White paper

VAST Data Vector Database

The Future of AI Data Pipelines

06/25 | Version 1.0

Table of Contents

Introduction 3

Legacy Sharding and Why Architecture Matters 3

The Limits of Sharded Database Architectures 3

VAST's DASE: Disaggregated Shared-Everything 3

From Table Design to S3 Integration 4

Comparison of Legacy Vector Capabilities vs. a Modern Approach 5

PGVector (PostgreSQL + Vector Extension) 5

Milvus (Native Vector Database - Open Source) 5

Pinecone (Managed Vector DBaaS) 6

VAST DataBase Advantages 7

Vision for AI Pipelines 9

Conclusion 10

Introduction

AI workloads are growing exponentially in scale and complexity, pushing the limits of traditional data architectures. Modern AI applications (from large language models to real-time recommender systems) demand ultra-fast access to massive datasets, continuous ingestion of new data, and seamless integration of structured and unstructured information. Traditional databases and data lakes, even when augmented with caching or distributed frameworks, are struggling to meet these demands. They quickly reach limits in throughput and scalability when faced with trillions of records, millisecond latency requirements, and complex pipelines that mix transaction updates with analytical queries. As a result, organizations have been forced to stitch together disparate systems (OLTP databases, data warehouses, file/object stores, vector search engines, streaming processors) to handle AI data workflows. This fragmented approach introduces significant complexity, latency, and operational overhead, often becoming a bottleneck in AI development and deployment.

The **VAST AI Operating System** – and specifically its vector-optimized database component – represents a new breed of data infrastructure designed for the AI era. By combining the transactional performance of databases with the scalable analytics of data lakes, and by bridging structured and unstructured data in one system, VAST aims to eliminate the traditional trade-offs. This white paper will explain why VAST's vector database, built on a Disaggregated Shared-Everything (DASE) architecture, is uniquely suited to power future AI data pipelines. We will discuss how VAST's approach overcomes the sharding complexity of other systems, how it compares to both retrofitted vector solutions such as PGVector for PostgreSQL and native vector databases such as Milvus and Pinecone, and how its design principles of small-columnar chunks, integrated object storage, and real-time triggers provide a superior foundation for AI/ML workflows.

Legacy Sharding and Why Architecture Matters

The Limits of Sharded Database Architectures

Most modern distributed databases achieve scale by sharding data at the database software layer – partitioning tables or indexes across nodes. For example, a traditional RDBMS cluster (or even NoSQL stores) might require you to manually decide which subset of data lives on which node (or use a hash/range partition key). This shared-nothing approach introduces complexity and inefficiency. Developers and DBAs must spend significant effort on partitioning strategy, rebalancing shards, and handling cross-shard queries or transactions. Each node manages only its portion of data, so a large query often needs to fan out to all shards and then merge results, incurring heavy network (east-west) traffic and coordination overhead. As data and workload scale, these architectures hit a wall: adding more nodes increases overhead, scaling becomes complicated and "hot" shards or uneven data distribution can bottleneck performance.

VAST's DASE: Disaggregated Shared-Everything

VAST Data takes a fundamentally different approach by separating the compute tier from the storage tier. Each VAST cluster has two primary components:

1. **Stateless Compute Nodes**

   These containers run the VAST database engine but do not own any specific slice of the data. Instead, they connect to all storage devices in the cluster over NVMe-oF (NVMe over Fabrics). Because every compute node can see all the data in the system, the system avoids local data caches or the need to coordinate among partition owners. Any compute node can address any piece of data on demand, enabling linear, shared-nothing scalability without complex sharding.

2. **Highly Available Data Nodes (Storage)**

   The storage tier comprises NVMe enclosures with Storage Class Memory (SCM) for ultra-fast writes and metadata, plus high-capacity QLC flash for the bulk of data. SCM devices absorb incoming writes at low latency, providing a global write buffer that eliminates hotspots. Once written, data is chunked into durable stripes and stored across QLC SSDs. This separation of duties allows the system to efficiently handle both high-IOPS workloads and massive capacity, leveraging QLC's density for cost-effective scale.

VAST further optimizes data management with advanced data reduction technologies, including inline compression, deduplication, and "similarity" reduction. Every write is examined in memory (using SCM as a buffer) to find common patterns or near-duplicates across the global dataset. These techniques are especially effective with large embeddings or media files, since repeated segments can be eliminated globally rather than on a single node. As a result, organizations minimize physical storage footprint even at multi-petabyte scale.

Putting it all together, VAST's DASE architecture grants every compute node direct access to a shared, globally optimized data pool backed by flash. The system avoids traditional sharding overhead, ensures consistent low-latency reads and writes, and achieves high resiliency through broad striping. Coupled with universal data reduction, this design offers a streamlined foundation for demanding AI workflows, from large-scale vector searches to deep learning inference.

This architecture delivers two key benefits:

1. **Linear scalability**

   Need more throughput or query capacity? Just add more stateless servers – they all immediately contribute to serving the same data, with no re-partitioning necessary. There is no notion of a "primary node" for a particular partition that could become a bottleneck; all nodes share the load for all data. VAST's DASE design has been shown to scale active data and query throughput linearly across thousands of cores.

2. **Greatly simplified data management**

   The lack of inter-node data silos means no complicated resharding, failover of partition owners, or distributed transaction coordination in the typical sense. In fact, VAST's global data structures allow each write to occur without coordinating with other CPUs, using an innovative consistency mechanism built into the storage fabric. The result is a platform that delivers simple and resilient operation even at extreme scales (exabyte storage, billions or trillions of objects).

From Table Design to S3 Integration

Unlike monolithic databases that were retrofitted to work on distributed file storage, and systems fitted over inexpensive object storage, the VAST DataBase is a VAST-native table format built on an architecture that solves the issues with flash storage costs. Tables are built with a 32KB chunk size that enables fast transactional capability, instant search capability and high-throughput all at once. The system is built for massive concurrency, utilizing distributed parallel tree-scans that can quickly locate unsorted data and advanced indexes and adaptive table ordering for similarity search. With traditional storage trade-offs broken, it's now possible to build a massive-scale vector database that grows and performs effortlessly without management complexity or operational trade-offs.

Comparison of Legacy Vector Capabilities vs. a Modern Approach

The AI database landscape can be confusing. Some options are adaptations of legacy systems, while others are specialized but narrow in scope. As AI applications increasingly rely on vector embeddings—whether for similarity search or semantic queries—engineers typically face two choices: bolt on vector functionality to a traditional database, or deploy a specialized vector engine. Each route brings potential trade-offs, from sharding overhead to weak operational features.

By contrast, the VAST AI Operating System was engineered from the ground up for AI-era workloads. Its disaggregated, shared-everything architecture avoids the scaling pitfalls of older systems and the siloed complexity of specialized engines. Instead, VAST unifies vector search with robust data management, delivering a solution that can future-proof AI initiatives for the long haul. Below is a look at popular alternatives and the critical limitations that VAST overcomes.



PGVector (PostgreSQL + Vector Extension)

PGVector extends PostgreSQL with vector similarity search but inherits Postgres's inherent scaling challenges. Large datasets often require manual sharding or replication, adding operational overhead and complexity. Performance can degrade if the index outgrows local RAM or SSDs, especially under mixed OLTP and vector query loads. Moreover, running vector operations alongside traditional SQL queries can create resource contention, and maintaining a separate vector-enabled Postgres cluster under high throughput can be cumbersome.

By contrast, the VAST Database uses a disaggregated, shared-everything architecture to eliminate manual sharding and local storage constraints. Vectors and metadata reside in the same globally accessible space, so any compute node can perform searches without bottlenecks or exhaustive broadcasting. This streamlined design preserves Postgres-like transactional capabilities while scaling linearly for AI workloads, making VAST more suitable for enterprise-scale vector search.



Milvus (Native Vector Database - Open Source)

Milvus relies heavily on memory and optional GPU acceleration to deliver fast similarity searches. This design forces most or all of the vector index into RAM or GPU memory, so when the dataset grows beyond a node's resources, operators must shard or partition the data among multiple query nodes. Complexities arise: queries must often broadcast to all nodes, and additional GPUs can be expensive or impractical. Even though newer on-disk indexes reduce some memory pressure, purely disk-based searches remain slower.

Milvus also provides only basic support for storing scalar fields and metadata, requiring external systems for richer queries or managing raw documents. As a relatively new platform, enterprise-level features like replication and high availability still require additional tuning and care.

By contrast, the VAST DataBase avoids these limitations. It places all data—vectors, metadata, and raw objects—on a single, scalable layer that doesn't depend on GPU memory sizing but can still take advantage of GPUs for rapid responses. Rather than broadcasting queries to multiple partitions, VAST's disaggregated, shared-everything design lets any node address the entire dataset, eliminating artificial sharding. This lowers infrastructure costs, reduces operational complexity, and removes the need for a two-step retrieval process, making VAST a more holistic and scalable option for advanced vector search.



Pinecone (Managed Vector DBaaS)

Pinecone's fully managed approach can simplify vector search for small teams, but it comes with significant drawbacks for enterprise-scale operations. Its proprietary SaaS model enforces vendor lock-in and limits hardware choices, making on-prem deployments or specialized hardware configurations nearly impossible. Scaling relies on fixed-size "pods," so large indexes must be split across multiple pods, adding latency and complexity. Pinecone also keeps extensive data in external stores, forcing two-step retrieval and risking inconsistencies. Lastly, high query volumes may incur steep network costs, and Pinecone's evolving enterprise features (like security or self-hosting) lag behind more mature platforms.

In contrast, the VAST Database solves these issues by co-locating all data—vectors and metadata—in a single environment that can scale seamlessly without artificial pod limits or sharding. Its disaggregated, shared-everything design avoids the constraints of fixed SaaS pods, while native integration of vector indexing and data management eliminates the two-step retrieval process. As a result, VAST delivers lower latency, stronger consistency, and flexibility to run on-prem or in the cloud, offering a more robust alternative for enterprises requiring large-scale vector search.

In summary, traditional databases with vector add-ons offer mature management but struggle to scale vector search, while specialized vector databases offer speed but force new silos with operational trade-offs. VAST Data's vector database seeks to combine the best of both worlds: the scalability and simplicity of an integrated platform with the performance of a specialized engine. Below, we detail how VAST achieves this.

VAST DataBase Advantages



Built for Vector-Native Workloads

The VAST DataBase was architected from the ground up to treat vectors as a first-class data type alongside traditional structured data. You can store an embedding vector in the same table as its associated metadata (IDs, text, categories, etc.), and query them together seamlessly. Because of the disaggregated shared-everything design, every query node can search the entire vector space in parallel – there is no need to manually split the index or limit the search scope. This stands in contrast to other solutions that partition the vector space. VAST's vector index is globally accessible and updated in real time as data is ingested, supporting trillion-vector scale without degrading as the dataset grows. Organizations can index all enterprise data (images, text, audio, etc.) in one place, confident that similarity searches will not slow down at scale.



No Sharding & No Separate Silos

Unlike PGVector, you do not need to spin up multiple database instances, and unlike Milvus or Pinecone, you don't need a separate object store for large data or metadata. VAST co-locates vectors with the original data ("chunks") in the same platform. This means when a similarity search returns a set of vector IDs, the system can directly retrieve the corresponding records (including images, text, etc.) from the storage layer immediately. There is no "fetch from S3" second hop, so end-to-end latency is lower. It also guarantees transactional consistency: you insert or update a row with a vector and associated fields in one atomic operation, and queries reflect it immediately. Many competing systems only offer eventual consistency when referencing objects in external stores.



High-Performance Indexing and Retrieval

VAST's indexing approach does more work at write time so that reads are as fast as possible. On data ingestion, the system automatically inserts your data into various index structures (zone maps for scalar fields, the main vector index, and any secondary indexes). These indexes live alongside the data chunks and can use CPU or GPU-accelerated algorithms like CAGRA. As a result, when a query arrives, VAST can prune irrelevant chunks in microseconds and quickly retrieve the nearest vectors. This also means GPU-accelerated retrieval can be performed without having to cache the index in GPU memory. The small, columnar chunk format also benefits analytics on embeddings (like computing aggregate statistics or transformations), as data can be loaded in contiguous blocks. Additionally, VAST supports projections – precomputed partial materializations or sorted orders – to optimize common query patterns without scanning large volumes of data.



Row and Column Level Security

One of the biggest challenges faced by customers looking to expose private datasets to chatbots and AI agents is the fear that by doing so either the raw data will be exposed to individuals who don't otherwise have access, or it will be incorporated into decisions and analyses inadvertently. VAST's DataBase provides out of the box row and column level security. This means that as data lands on the platform the permissions associated with the source data, even if it is from a SharePoint or Google Drive are preserved, thus limiting what indexes and queries it is visible to. Further VAST has robust audit capabilities. So as that data is accessed by various AI training, fine-tuning and inference functions an audit trail is preserved and most importantly that audit trail is itself stored in a table that can be rapidly queried by any number of security solutions.



Architectural Simplicity and Reliability

DASE eliminates the single point of failure or bottleneck seen in systems that rely on a specialized "index server." All compute nodes can serve all queries, and if one fails, others seamlessly pick up the load. Adding new compute nodes instantly scales capacity and query throughput without data reorganization. VAST's storage layer uses strong consistency mechanisms (like decentralized locks at the file or object level), so even at huge scale the system maintains ACID guarantees. This means no complicated eventual-consistency pitfalls: if an update is committed, subsequent queries see it. For AI applications, this reliability is crucial, especially when mixing real-time ingestion with analytical queries.



Graph-Style and Multi-Relational Workloads

Many AI scenarios require not just vector search but also the ability to join or traverse data. VAST supports flexible schemas (including arrays, maps, and nested data). Any node can see the entire dataset, enabling true global algorithms without moving data. Combining vectors with context is trivial; for example, you can run a similarity search within a subset of data identified by a graph relationship or a relational join. In effect, VAST unifies vector search, relational queries, and even file-based data into one place.



Native S3 Access and Data Locality

Training frameworks like TensorFlow or PyTorch can pull data in parallel using standard S3 APIs, bypassing the overhead of copying data to local storage. Because VAST has no tiering of data and can leverage GPU Direct Storage or NVMe-oF, data can stream to GPUs at wire speed, keeping expensive accelerators busy. This addresses a common AI bottleneck where GPUs idle waiting for data. The single data fabric means you do not worry about synchronization across multiple systems.

In short, the VAST DataBase offers unmatched scalability, simplicity, and versatility for vector and graph-style workloads. It provides the performance of specialized engines (through advanced indexing and flash-optimized design) alongside the convenience of a single data hub. This makes it ideal for next-generation AI applications where multiple data types and query paradigms converge.

Vision for AI Pipelines

Modern AI pipelines involve ingestion of raw data, transformation to embeddings or features, indexing for quick lookup, and real-time inference or periodic model training. VAST's platform is designed to streamline this entire flow in real time without bridging multiple systems.



Real-Time Data Ingestion and Vectorization

VAST can ingest high-volume data streams directly, buffering writes in persistent memory before eventually flushing to flash. Then, serverless functions can automatically trigger to compute embeddings or metadata from new data. Because compute is co-located, no external pipeline is required for immediate indexing.



Unified Batch and Stream Processing

Traditional pipelines separate real-time data from historical data, often placing them into different stores. VAST provides a single, strictly consistent namespace for both. An event-driven function can handle incremental updates, ensuring that new data is searchable within milliseconds. Background jobs can train or update models using both new and historical data without duplication.



Acceleration of Model Training

Model training jobs typically need to read large volumes of data in parallel. VAST's all-flash storage and distributed architecture allow many GPUs to read from a single source at low latency. Because data is chunked at 32 KB boundaries, partial reads or random sampling are efficient. VAST also supports snapshot semantics so training can read a consistent view of data even while new writes occur.



Low-Latency Inferencing and RAG

In retrieval-augmented generation or other real-time inference workflows, queries need to rapidly locate relevant documents, images, or text chunks by embedding similarity. VAST performs the vector search on the same platform that stores and returns the underlying data, minimizing round-trip latency. Fine-grained security controls can also ensure compliance by restricting which data each user or system can query.



Scalable Compute and Serverless Extensibility

As workloads grow, adding more stateless compute nodes increases parallel ingestion, indexing, query processing, and function execution. Developers can embed custom AI logic into the platform via serverless functions, pushing transformations close to the data. This reduces network traffic and speeds up iterative AI pipelines, a critical advantage for large-scale or time-sensitive applications.



End-to-End Security and Governance

Because VAST is a single system spanning files, objects, tables, and vectors, it can apply consistent security, encryption, and auditing policies across all data. Meeting compliance requirements is simpler when updates or deletions are handled atomically in one platform rather than across multiple external stores.

Conclusion

AI-centric companies are increasingly looking for data platforms that can bridge the gap between unstructured data, structured data, and real-time pipeline needs – all at massive scale. The VAST Data vector database is uniquely positioned to be that bridge. By fusing the capabilities of a transactional database, a data warehouse, a vector search engine, and an object store into one cohesive architecture, VAST eliminates the fragmentation that plagues today's AI data stacks. It offers the performance and scale required for cutting-edge AI (trillion-vector constant-time search, exabyte-scale capacity, GPU-optimized I/O), without forcing a trade-off with consistency, security, or ease of use.

We have seen that traditional databases augmented with vector search quickly run into scaling and performance issues, while standalone vector databases lack data management richness and operational maturity. VAST resolves these dichotomies: its disaggregated shared-everything design provides linear scale-out simply by adding resources, with no re-partitioning or re-engineering of the data model. Its shared global namespace means all data (whether a log entry, a table row, an image, or a vector embedding) lives in one logical space, so applications no longer need complex orchestration to get a complete view of data. Structured and unstructured data converge in VAST's DataSpace, enabling AI models to seamlessly draw on both context and content.

For technical leaders, the implications are profound: a simpler architecture that is easier to maintain and scale, a faster path from data to insight, and a future-proof foundation for AI workloads. VAST's platform allows organizations to focus on building AI-driven features, rather than wrangling data pipelines. By unifying vector search, event-driven processing, and analytics in one platform, it removes the complexity and latency that come with stitching together point solutions. The result is not just incremental improvement, but a step-change in capability: real-time AI on fresh data, secure and governed by default, with the ability to handle any data type or query pattern.

In conclusion, the VAST Data vector database represents the best choice for future AI data pipelines because it was designed for exactly that purpose. It breaks long-standing trade-offs – delivering both high-performance and high-capacity, both transactional and analytical processing, both consistency and global scale. It brings the reliability and familiarity of database systems to the new world of AI and vectors, and the flexibility of a data lake to the rigor of enterprise IT. For organizations preparing for the next wave of AI innovation, VAST provides a rock-solid yet forward-looking foundation to build on. It is, in a phrase, AI-native infrastructure – allowing any enterprise to leverage their vast data in the most intelligent and efficient way possible.

©2025 VAST Data, Inc. All rights reserved. All trademarks belong to their respective owners.