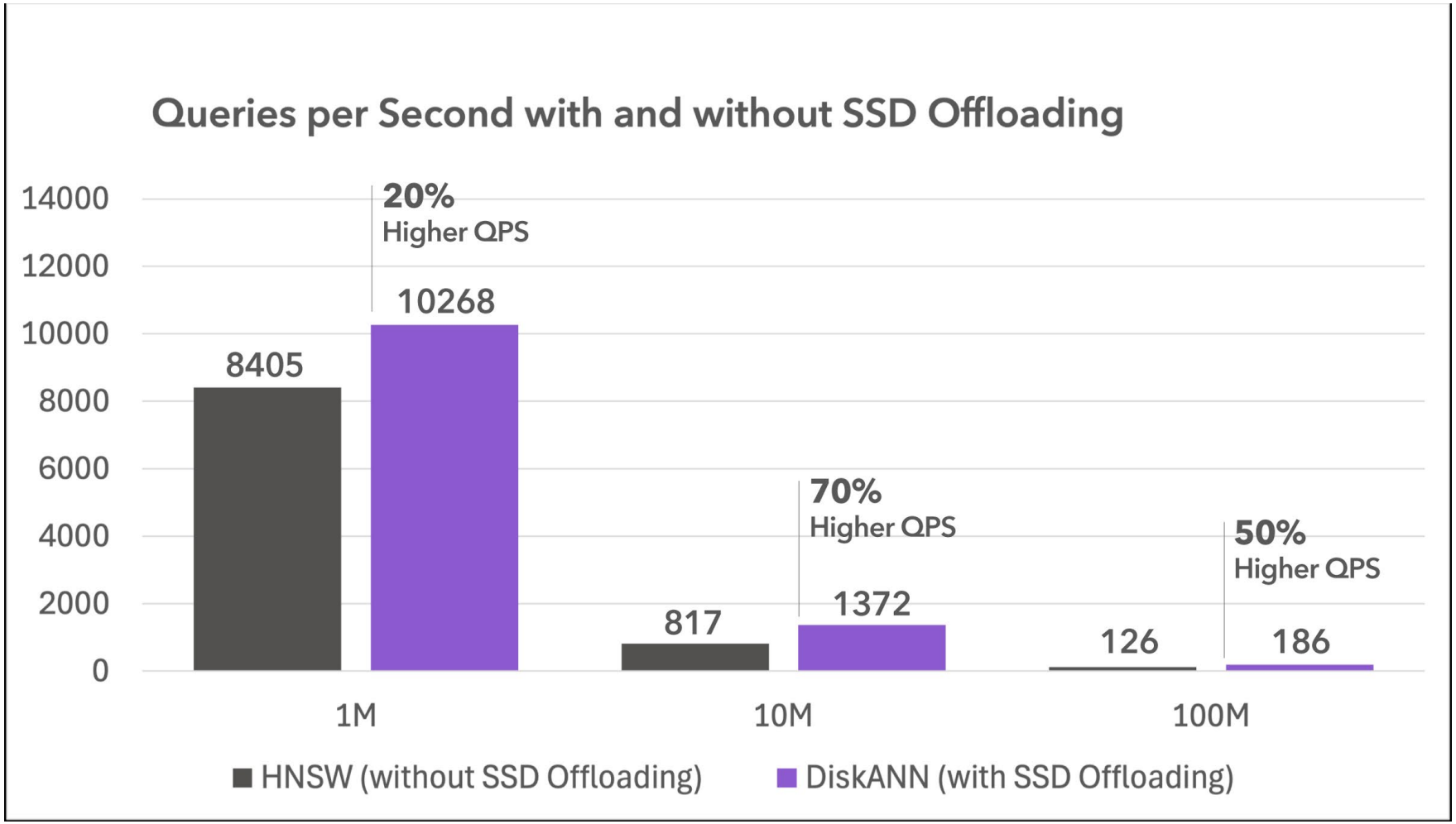
 ©2025 Solidigm. All Rights Reserved.

DRAM Usage: 57% Lower on 100M Vector Dataset



 ©2025 Solidigm. All Rights Reserved.

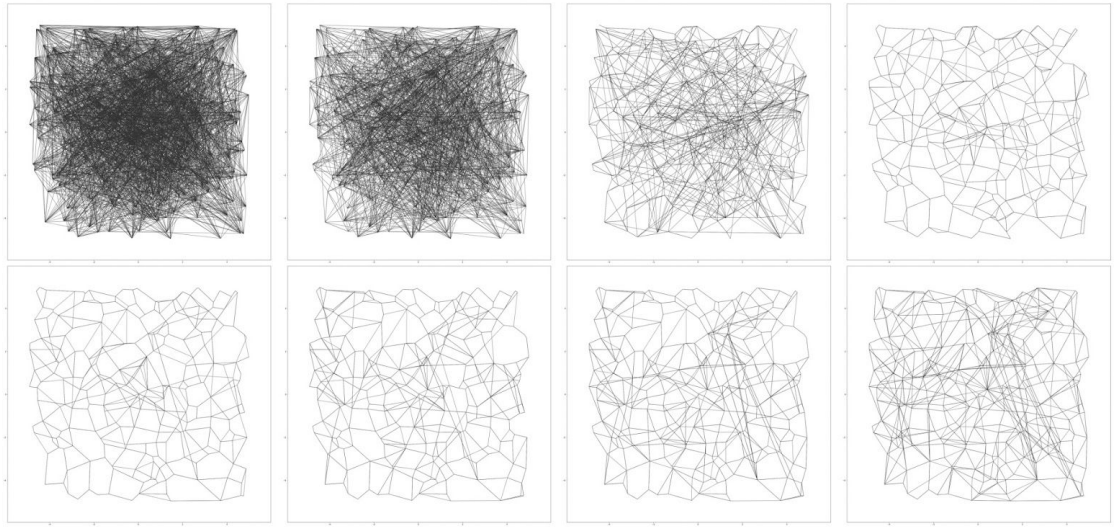
Performance: 50-70% Higher QPS



 ©2025 Solidigm. All Rights Reserved.

Why?

- The **Vamana algorithm** used by DiskANN is incredibly efficient at packing vectors into SSD for quick retrieval
  - Relative neighbor graph with innovations including greedy search and robust prune



Source: Milvus / Zilliz

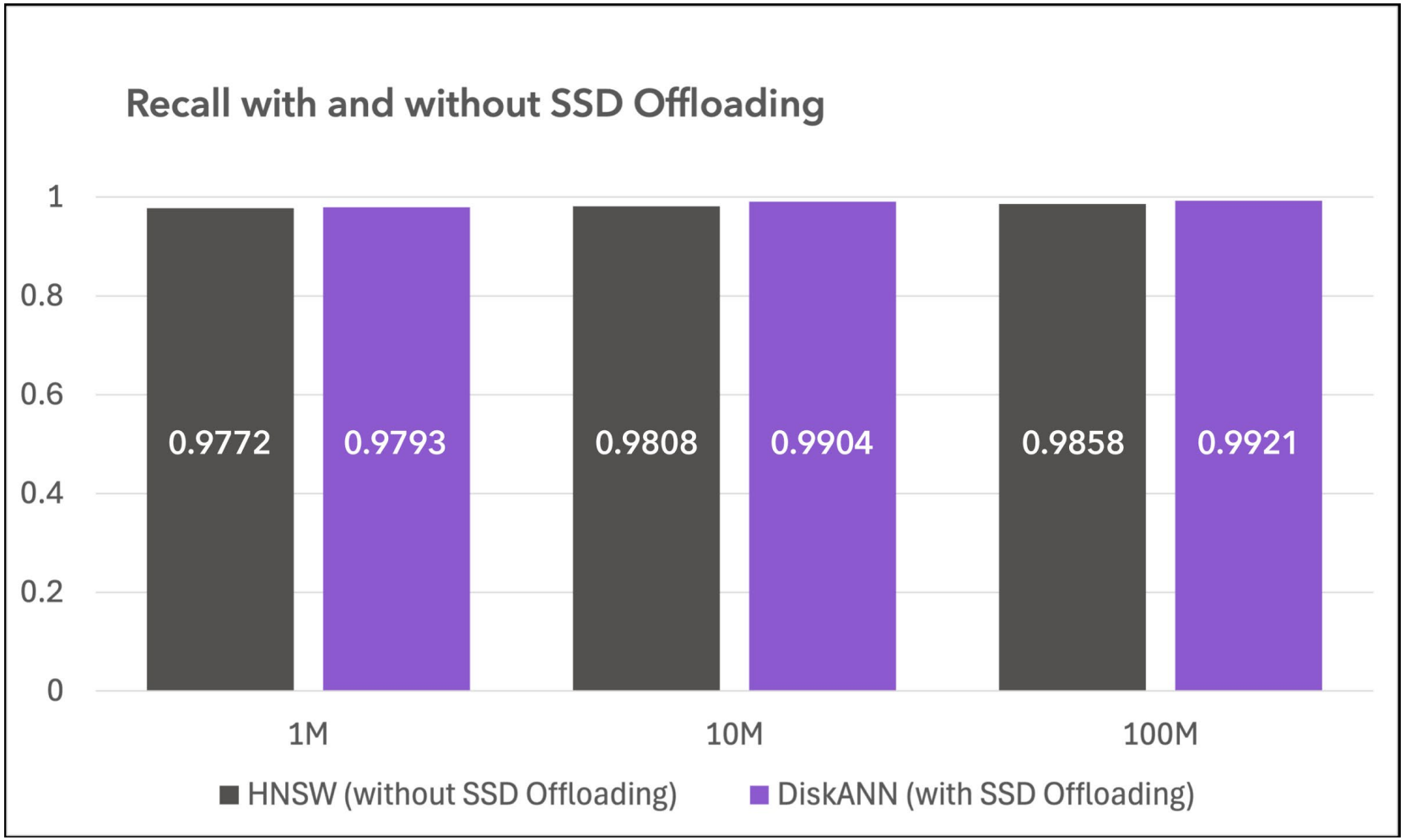
 ©2025 Solidigm. All Rights Reserved.

Cost-Efficient Performance

100M Vector Dataset	HNSW (No SSD Offload)	DiskANN (SSD Offload)
QPS	**126**	**186**
Cost (DRAM+SSD)	**$3,560**	**$2,480**
QPS/$	**0.035**	**0.075**

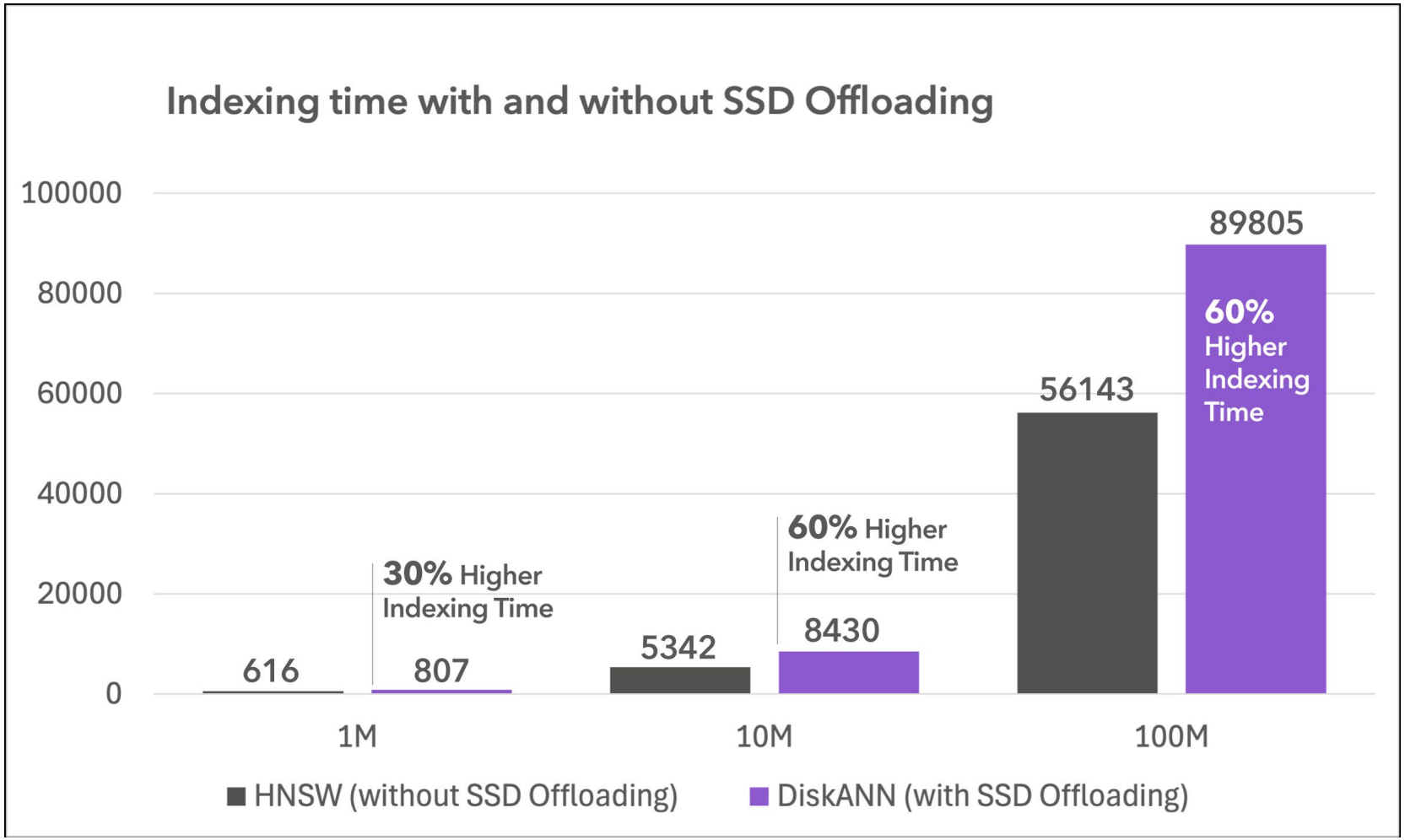
 ©2025 Solidigm. All Rights Reserved.

No Impact to Accuracy



 ©2025 Solidigm. All Rights Reserved.

Key Tradeoff: 30-60% Higher Indexing Time

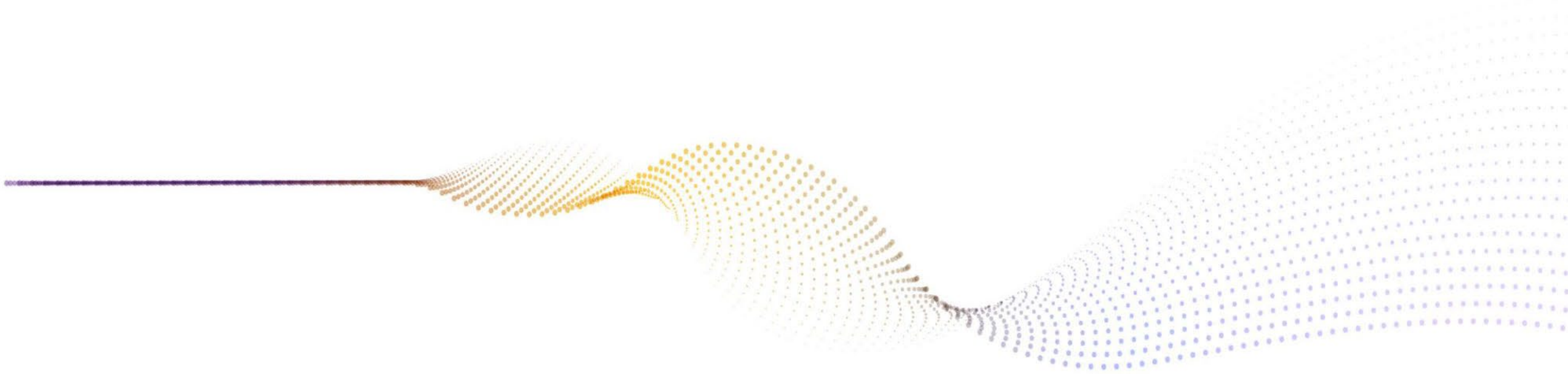


 ©2025 Solidigm. All Rights Reserved.

Try It Out

- This approach is freely available and built on open-source components
- Code and documentation available on Github
- **https://github.com/solidigm → nvme-optimized-rag**

 ©2025 Solidigm. All Rights Reserved.



Next Steps

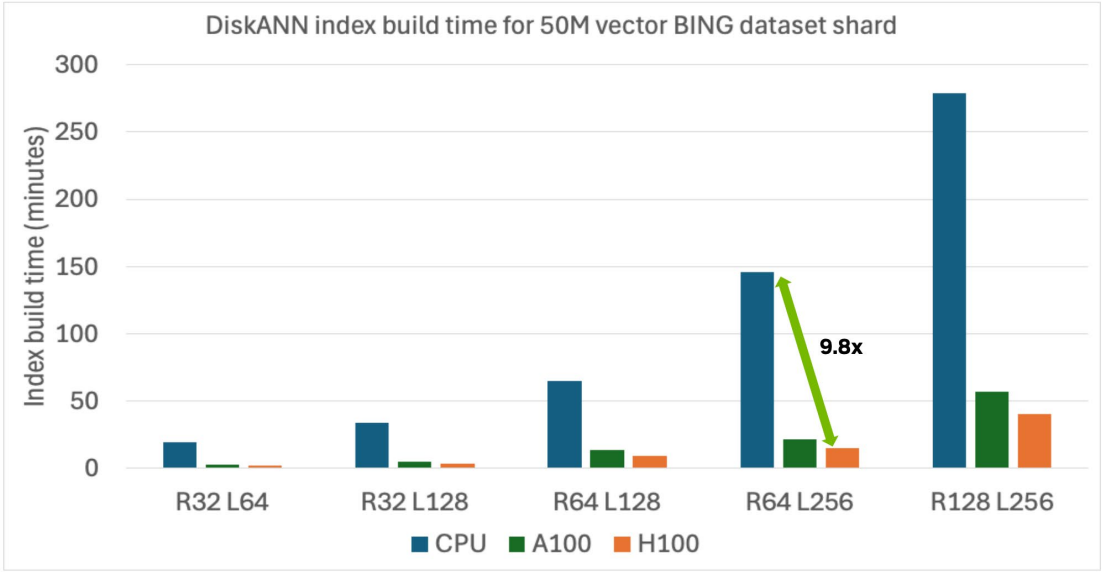
 ©2025 Solidigm. All Rights Reserved.

cuVS GPU-Accelerated Indexing

- NVIDIA cuVS library enables faster vector searches and index builds with GPU acceleration
  - Recently added GPU-accelerated DiskANN index build

(Use case 1) BING Web Search index construction

Comparing CPU and GPU DiskANN index construction



Source: NVIDIA, GTC '25

 ©2025 Solidigm. All Rights Reserved.

Bigger, More Complex, More Distributed Workloads

- More ingest data, heterogenous types, stored on high-density QLC SSDs over the network
- AI agents generating incremental prompts and data
- Chain of reasoning
- Increased solution portability and applicability (generalization)

 ©2025 Solidigm. All Rights Reserved.

DiskANN Enablement

- VectorDB DiskANN support is limited

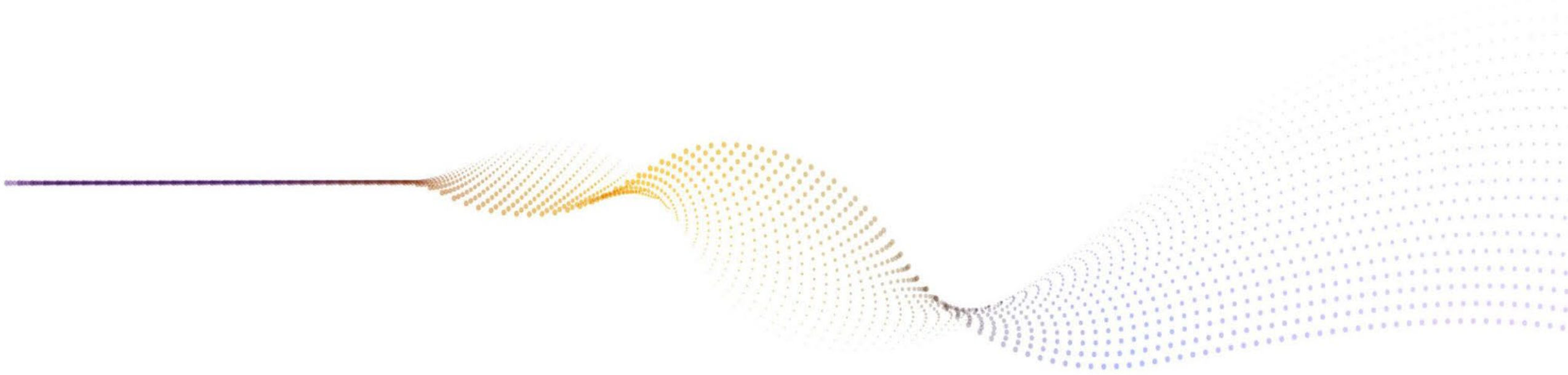
Vector Database	Disk-Based Indexing
Milvus	✅ DiskANN
Qdrant	❌ In-memory only
Chroma	❌ In-memory only
Weaviate	❌ In-memory only
Pinecone	❌ (Proprietary)

 ©2025 Solidigm. All Rights Reserved.

Key Takeaways

- Scaling AI inference with more complex models and bigger RAG datasets ≠ astronomical spend on DRAM
- Fast storage and clever indexing can optimize not just HW BOM cost, but cost-efficient performance
- More work is needed to optimize indexing times and enable this approach using different vector DBs

 ©2025 Solidigm. All Rights Reserved.



Thank you for attending!

Please remember to rate this session. You get access the presentations at
http://sniadeveloper.org/conference

 ©2025 Solidigm. All Rights Reserved.