Cloud Infrastructure Analysis

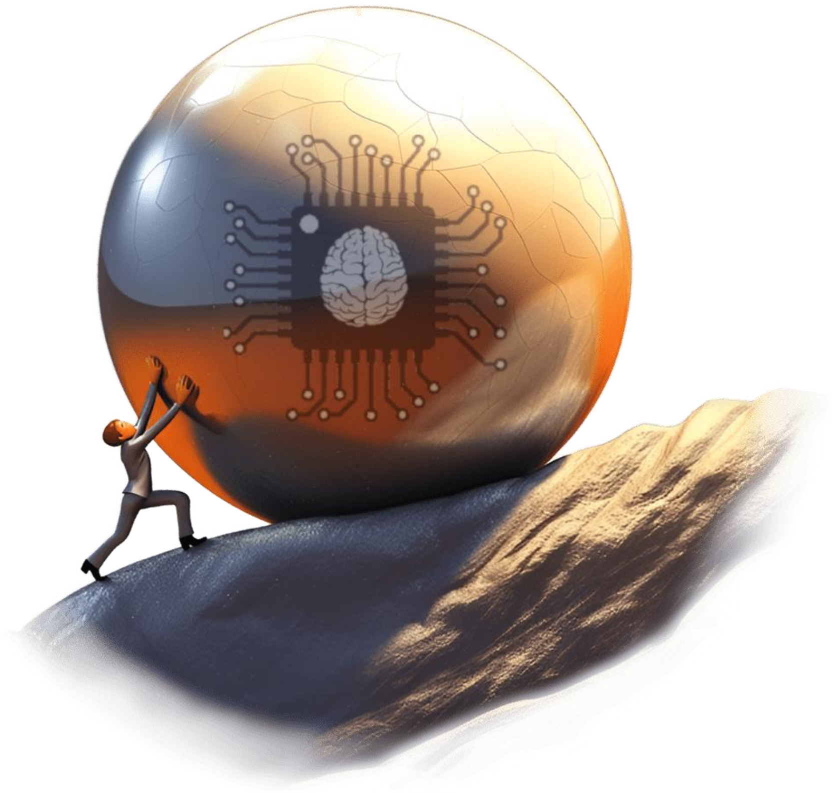
How Oracle Accelerates Enterprise Vector Database & Vector Search Adoption

2024

Written by: Marc Staimer, Sr. Analyst, theCUBE Research, & President, Dragon Slayer Consulting

Introduction

This research assesses database related obstacles to Generative AI (GenAI) and Large Language Model (LLM) adoption, why IT organizations assume they just have to accept them, and how Oracle has changed the game. Of course it starts with the questions: “What is GenAI, LLMs, and the role of Retrieval Augmented Generation (RAG), vector databases, and vector search?”



GenAI is a relatively new specific type of AI[1]. Even though it’s been around for more than a decade, the real breakthroughs came about over the past 3 years. GenAI creates or generates content. That content can be text, images, audio, video, visual art, conversations, code, reports, summaries, and more.

Only IT professionals and consumers completely off the grid have not heard about these new GenAI models such as OpenAI ChatGPT, Google Gemini, Meta Llama, Microsoft Vall-E, Amazon CodeWhisperer, Oracle GenAI Service, and many more.

[1] Previous AI models historically have been and still are focused on improving data processing, data analyzation, and data interpretation. These are commonly referred to as process and predictive AI.

The most common GenAI models—generative adversarial networks (GANs), variational autoencoders (VAEs), diffusion models, transformers, and neural radiance fields (NeRFs)—are built on unique machine learning (ML) algorithms. Each of these algorithms have their own competencies.

What excites most consumers and IT pros is the Large Language Model (LLM) found at the core of every GenAI environment. LLMs are sophisticated machine learning (ML) and deep machine learning (DML) models. The training is commonly based on massive amounts of data found primarily from accessible sources such as the Internet. Trained LLMs are referred to as foundational models (FM). These FMs are often fine-tuned and customized to improve their accuracy and reduce what are called LLM hallucinations, or inaccuracies – results that appear correct and rational but are not. One needs to keep in mind that the information contained in an LLM is current as of when the training ended. Keeping an LLM up to date by retraining is very expensive. Retrieval Augmented Generation provides a much less expensive option to provide answers based on up-to-date information.

What's rapidly become the current best accepted methodology to considerably mitigate those hallucinations is Retrieval Augmented Generation (RAG) as depicted in Figure 1.

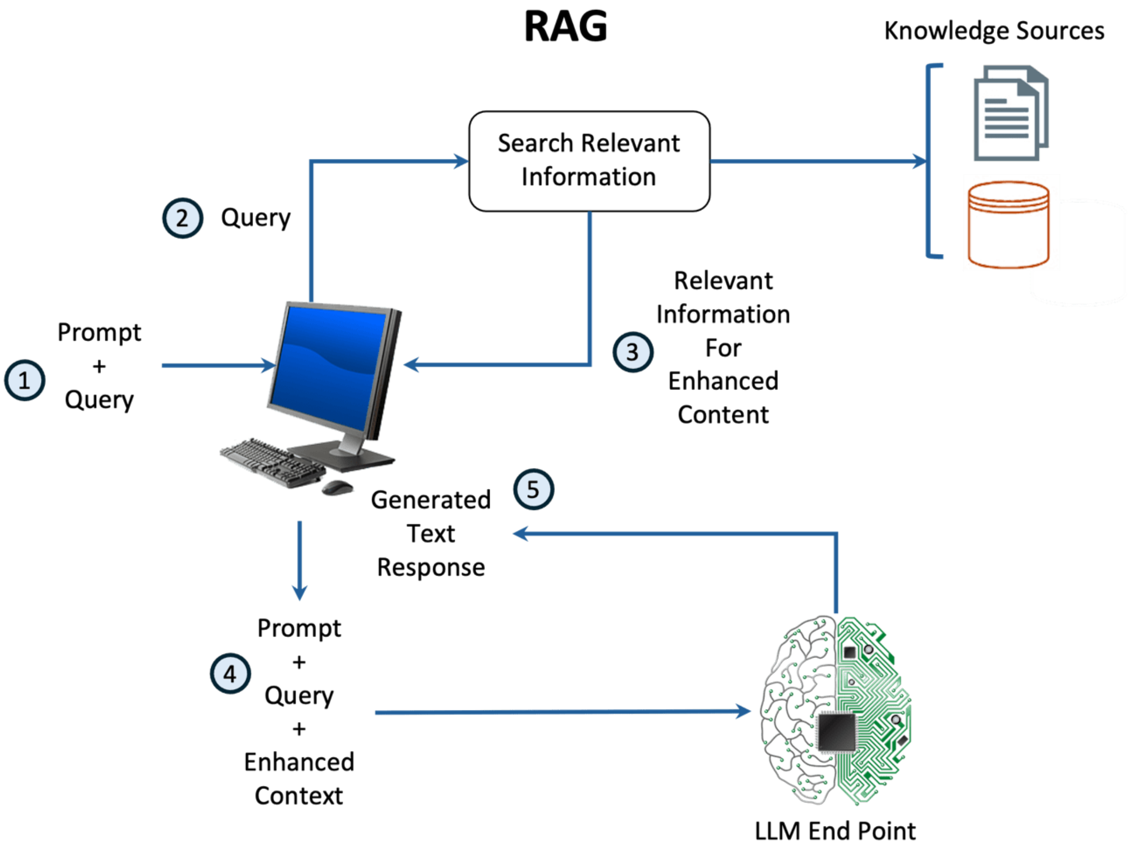

RAG
Knowledge Sources
Search Relevant Information
2 Query
1 Prompt + Query
3 Relevant Information For Enhanced Content
5 Generated Text Response
4 Prompt + Query + Enhanced Context
LLM End Point

Figure 1: Retrieval Augmented Generation

RAG generally leverages internal data from databases, files, and data lakes. This is indispensable for enterprises and government organizations' effective use of open and shared GenAI LLMs. They do not want their information being used outside their organization and it helps them meet evolving regulations.

Retrieving relevant data requires encoding of unstructured data such as documents, video, audio, photographs, illustrations, drawings, spreadsheets, their associated metadata, and more using an embedding model that abstracts features of unstructured data into a set of numeric values that are thought of mathematically as vectors. Vectors represent the data content and not the underlying words or pixels. These vector embeddings are then indexed and mapped relative to each other, creating a vector database.

A technology called "vector search" lets GenAI LLM user questions also be represented as vectors and mapped to the relevant source information. Proximity to or distance from one each entry in a vector database search determines their contextual relationship and the goal of vector search is to identify the top vector database entries that are most closely related to the input question's vector representation. Vector databases enable the storage and querying of vectors to occur at scale with high performance when finding closely related elements in the vector space in response to queries.

It is important to note that vector databases are not very effective at exact vector search when the number of records is large. An exact vector search can take a very long time to respond or even time out when there are millions or billions of vectors. Getting real-time search results with high-dimensional vectors necessitates "approximate nearest neighbor" (ANN) algorithms that work in conjunction with vector indexes.

The following is a set of use cases that can benefit from ***faster*** and more ***accurate*** similarity search:

- ***Finding similar documents*** - match job candidates with open positions, patient-diagnosis matching, or e-discovery
- ***Computer vision*** - facial recognition, biometric identification, or object identification
- ***Content filtering*** - individualized recommendations comparable to what comes from streaming services and online retail, or image matching to items for sale
- ***Biomedical research*** - DNA, protein, and molecular decoding/analysis/matching
- ***Natural language processing (NLP)*** - text classification, text clustering, and automated SQL generation
- ***Geographic information systems (GIS)*** - spatial analysis and map rendering
- ***Data analytics*** - anomaly detection such as fraud, and pattern recognition
- ***Industrial applications*** - quality control, predictive maintenance, and system malfunction prediction/detection/servicing/avoidance.

GenAI LLMs and AI vector search are truly revolutionary. They democratize AI in ways that were previously unobtainable by making it incredibly simple to build and use. There are now standard pre-trained embedding models, and data scientists are no longer required to build those AI applications. Anyone can now be an AI user.

However, all vector databases are not created equal. There are several fundamental and concerning problems with most vector databases. These are problems that should not be ignored because they can negatively impact the performance and accuracy of GenAI solutions.

Non-Trivial Vector Database Problems

Siloed Data

The first and most difficult vector database problem is siloed data in multiple databases or data stores across an organization. Each data source needs to have its elements chunked, embedding models run on the chunks, and indexes generated and then tuned on an ongoing basis. All of which is time consuming and costly.

Data can be siloed based in distinct separate database models and data types. Specialized databases for OLTP, OLAP, time series, JSON (document), XML (object), blockchain, AI machine learning, graphic/spatial, and now vector are often separated and have their own storage as well. Moving data from one model to another duplicates the storage. This is not a new problem. It's just a very difficult one to solve.

There are two primary methods to create a vector database separate from the source data. The first and most common, is extract the data from where it exists, copy it, do the embeddings, import the data to a vector database, and then index it. The industry term for this is extract, transform, and load or ETL. The second is to create and store the vector embeddings in the database where the source data lives.

ETLs are well known and have been around for a very long time because of a long history of siloed and specialized databases. They take time, skill, programming, manual tools, and even tools that provide extensive automation. All are time-consuming and costly. What makes ETLs worse is that the data that's being processed is not analyzable in real-time. That means by the time the data can be queried, it has become stale.

In contrast, creating, managing, and searching vector data in the source database helps eliminate most of this complexity and time lags, and several vendors are now in the process of doing this. Although a very positive development, it does not in and of itself solve the siloed data store problem.

Performance

Vector databases combined with vector search can help improve LLM accuracy and reduce their hallucinations. Doing so requires the vector search be extremely fast while providing responses in real-time.

The old axiom "time is money" is an essential fact of life. GenAI LLM real-time sub-second responses have a direct impact on user productivity, work quality, morale, turnover, time-to-action, and time-to-unique revenues and profits. Vector search with real-time query performance is not an enterprise nice-to-have, it's a must have. Vector database/vector search performance issues that have an oversized impact on response times include:

Indexing Models

Indexes are critical when searching through large sets of data. The concept of indexing pre-dates vector databases, but the methods used for vector databases are different from those used in relational databases. There are several vector indexing models to choose from, but Hierarchical Navigable Small World Graphs (HNSW), and Inverted File Index (IVF) are by far the most popular. HNSW indexes are optimized for performance and use in-memory indexes while IVF indexes focus on huge capacity via disk-based indexes.

Incremental Indexing

Another problem that has a large impact on vector database performance is the incremental indexing. The specific problem is updating these vector indexes. It's hard to do and many vector database do not keep their vector indexes up to date. For applications that continuously stream new vectors it's a problem that needs to be thoughtfully addressed. The general workaround to the incremental vector indexing problem is to rebuild the indexes periodically. This can take hours of expensive compute time. Meanwhile, vector indexes may use stale data.

Scalable Vector Search

It's imperative to understand that vector search is not the same as a vector database. Vector search is the algorithm that analyzes the vectors in a vector database. Scaling vector search algorithms may seem like an obvious requirement for enterprise database administrators, but numerous algorithms have not proven to be scalable to the level required by many enterprises.

Vector Database Data Protection

Data protection is so critical today to enterprises that they generally expect it to be built into their databases. Although most relational and document databases have long solved the data protection problem, many vector databases have not. In this world of hardware outages, software vulnerabilities, site failures, network disruptions, human errors, natural disasters, malware, and ransomware, data protection is definitely not optional.

These are hard problems to solve. Paraphrasing JFK's moon speech, that's why Oracle solved them.

Oracle Database: AI Vector Search

Oracle is the well-known and acknowledged leader in enterprise-class, mission-critical databases and has been for more than four decades. That's not an opinion. It's a fact. Consider that the uniquely integrated Oracle Database on Exadata is the database of choice for 10 of the top 10 Banks, 10 of the top 10 communications companies, 10 of the top 10 food and drug firms, 9 of the top 10 automotive corporations, 9 of the top 10 healthcare organizations, and 8 of the top 10 retail operations.

Oracle is the platinum standard for enterprise databases. And they have a history of elegantly adding new capabilities while continuing to improve performance, scalability, and data protection for every enterprise database model. They are leveraging that experience for vector databases and vector search with Oracle Database AI Vector Search. This new functionality delivers seamless integration of vector data types, indexing, and search with core Oracle Database functionality for enterprise-grade performance and reliability. Existing Oracle Database components including Real Application Clusters (RAC), sharding, partitioning, pluggable databases (PDB), multi-tenant container databases (CDB), massively parallel execution, Data Guard, security, zero data loss data protection, and integration with the extremely performant Exadata all work with AI Vector Search – on-premises or in the Oracle or Microsoft clouds.

What Oracle Database Vector Database and Vector Search Brings to the Table

Oracle's vector database model is quite comprehensive. It provides exceptional vector capabilities right from the start. Whether it's the incomparably intuitive way to implement vector searches or the highly efficient indexing, Oracle knows how to provide vector database capabilities. A deeper dive shows how Oracle solves each of the aforementioned problems.

What's in Oracle Database AI Vector Search

With AI Vector Search, Oracle delivers native support for generating vectors with its new SQL embedding function that runs inside the database removing the need to ETL data in order to create vectors or search them. It does this by importing embedding models via ONNX (Open Neural Network eXchange). Oracle's AI Vector Search works with SQL syntax and adds the capability to enable rapid searches with extremely good but not perfect accuracy. Of course, an exact accuracy search can still be conducted, but expect results to take a while.

And that's just the beginning.

Customers can specify vector search accuracy directly without fiddling with low level parameters. They can specify a target accuracy (in percentages) for similarity search in two different places:

1. Vector Index creation—for the default accuracy provided by the index.
2. Similarity search queries—to override the default index accuracy.

That accuracy is chosen based on use case. Keep in mind that as the accuracy target increases so does the response time. A law enforcement search for a person-of-interest needs a highly accurate match and it's generally okay to have a slower response time. However, online ecommerce sites will prefer a faster response time and are okay with a less accurate match. This is an Oracle vector database advantage. Many vector databases either hide all parameters or expose them at a low level. They assume the customer knows which low level parameters affects latency and response time. In reality, accuracy and response time are intuitive but low level parameters are not.

Oracle Database AI Vector Search also supports multi-vector searches. This is common when there is a requirement to have multiple inputs from a variety of sources and not from just one location. A person might have multiple photos. Each photo may have multiple chunks for different people, places, date, time,

scenery, etc. If the goal is to find sunset pictures from multiple places, what's required is for the results of a vector search that are grouped around sunsets and selects them from multiple locations. In figure 2, the red grouping may be photos over a desert, the blue ones over a body of water, and the green ones over mountains.

No current purpose-built vector database can currently do this.

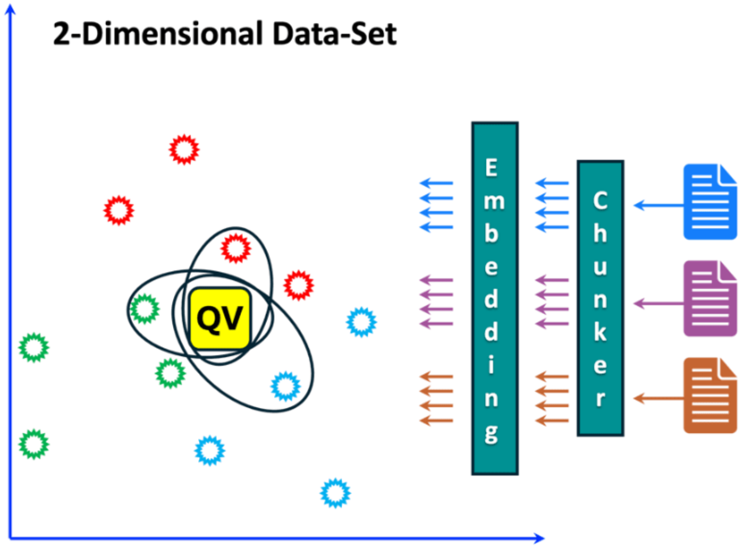


Figure 2: Two-dimensional Dataset

Converged Oracle Database

Oracle Database is a converged database. It supports the all types of workloads and database models including Relational, OLTP (transactional), OLAP (data warehouse), JSON (documents), XML (objects), time series, AI machine learning (ML), block-chain, spatial, graphics, data lake, and now vector as illustrated in figure 3.

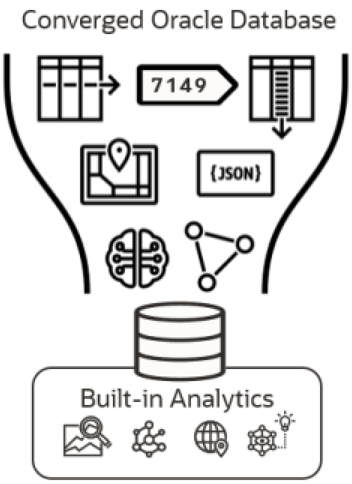


Figure 3: Converged Oracle Database

It also supports full SQL capabilities including operators and functionality such as window analytic functions, stored procedures, aggregation, and more. This makes it easy to combine vector search results with traditional SQL queries.

An exceptional aspect of Oracle AI vector search is its ability to work in conjunction with SQL joins, a capability common in relational databases because enterprise data is usually normalized. Efficiently combining AI Vector Search with traditional SQL queries and joins requires an enterprise-grade, cost-based optimizer to deliver results quickly and with low resource usage. Luckily, Oracle has a long history of delivering such optimizers and has adapted the optimizer in Oracle Database 23ai to do this while no other current purpose-built vector database can.

Vector Indexing

Oracle employs the two most popular vector indexes HNSW (graph vector index) and IVF_Flat (partition vector index). As previously discussed, these vector indexes have the best combination of accuracy and speed.

Oracle Database AI Vector Search's HNSW index is a multi-layer in-memory graph vector index that is designed for speed and accuracy (see figure 4). It's considered the "B+ tree index for vectors". The way it's constructed is the lowest graph layer and has all of the vectors. The higher layers have a subset of the layers below it. Vectors are connected based on similarity. Searches begin from the top layer. When the nearest vector is found, search continues to the layer below. The search completes in the lowest layer when the top K nearest vectors to the query vector are found.

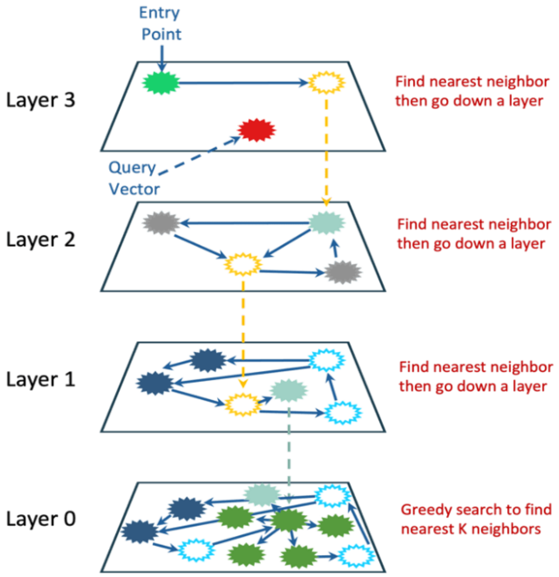

Entry Point
Layer 3
Find nearest neighbor then go down a layer
Query Vector
Layer 2
Find nearest neighbor then go down a layer
Layer 1
Find nearest neighbor then go down a layer
Layer 0
Greedy search to find nearest K neighbors

Figure 4: Multi-layer in-memory graph index

AI Vector Search also implements the IVF_Flat index, which is a partition-based index where vectors are clustered into table partitions based on similarity. It's a very efficient scale-out index, with seamless transactional support (see figure 5).

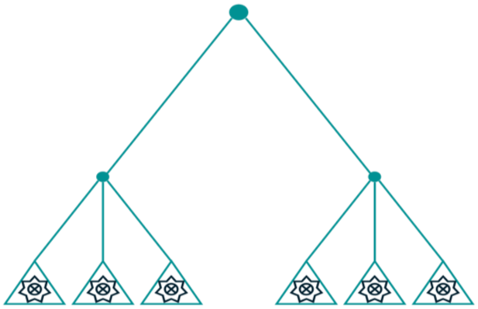


Figure 5: Partition Vector Index

Using a neighbor partition vector index search with a 2-dimensional data set would:

- Group vectors into partitions using K-means clustering algo (K = 5)
- Compute distance from query vector to each partition's centroids
- Identify the 2 nearest partitions
- Compute distance from query vector to all points in Cluster #1 & #3 to find top 5 closest matches.

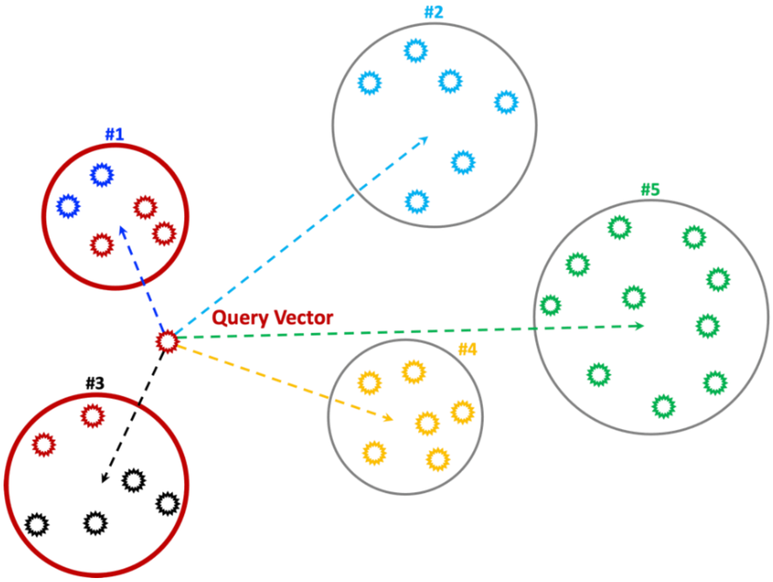


Figure 6: 2-Dimensional Data-Set

How the Oracle Database Solves the Vector Database and Vector Search Problems

Eliminates Siloed Data

As previously discussed, specialized databases require the customer to implement multiple database models, license multiple database technologies, and integrate them. Many of those models, including vector databases, are NOT the origin points for the underlying data. That means the data has to move, i.e., ETL between database models. This process is laborious, time consuming, and error prone. The result is that the receiving database model, such as the vector database, is going to be working with data that is always a little stale. This makes real-time LLM query responses extremely difficult.

Oracle solved the siloed data many years ago by converging all of the major database models into a single, multi-data, multi-model database. Oracle Database is a database engine with each database model as a feature with one stored data copy. No data migration. No ETLs.

By adding vector data types, indexes, and search to the latest Oracle Database 23ai release, the problem of creating silos of vector data is eliminated. Remember, Oracle already holds most of the world's operational enterprise data, so making vector capabilities available for that data makes perfect sense. This is an enormous boon to those thousands of enterprise Oracle Database users. Customers with multiple petabytes of data residing inside the Oracle Database, can now embed their data and perform vector searches where their data originates and resides. No ETLs, data movement, or additional storage are required. This enormous reduction in effort, time, and cost while at the same time providing real-time vectorization and vector search of their data is a massive win for customers. It accelerates practical GenAI adoption on-premises and in the cloud.

Unparalleled Performance

Oracle Database is one of the most complete databases ever developed. The fastest place to run it is on Oracle Exadata, especially Exadata platforms based on the latest AMD EPYC™ processors. Exadata is co-engineered with the Oracle Database to get the highest possible performance from the advanced capabilities provided by AMD EPYC processors. It combines ideal database hardware with database aware system software and automated management.

Figure 7: Exadata Database Service

Exadata's exclusive functionality reduce SQL read latency, a critical factor for transactional applications, to ≤ 19 µs. That's up to 50x faster than the read latency for database IOs in other cloud databases. It uses Exadata Smart Scan to offload data-intensive analytics and AI/ML queries to intelligent storage servers, eliminating the need to copy all data to database servers for processing and providing up to 2,880 GB/s of scan throughput. That's fast!

This level of performance is extremely important for Oracle Database AI Vector Search because very fast response time is essential for mission-critical applications. Applications such as real-time fraudulent financial transaction detection, real-time person of interest image matching, real-time customer risk and credit worthiness matching to customer profiles, real-time cellular network congestion detection and rerouting, and more all benefit from quick responses. In fact, anything that involved people asking questions or systems that need to operate in real time benefit from the massive performance improvements available with Oracle Exadata.

One way that Exadata accelerates the performance of the AI Vector Search ecosystem is by offloading index creation to storage servers. This allows AMD EPYC processor cores in storage servers to create updated vector indexes in the background with minimal impact on traditional database operations driven by the database servers. And, for organizations that use IVF indexes, searching is offloaded to storage servers to increase performance and lower latency.

Exadata's ideal database hardware is the result of more than 16 years of co-engineering with Oracle Database, Oracle Linux, and partners like AMD—whose EPYC processors power Exadata database and storage servers. AMD EPYC processors provide high processing density with up to 192 processor cores per database server, large memory capabilities, and very high memory bandwidth. Combined, these capabilities provide exceptional time to results for business-critical applications while delivering optimized performance for database workloads. The Exadata platform also includes dedicated RDMA networking between database servers and storage servers, high-speed network-based synchronization, massively parallel processing of analytics and, and automated prioritization of latency-sensitive IO operations that enable workloads to scale across thousands of cores in dozens of database servers.

No other database infrastructure on-premises or in the cloud matches the performance of Oracle Database on Exadata.

Automated Incremental Indexing

All IVF indexes within the Oracle Database are maintained with data manipulation language (DML). DML inserts, deletes, and updates the data in the vector database. New inserts are the equivalent to inserting a row into a portioned table. IVF indexes are constantly maintained and are consistent with base table with every DML. When used as part of the Oracle Autonomous Database, it's completely automated.

Rebuilding the vector index is not an automatic operation...yet. User intervention is currently required; however, the customer can run one the Oracle Database's *index accuracy report* API to help determine if the index is stale. If so, it's a simple matter to trigger the rebuild.

For HNSW indexes are not currently maintained nor automated with DML. Based on Oracle's history, it is likely in a future release.

Substantial Scalability

Exadata supports Oracle Real Application Clusters (RAC), which not only enable database environments to scale from 2 to 1000's of CPU cores, but also enable the database platform to provide high availability so Oracle Database environments continue to run when individual components or servers fail without an outage—a necessity for business-critical workloads.

Exadata Cloud Infrastructure uses AMD EPYC processors to provide 252 to more than 4,000 cores of processing power in database servers in public cloud, hybrid cloud, and on-premises environments. This incredible level of scalability can support thousands of concurrent transactions, complex business analytics, graph and spatial analyses, in-memory processing, *and* vector search—all at the same time. Based on previous theCUBE research, no other combinations of processing, networking, and storage comes anywhere near Exadata's low latencies, high IOPS, and high throughput for Oracle Database services.

Each Exadata system in the Oracle Cloud supports from 190 TB to 4.2 PB of usable storage capacity. It can be used for any mix of the Oracle Database models and data types including vector databases.[2]

As previously noted, the Exadata's smart storage servers take full advantage of their CPU cores[3] to locally run SQL queries for analytics, ML model building, and now critical AI Vector Search capabilities such as vector index creation. When combined with Oracle Database Hybrid Columnar Compression, each Exadata platform in the cloud can support up to 30 PB of data warehouses.

Oracle Database AI Vector Search takes full advantage of everything Exadata has to offer. It does so in a highly sophisticated, flexible, and performant way. It leverages Exadata's intelligent storage servers and the high core counts provided by AMD EPYC processors to empower customers' use of their corporate data stored in Oracle databases.

[2] Non-cloud Exadatas can scale to 14 racks whereas Exadata Cloud@Customer can scale to 6 racks.

[3] Exadata Intelligent Storage Server cores do not count against Oracle Database licensing on-premises or in the cloud.

Built-in Vector Database Data Protection

Because the vector database and vector search are now features of the Oracle Database, they automatically get all of the HA, DR, and business continuity that's built-in and integrated into it. These capabilities are further improved and accelerated by running Oracle Databases, including Autonomous Database on Exadata platforms. The level of HA, DR, and business continuity is unsurpassed by any other cloud or database platform currently on the market.

It starts with Oracle RAC, which allows customers to run their databases non-stop while scaling and patching or when faced with hardware outages. RAC distributes a customer's Oracle Database across an Exadata VM cluster and allows the number of cores per server and the number of servers to scale up or down online. This means that customers using Oracle Database on Exadata platforms can scale consumption to meet changing workloads, can patch and upgrade software, and can have database servers and storage serviced without incurring application-level downtime. RAC is an essential part of meeting customer requirements for high availability and scalability for their business-critical applications.

That's just the beginning. To provide even greater availability, there's Oracle Active Data Guard within or between cloud regions, customer premises, or between customer premises and cloud regions. Set up one location as a standby, or run workloads in one location while using the secondary location for DR.

With the combination of Oracle Database and Exadata high availability designs, intelligent software, and the deeply integrated HA, DR, and business continuity, vector database data protection obstacles are eliminated.

A Word About Costs

There is a market perception that the Oracle Database and Exadata are very expensive. Past theCUBE research has proven that perception to be grossly incorrect. TCO comparisons both in the cloud and on-premises have shown that Oracle's costs are significantly lower than their competition when performance is equalized as much as it can be. That means scaling down Oracle Exadata configurations to provide a reasonable comparison. When looking at TCO/performance, Oracle literally laps their competition many times over.

Conclusion

Oracle Database AI Vector Search is another Oracle game changer. This latest release gets to take advantage of all the capabilities in the Oracle Database and Exadata, while solving the hard vector database and vector search problems others have not or cannot solve. With a single Oracle Database to manage and Exadata's high-performance, customers can start their AI journey without technology debt.

For More Information

Go to:

Oracle Database 23ai with AI Vector Search