



MongoDB Search & Vector Search for AI: Practical Applications on Enterprise Server

**MongoDB Day
Frankfurt
September, 2025**



Bart Polot
Solutions Architect
FSI & Telco



Agenda

The Evolution of the MongoDB Data Platform

MongoDB Search & Vector Search Public Preview
with MongoDB Enterprise Server

Getting Started and Code Samples

Architecture Diagram and Demo

Recap

The evolution of MongoDB



All roadmap promises
delivered

Pre-2018

Doc-Level Concurrency
Retryable Writes
Changestreams
Sharded Secondary reads
RAFT / Fast Failover
\$lookup
Ops Manager
Query Advisor
Charts
Compression
≤50 replicas
Aggregation ++
Encrypted and In-Memory storage engines
BI Connector ++
Auto Storage Scaling
Compass
Charts
Views Graph Processing Zones
++Aggregation ++ Auto-balancing
++ Linearizable Reads Decimal
Intra-cluster Compression Log
Redaction Spark Connector ++ BI
Connector ++

**WiredTiger (Acquisition +
Integration)**

MongoDB Atlas

2018

4.0

Replica set transactions
Atlas global clusters
40% Faster Shard Migrations
Atlas HIPAA
Atlas LDAP
Atlas Audit
Atlas Enc. Storage Engine
Atlas Backup Snapshots
Type Conversions
Snapshot Reads
Non-Blocking Sec. Reads
Compass Agg Pipeline Builder
Compass Export to Code
Free Monitoring Cloud Service
Ops Manager K8s Beta

**ACID
(Transactions)**

2019

4.2

Distributed Transactions
Client-Side Field Level Encryption
Materialized Views
Wildcard Indexes
Global PIT Reads
Large Transactions
Mutable Shard Key Values
Atlas Data Lake (Beta)
Atlas Auto Scaling (Beta)
Atlas Search (Beta)
Multi-CAs
Expressive Updates
Apache Kafka Connector
MongoDB Charts GA
Retryable Reads & Writes
New Index Builds
10x Faster stepDown
Storage Node Watchdog
Zstandard Compression

Atlas Search

**Distributed
Transactions**

2020

4.4

Union
Custom Agg Expressions
Refinable Shard Keys
Compound Hashed Shard Keys
Mirrored Reads
Hedged Reads
Resumable Initial Sync
Time-Based Oplag Retention
Simultaneous Indexing
Hidden Indexes
Streaming Replication
Global Read/Write Concerns
Rust & Swift Drivers GA
TLS 1.3 & Faster Client Auth
OCSP Stapling
Kerberos Utility
Atlas Online Archive
Auto-Scaling
Schema Recommendations
AWS IAM Auth & Atlas x509
Federated Queries
Multi-cloud clusters

Inter-Cloud

2021-2022

5.x

Time Series collections
Clustered indexes
Window functions
Live resharding
Client-Side FLE KMIP & cloud KMS
Atlas Serverless (preview)
Atlas Search fast facets, function
scores, synonyms
Long running snapshot reads
Sharded \$lookup & \$graphlookup
Majority write concern default
4x faster initial sync
Schema validation diagnostics
New MongoDB Shell GA
Resumable index builds
Rewritten C# LINQ provider and
.NET Analyzer
PyMongoArrow API
New accumulator operators
Charts on Data Lake
x509 certificate rotation
Auditing ++
Atlas K8S controller
Ops Manager migration wizard

Time Series

Rapid Releases

2022-2023

6.x

Queryable encryption (preview)
Atlas Serverless GA
Cluster-to-cluster sync
Independent analytic nodes scaling
Flexible sync GA
Asymmetric sync preview
Kotlin and Flutter SDKs for Realm
\$lookup perf improvements
Atlas CLI GA
Atlas Data API GA
Secondary index on measurements
for time series
Time series read perf +
Atlas Data Lake with fully managed
storage
Atlas Data Federation
Atlas Charts dashboard embedding
Atlas SQL interface
Encrypted audit log
Change streams improvements
Relational Migrator (preview)

**Relational
Migrator**

**Queryable
Encryption**

2023-2024

7.x

Vector Search (Preview)
Queryable Encryption GA
Stream Processing (Preview)
Relational Migrator GA
Compound Wildcard Index
Improved Query Execution
Approximate percentiles
Time series performance and
scalability updates
Shard key analysis
updateOne w/out shard key
Shard chunk auto-merger
AWS, Azure, GCP ID
management
Make X.509 Parameters
configurable for Intra-node
Auth and Client-to-node Auth
OpenSSL 3.0 support in Server
C2C sync updates
Live Migration supports 6.0+
migrations
PyMongo support

Vector Search

**Stream
Processing**

2024-2025

8.x

Vector Search GA
Exact Nearest Neighbor VS
AI Frameworks support
Stream Processing GA
Relational Migrator GA
Compound Wildcard Index
Improved Query Execution
Approximate percentiles
Time series performance and
scalability updates
Shard key analysis
updateOne w/out shard key
Shard chunk auto-merger
AWS, Azure, GCP ID
management
Make X.509 Parameters
configurable for Intra-node
Auth and Client-to-node Auth
OpenSSL 3.0 support in Server
C2C sync updates
Live Migration supports 6.0+
migrations
PyMongo support

**GenAI Developer
Experiences**

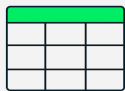
MongoDB Data Platform



MongoDB Database



JSON



Tabular



Key-Value



Text



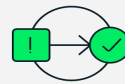
Geospatial



Graph



Time Series



Events



Vector &
Search



MongoDB®



MongoDB Search & Vector Search **Public Preview** with MongoDB Enterprise Server



“

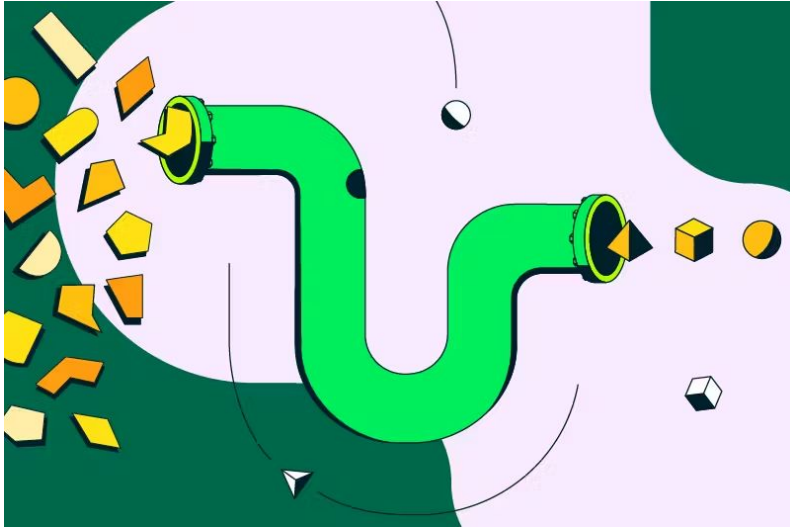
“Our enterprise customers have told us they want to harness the power of generative AI on their own proprietary data, but the architectural complexity of bolting on separate search and vector databases has been a major barrier, by bringing our proven Atlas Search capabilities to our self-managed customers, we are giving them the tools to begin innovating faster on their own terms.”



JIM SCHARF

Chief Technology Officer
MongoDB

Key Pain Points Addressed



- Complexity of maintaining separate search engines
- Inability to use Atlas Search due to regulatory requirements
- Limited search capabilities in local development
- Fragmented development experience between local and production environments
- Risk of investing in architecture that might not align with final product



MongoDB Search and Vector Search with Enterprise Server will enable you to

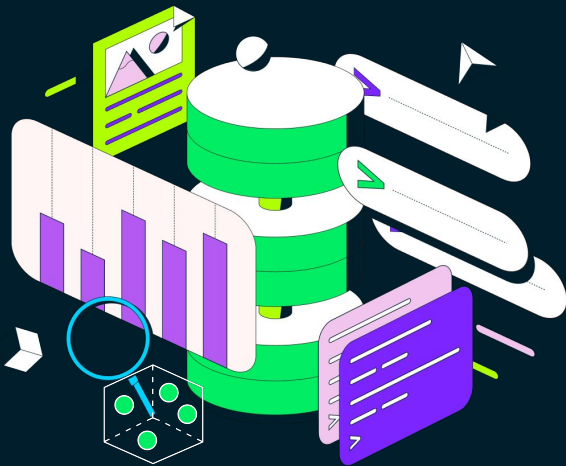


- Build next-generation AI applications on **your own infrastructure**.
- **Eliminate the architectural complexity** and operational overhead of separate systems.
- **Accelerate** development and **reduce costs**.
- Deploy the same application code **across multiple environments**.

With MongoDB deliver faster, more innovative, and engaging user experiences.

Enterprise
Server

Search & Vector Search on self-managed clusters



Benefits

Run your search workloads anywhere

You no longer have to run on MongoDB Atlas to get all the benefits of MongoDB Search and Vector Search. Build AI powered applications on-prem with ease.

Avoid the synchronization tax

With Search and Vector Search, data is automatically and dynamically synced from the database to Search and Vector Search indexes.

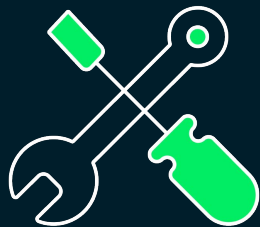
Superior Developer Experience

Use a single, unified API across database and search operations. Create a search index with a few clicks or a single API call.

Getting Started

1

Prerequisites



A running MongoDB Enterprise Server cluster.

Version 8.0.10+ (for MCK 1.4) or version 8.2+ (for MCK 1.5+)



A Kubernetes environment.



The MongoDB Controllers for Kubernetes (MCK) Operator installed in the Kubernetes cluster.





2

Deploy Search Nodes on Kubernetes

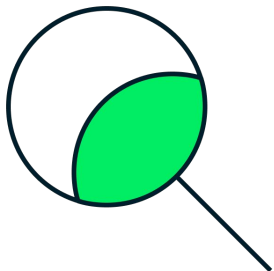


Creating a **Search custom resource (CR)** in Kubernetes. This CR tells the MCK Operator to deploy the required search nodes and connect them to your existing MongoDB cluster.

You can target MongoDB Enterprise Server instances deployed **outside or inside** Kubernetes.

3

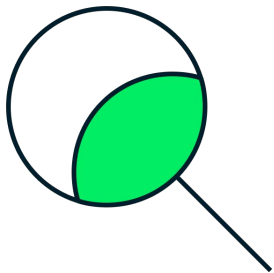
Create search indexes



```
db.collection.createSearchIndex({  
  name: "full-text-index",  
  definition: {  
    mappings: { "dynamic": true }  
  }  
});
```

3

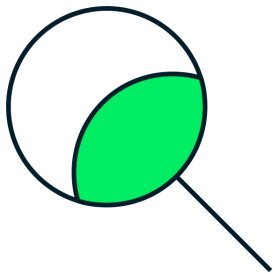
Create vector search indexes



```
db.collection.createSearchIndex({
  name: "vector-index",
  definition: {
    mappings: {
      fields: [
        {
          path      : "description",
          type       : "vector",
          numDimensions : 1536,
          similarity  : "cosine",
          quantization : "none"
        }
      ]
    }
  }
});
```

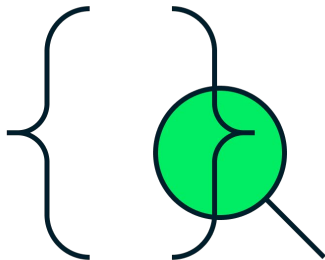
3

Create vector search indexes with (pre)filters



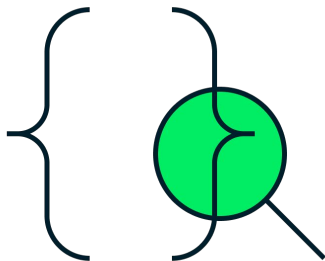
```
db.collection.createSearchIndex({
  name: "vector-index",
  definition: {
    mappings: {
      fields: [
        {
          path      : "description",
          type      : "vector",
          numDimensions : 1536,
          similarity  : "cosine",
          quantization : "none"
        }, {
          path      : "tenant_id",
          type      : "filter"
        }
      ]
    }
  }
});
```

Query your data



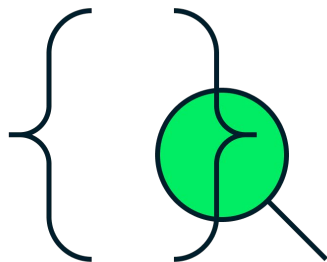
```
db.collection.aggregate([  
  { $search: {  
    index: "full-text-index",  
    text: {  
      query: "sci-fi",  
      path: "genres"  
    }  
  }  
}]);
```

Query your data



```
db.collection.aggregate([  
  { $vectorSearch: {  
    index: "vector-index",  
    queryVector: [0.1, -0.4, 0.23, ...],  
    path: "description",  
    numCandidates: 100,  
    limit: 10 }  
  }  
]);
```

Query your data

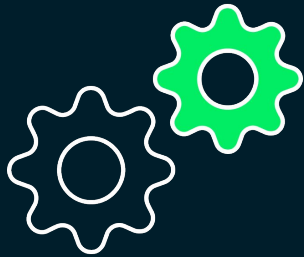


```
db.collection.aggregate([
  { $vectorSearch: {
    index: "vector-index",
    queryVector: [0.1, -0.4, 0.23, ...],
    filter: { tenant_id: 42 },
    path: "description",
    numCandidates: 100,
    limit: 10 }
  }
]);
```



5

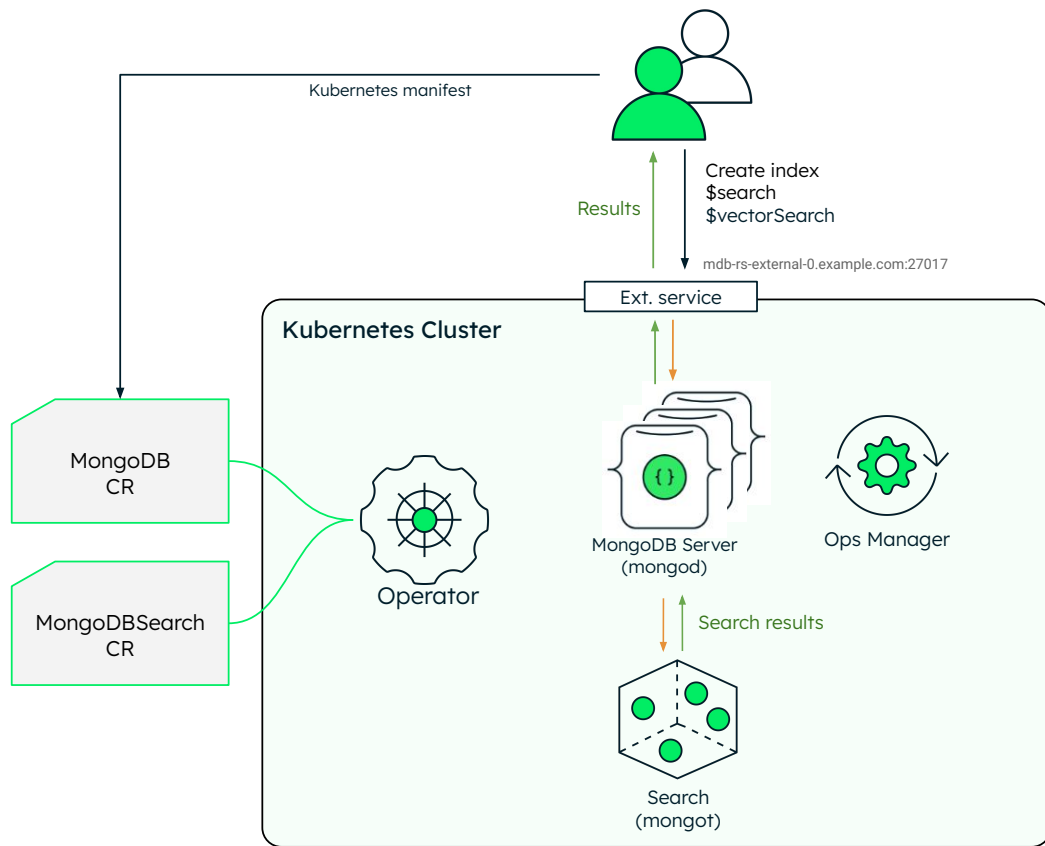
Monitor Search Nodes



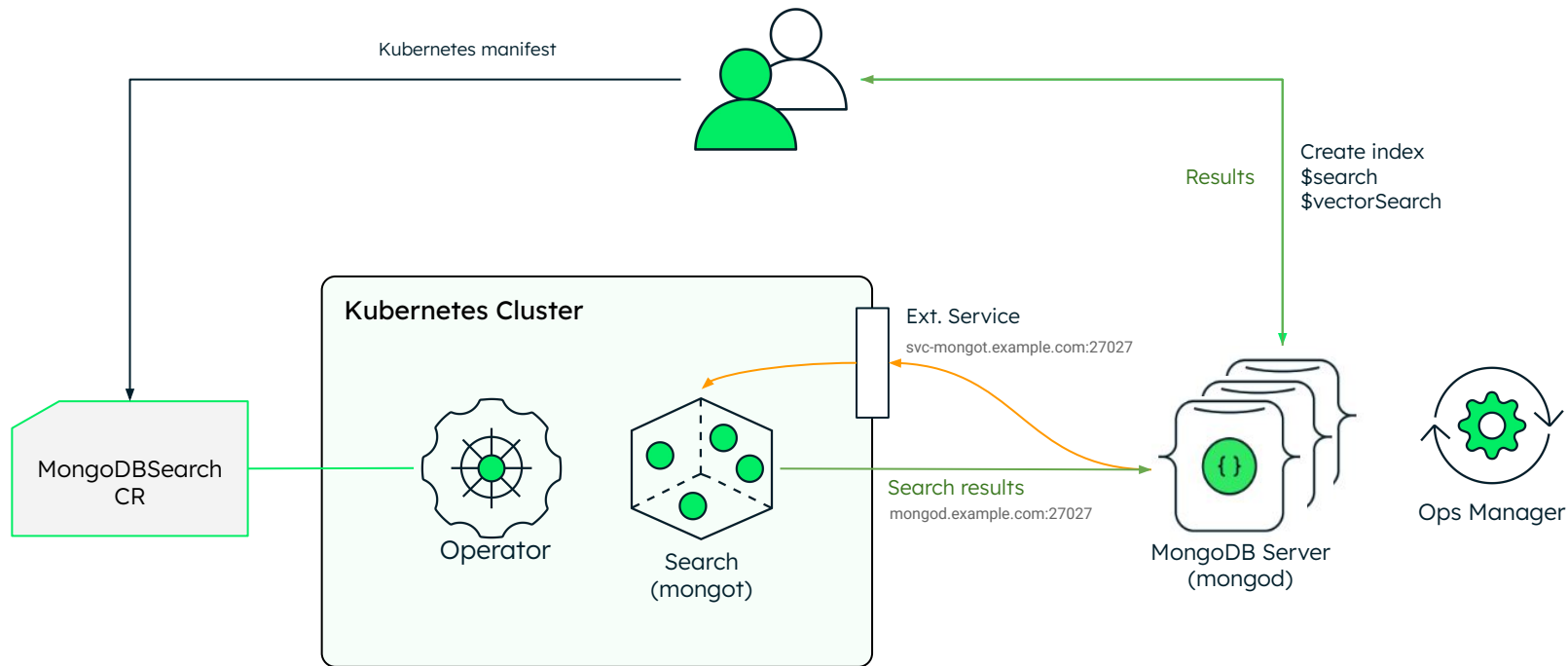
You can **monitor** the health and status of the search node pods using **standard Kubernetes tooling**.

Architecture Diagram

MongoDB Server inside of Kubernetes



MongoDB Server outside of Kubernetes



Demo Time

⚙️ .env

backend > advanced-search-ms > ⚙️ .env

```
1  # MongoDB Atlas settings
2
3  #ATLAS CONNECTION STRING
4  MONGODB_URI=mongodb+srv://ist-retail-domain:UWwD8vud8wR6t@ist-shared.n0kts.mongodb.net/?retryWrites=true&w=majority
5
6  # ENTERPRISE SERVER CONNECTION STRING
7  #MONGODB_URI=mongodb://mdb-001-001-001-001-001-001-001-001@mdb-rs-0.mdb-rs-svc.mongodb.svc.cluster.local:27017,mdb-rs-1.mdb-rs-2.mdb-rs-svc.mongodb.svc.cluster.local:27017,mdb-rs-3.mdb-rs-svc.mongodb.svc.cluster.local:27017/?retryWrites=true&w=majority
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL

> ▼ **TERMINAL**

```
🔌 2025-08-28 21:18:45,465 INFO advanced-search-ms.domain - [DOMAIN] Mapping MongoDB document to Product domain model
🔌 2025-08-28 21:18:45,465 INFO advanced-search-ms.domain - [DOMAIN] Mapping MongoDB document to Product domain model
```



.env

backend > advanced-search-ms > .env

```
1  # MongoDB Atlas settings
2
3  #ATLAS CONNECTION STRING
4  #MONGODB_URI=mongodb+srv://test-research-domain:QWERTY@mongodb-01@ist-shared.n0kts.mongodb.net/?retryWrites=true&w=majority
5
6  # ENTERPRISE SERVER CONNECTION STRING
7  MONGODB_URI=mongodb://mongodb-admin:admin-user-password-CHANGEME@mdb-rs-0.mdb-rs-svc.mongodb.svc.cluster.local:27017,mdb-rs-1.mdb-rs-2.mdb-rs-svc.mongodb.svc.cluster.local:27017
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
```

PROBLEMS 10

OUTPUT

DEBUG CONSOLE

TERMINAL

> ▼ TERMINAL



```
2025-08-28 21:18:45,465 INFO advanced-search-ms.domain - [DOMAIN] Mapping MongoDB document to Product domain model
2025-08-28 21:18:45,465 INFO advanced-search-ms.domain - [DOMAIN] Mapping MongoDB document to Product domain model
```

.env

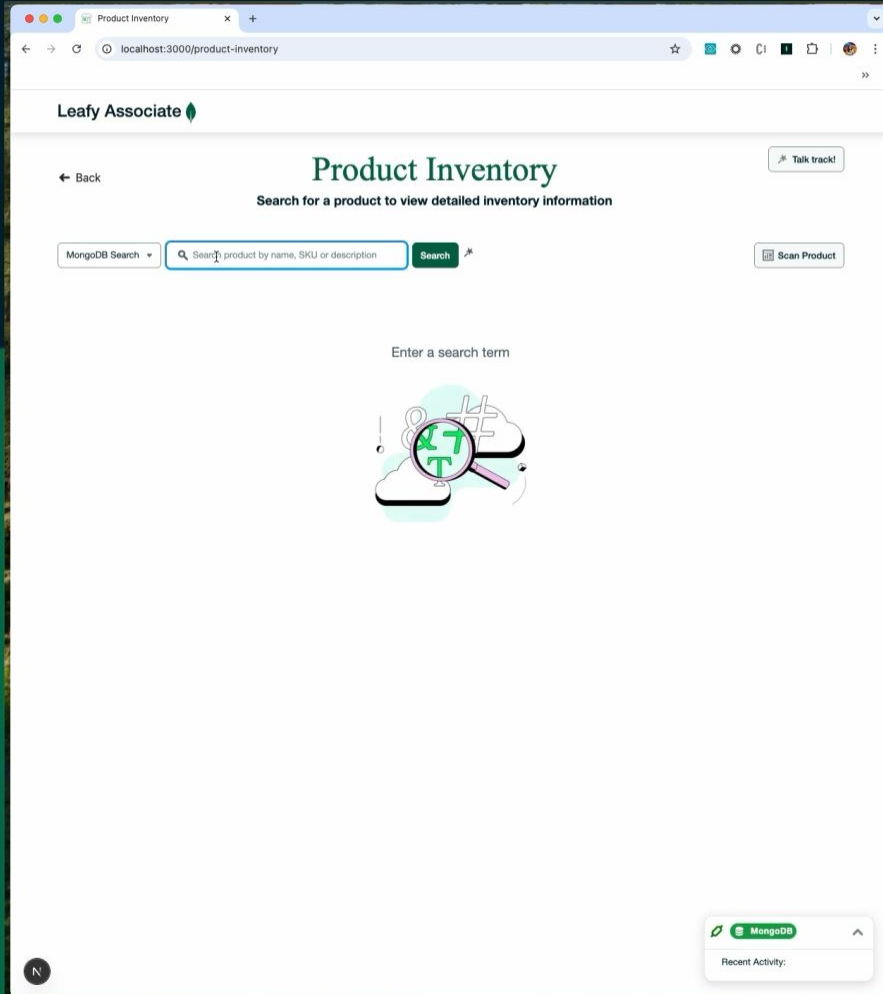
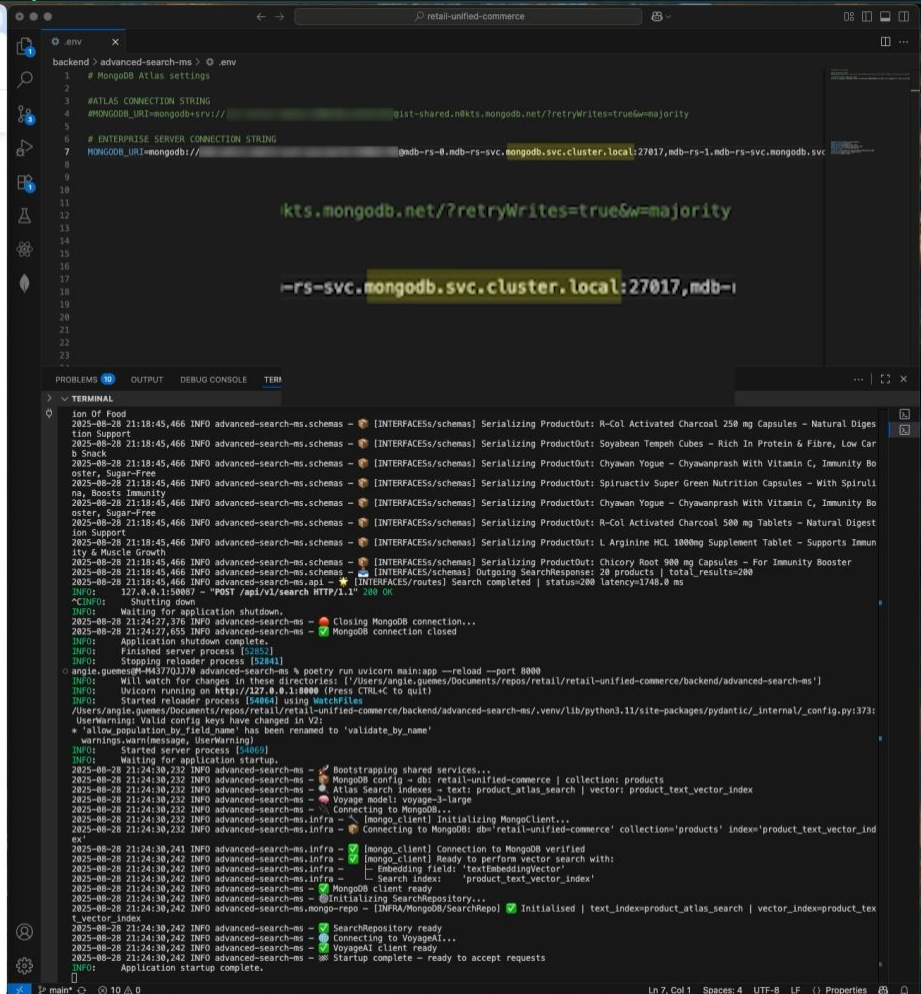
backend > advanced-search-ms > .env

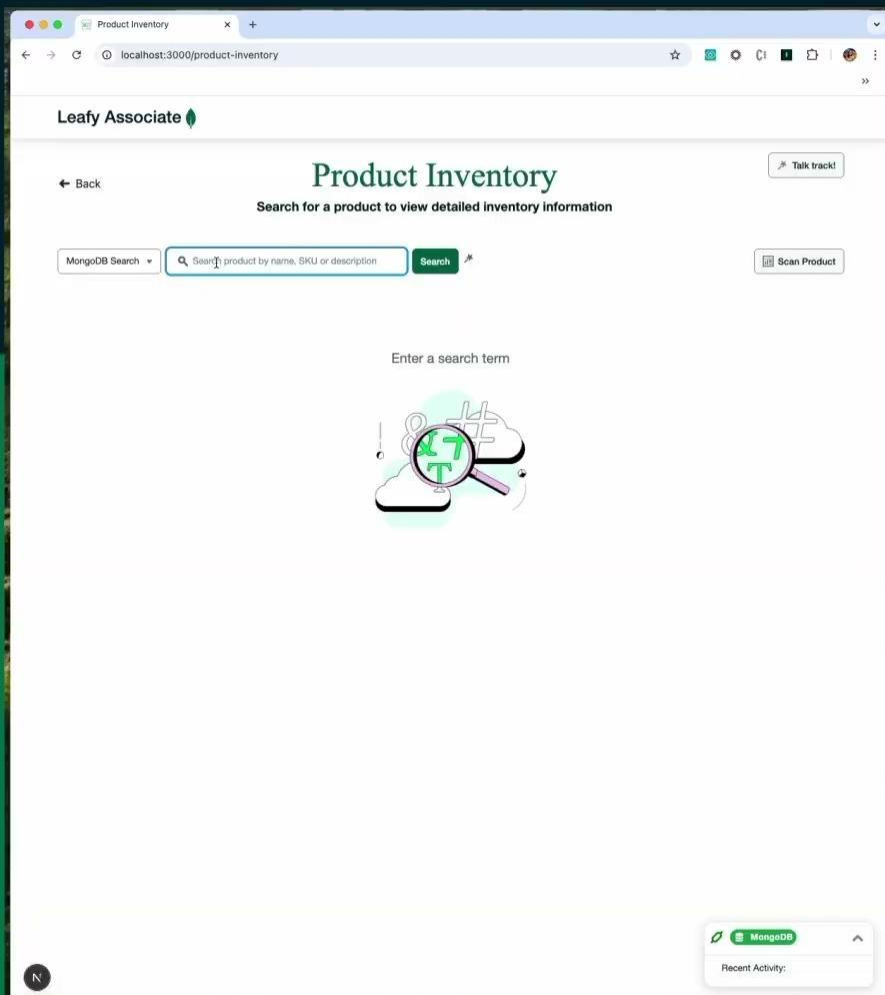
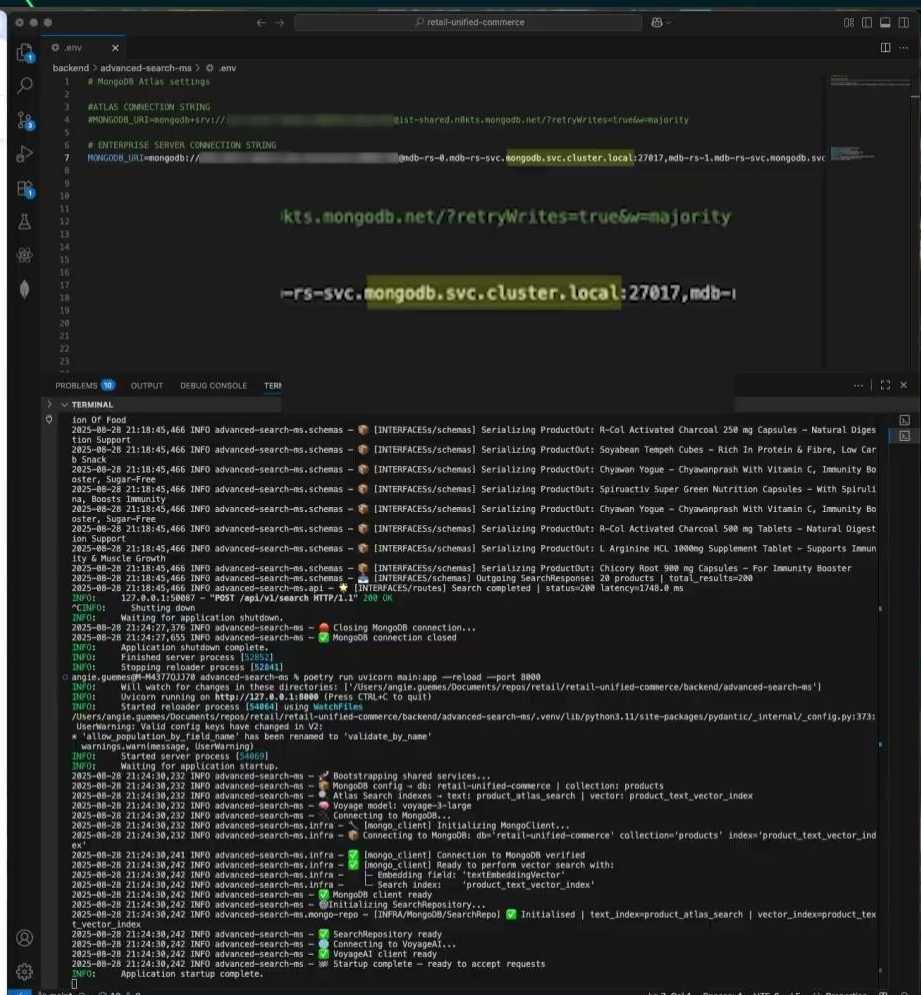
```
1  # MongoDB Atlas settings
2
3  #ATLAS CONNECTION STRING
4  MONGODB_URI=mongodb+srv://root-replset-domain:root-replset-domain@ist-shared.n0kts.mongodb.net/?retryWrites=true&w=majority
5
6  # ENTERPRISE SERVER CONNECTION STRING
7  #MONGODB_URI=mongodb://mongodb-replset-domain:root-replset-domain@mongodb-rs-0.mdb-rs-svc.mongodb.svc.cluster.local:27017,mdb-rs-1.mdb-
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL

> **TERMINAL**

```
2025-08-28 21:18:45,465 INFO advanced-search-ms.domain - [DOMAIN] Mapping MongoDB document to Product domain model
2025-08-28 21:18:45,465 INFO advanced-search-ms.domain - [DOMAIN] Mapping MongoDB document to Product domain model
```





Setup

Last login: Fri Sep 5 08:26:55 on ttys006

angie.guemes@M-M4377QJJ70 ~ % kubectl apply --context "\${K8S_CTX}" -n "\${MDB_NS}" -f - <<EOF

apiVersion: mongodb.com/v1

kind: MongoDBSearch

metadata:

name: \${MDB_RESOURCE_NAME}

spec:

resourceRequirements:

limits:

cpu: "3"

memory: 5Gi

requests:

cpu: "2"

memory: 3Gi

EOF



! YAML ×



```
1  apiVersion: mongodb.com/v1
2  kind: MongoDBSearch
3  metadata:
4    name: mdb-rs
5    namespace: mongodb
6  spec:
7    source:
8      # point to a MongoDB Enterprise server deployed on the same cluster
9      mongodbResourceRef:
10       name: mdb-rs
11      # or a MongoDB Enterprise server running outside Kubernetes
12      # external:
13      #   hostAndPorts:
14      #     - replicaset-node-first.acme.com:27017
15      #     - replicaset-node-second.acme.com:27017
16      #     - replicaset-node-third.acme.com:27017
17  resourceRequirements:
18    limits:
19      cpu: "3"
20      memory: 5Gi
21    requests:
22      cpu: "2"
23      memory: 3Gi
```





! YAML x



```
1  apiVersion: mongodb.com/v1
2  kind: MongoDBSearch
3  metadata:
4    name: mdb-rs
5    namespace: mongodb
6  spec:
7    source:
8      # point to a MongoDB Enterprise server deployed on the same cluster
9      mongodbResourceRef:
10       name: mdb-rs
11     # or a MongoDB Enterprise server running outside Kubernetes
12     # external:
13     #   hostAndPorts:
14     #     - replicaset-node-first.acme.com:27017
15     #     - replicaset-node-second.acme.com:27017
16     #     - replicaset-node-third.acme.com:27017
17   resourceRequirements:
18     limits:
19       cpu: "3"
20       memory: 5Gi
21     requests:
22       cpu: "2"
23       memory: 3Gi
```





! YAML x



```
1  apiVersion: mongodb.com/v1
2  kind: MongoDBSearch
3  metadata:
4    name: mdb-rs
5    namespace: mongodb
6  spec:
7    source:
8      # point to a MongoDB Enterprise server deployed on the same cluster
9      mongodbResourceRef:
10       name: mdb-rs
11      # or a MongoDB Enterprise server running outside Kubernetes
12      # external:
13      #   hostAndPorts:
14      #     - replicaset-node-first.acme.com:27017
15      #     - replicaset-node-second.acme.com:27017
16      #     - replicaset-node-third.acme.com:27017
17  resourceRequirements:
18    limits:
19      cpu: "3"
20      memory: 5Gi
21    requests:
22      cpu: "2"
23      memory: 3Gi
```





```
angie.guemes@M-M4377QJJ70 ~ %  
angie.guemes@M-M4377QJJ70 ~ %  
angie.guemes@M-M4377QJJ70 ~ %
```

I



```
angie.guemes@M-M4377QJJ70 ~ %  
angie.guemes@M-M4377QJJ70 ~ %  
angie.guemes@M-M4377QJJ70 ~ % echo; echo "MongoDBCommunity resource"  
kubectl --context "${K8S_CTX}" -n "${MDB_NS}" get mdb/mdb-rs  
echo; echo "MongoDBSearch resource"  
kubectl --context "${K8S_CTX}" -n "${MDB_NS}" get mdb/mdb-rs  
echo; echo "Pods running in cluster ${K8S_CTX}"  
kubectl --context "${K8S_CTX}" -n "${MDB_NS}" get pods
```

MongoDBCommunity resource

NAME	PHASE	VERSION	TYPE	AGE
mdb-rs	Running	8.0.10	ReplicaSet	17d

MongoDBSearch resource

NAME	PHASE	AGE
mdb-rs	Running	17d

Pods running in cluster

NAME	READY	STATUS	RESTARTS	AGE
mdb-rs-0	1/1	Running	0	37h
mdb-rs-1	1/1	Running	0	27h
mdb-rs-2	1/1	Running	0	27h
mdb-rs-search-0	1/1	Running	0	37h
mongodb-kubernetes-operator-5ddfccf4d9-5kgph	1/1	Running	0	37h

```
angie.guemes@M-M4377QJJ70 ~ %
```

MongoDB Controllers for Kubernetes Operator

Copy page

Rate this page



IMPORTANT

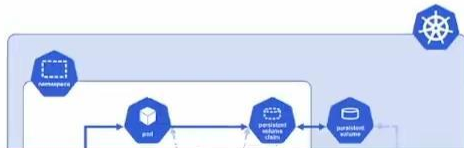
Kubernetes Operator deploys MongoDB Enterprise, [Ops Manager](#), and MongoDB Community to Kubernetes. Due to the breadth of configuration options available for MongoDB Enterprise and [Ops Manager](#), this guide focuses on these deployment options.

To deploy MongoDB Community in Kubernetes, see [the documentation on GitHub](#).

The MongoDB Controllers for Kubernetes Operator translates the human knowledge of creating a MongoDB instance into a scalable, repeatable, and standardized method. Kubernetes needs help creating and managing stateful applications like databases. It needs to configure the network, persist storage, and dedicate computing capacity without additional human effort on each container.

The Kubernetes Operator manages the typical lifecycle events for a MongoDB cluster: provisioning storage and computing power, configuring network connections, setting up users, and changing these settings as needed. It accomplishes this using the Kubernetes API and tools.

You provide the MongoDB Controllers for Kubernetes Operator with the specifications for your MongoDB cluster. The MongoDB Controllers for Kubernetes Operator uses this information to specify to Kubernetes how to configure that cluster including provisioning storage, setting up the network connections, and configuring other resources.



MongoDB Controllers for Kull

Atlas Vector Search Overview

mongodb.com/docs/atlas/atlas-vector-search/vector-search-overview/

MongoDB Docs

GET STARTED

DEVELOPMENT

MANAGEMENT

CLIENT LIBRARIES

TOOLS

ATLAS ARCHITECTURE CENTER

DATABASE MANUAL

8.0 (Current)

Overview

Documents

Databases & Collections

Client Libraries

Connect to Clusters

Database Users

CRUD Operations

Indexes

Data Modeling

Aggregation Operations

Atlas Search

Atlas Vector Search

Quick Start

Create Embeddings

Create and Manage Indexes

Create and Run Queries

Hybrid Search

Use Compatible Views

Transform Documents & Filter Collections

Vector Quantization

Retrieval-Augmented Generation (RAG)

Playground Chatbot Demo Builder

Build AI Agents

Search MongoDB Docs

Ask MongoDB AI

that contain vector embeddings at query-time. In your Atlas Vector Search index definition, you index the fields in your collection that contain your embeddings to enable vector search against those fields. Atlas Vector Search supports embeddings that are less than and equal to 8192 dimensions in length.

You can also pre-filter your data by indexing additional fields in your collection that you want to run your Atlas Vector Search queries against. You can filter on boolean, date, objectId, numeric, string, and UUID values, including arrays of these types. Filtering your data narrows the scope of your search and ensures that certain vector embeddings aren't considered for comparison.

To learn how to index fields for Atlas Vector Search, see [How to Index Fields for Vector Search](#).

Atlas Vector Search Queries

Atlas Vector Search supports approximate nearest neighbor (ANN) search with the [Hierarchical Navigable Small Worlds](#) algorithm and exact nearest neighbor (ENN) search.

To find the most similar vectors, Atlas Vector Search performs ANN search without scanning every vector embedding and ENN search exhaustively on all the indexed vector embeddings. To learn more, see [vectorSearch Definition](#).

Atlas Vector Search queries consist of [aggregation pipeline stages](#) where the [\\$vectorSearch](#) stage is the first stage in the pipeline. The process for a basic Atlas Vector Search query is as follows:

1. You select either ANN or ENN search and specify the [query vector](#), which is the vector embedding that represents your search query.
2. Atlas Vector Search finds vector embeddings in your data that are closest to the query vector.
3. Atlas Vector Search returns the documents that contain the most similar vectors.

To customize your vector search query, you can pre-filter your data on fields that you've indexed by using an [MQL](#) match expression with supported [query](#) or [aggregation operators](#), or you can add additional [aggregation stages](#) to further process and organize your results.

To learn how to create and run Atlas Vector Search queries, see [Run Vector Search Queries](#).

Next Steps

For a hands-on experience creating Atlas Vector Search indexes and running Atlas Vector Search queries against sample data, try the [Atlas Vector Search Course on MongoDB University](#) and the tutorials in the

Earn a Skill Badge

Master "Vector Search Fundamentals" for free!

[Learn more](#)

Rate this page

On this page

What is Vector Search?

Use Cases

AI Integrations

Key Concepts

Atlas Vector Search Indexes

Atlas Vector Search Queries

Next Steps

Use Cases

Order details

[Talk track!](#)

Summary

Date: Aug 25, 2025 at 2:37:17 AM**ID:** 68ac20bd9e2021484ad1f62**Status:** DELIVERED**Type:** [BUY ONLINE, GET DELIVERY AT HOME](#)**Address:** 226 W 108th St, New York City NY 10025, USA**Total:** \$183**Shipping:** \$0**Receipt:** [See details](#)

Status

**In process:** Aug 25, 2025 at 2:37:17 AM**Ready for delivery:** Aug 25, 2025 at 2:37:27 AM**Picked up from warehouse:** Aug 25, 2025 at 2:37:37 AM**In Transit:** Aug 25, 2025 at 2:37:47 AM**Delivered:** Aug 25, 2025 at 2:37:57 AM

Products

**Girls' Petite Sterling Silver Birthstone Open Heart Stud Earrings and 16" Pendant Necklace Jewelry Set****\$116****Amount:** 1**Flavia Women's Clutch (Wine)****\$67****Amount:** 1

Chat with our AI Assistant

[Get it](#)

Digital receipt

Receipt Document Digital receipts Behind the scenes Why MongoDB

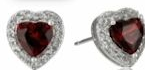
Loyalty

Points earned	+50
Tier	Gold

[Download receipt as PDF](#)

Based on this order you might also like

★ 0.90979



Amazon Collection
Sterling Silver
Birthstone Halo Heart
Stud Earrings, Garnet
Amazon Collection
\$77

★ 0.86807



Sterling Silver Rose
Gold Plated Polished
Bead Post 12mm
Stud Earrings
Amazon Collection
\$53

★ 0.86789



Platinum-Plated
Sterling Silver and
Swarovski Zirconia
Solitaire Pendant
Necklace, 16 inch
Amazon Collection
\$49

★ 0.86438



Silver Plated
Swarovski Emerald
Crystal Birthstone
Pendant Necklace
(May)
Amazon Collection
\$47

★ 0.86498



Yellow Gold Plated
Sterling Silver Stud
Earrings set with
Round Cut Swarovski
Zirconia (4 ctww)
Amazon Collection
\$107

★ 0.86469



10k Gold Made with
Swarovski Birthstone
November Stud
Earrings
Amazon Collection
\$82

[← Back to Home](#)

Transaction Simulator Powered by MongoDB

Select Customer

Customer

Stephen Burns

Profile Summary

Account	Risk Score	Avg. Transaction
PHTF5894P498516337	15.73	\$58.33

{ MongoDB Document }

Scenario Selection

- ☒ Normal Transaction
- ☐ Unusual Amount
- ☐ Unusual Location
- ☐ New Device
- ☐ Multiple Red Flags

Scenario Description

A typical transaction within customer's normal patterns and behaviors.

Transaction Details

Transaction Type

Purchase

Payment Method

Credit Card

Amount (USD)

\$ 58

Avg: \$58.33 (+\$38.8)

Merchant Category

Healthcare

Common: healthcare, restaurant, gas

Location

Using Customer's Common Location

Williamsborough, Louisiana, NL

Device

Select Device

desktop - macOS (Firefox)

Device Details

ID: f67f29f1-24a9-47f1-ab22-1aca3be1afa4

Type: desktop

OS: macOS

Browser: Firefox

[Evaluate Transaction](#)

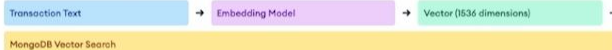


Vector Search Fraud Assessment

Vector Search Fraud Analysis

Using MongoDB Vector Search to analyze semantically similar transactions for fraud detection

Vector Embedding Process



Vector Search Risk Score Calculation SCORE: 50

Click to see the detailed mathematical breakdown of how the vector search risk score was calculated

Vector - Matched Transactions: (Showing top 5 of 15 vector matches)

\$681.43 purchase at healthcare

Vector Similarity: 88.2%

Date: 5/13/2025, 11:21:09 AM Risk Level: medium Risk Score: 57.525148591201216 Payment Method: Credit Card

Risk Flags:

Rare Transaction Time Velocity Alert

\$808.51 purchase at healthcare

Vector Similarity: 87.7%

Date: 5/8/2025, 1:16:02 AM Risk Level: medium Risk Score: 64.42560220727748 Payment Method: Credit Card

Risk Flags:

Unusual Amount Unexpected Location Velocity Alert

\$781.03 purchase at healthcare

Vector Similarity: 87.7%

Date: 5/6/2025, 1:40:43 PM Risk Level: medium Risk Score: 49.29527192518717 Payment Method: Credit Card

Risk Flags:

Unexpected Location Unusual Amount

\$247.79 purchase at healthcare

Vector Similarity: 87.4%

Date: 2/16/2025, 1:50:15 PM Risk Level: medium Risk Score: 37.307993062491186 Payment Method: Credit Card

Risk Flags:

Rare Transaction Time

\$82.55 purchase at healthcare

Vector Similarity: 86.8%

Date: 1/22/2025, 12:56:02 PM Risk Level: low Risk Score: 18.06907907596037 Payment Method: Debit Card

[Select Search Type](#)


Suggested advanced searches

[Show me cases where additional tests are recommended](#)
[Show me patients that need urgent action](#)
[Show patients with asymmetric densities and microcalcifications](#)
[Retrieve records indicating a need for biopsy](#)
[Find patients with benign findings and fatty breasts](#)
[Show records with ill-defined masses and architectural distortion](#)
[Retrieve patients with masses and BI-RADS 3 or 4](#)

[Select Search Type](#)

1

Suggested advanced searches

[Show me cases where additional tests are recommended](#)[Show me patients that need urgent action](#)[Show patients with asymmetric densities and microcalcifications](#)[Retrieve records indicating a need for biopsy](#)[Find patients with benign findings and fatty breasts](#)[Show records with ill-defined masses and architectural distortion](#)[Retrieve patients with masses and BI-RADS 3 or 4](#)53 years old
Report Date: 12/29/2023, 5:14:00 AM[Ask a Question](#)**Clinical notes**

Patient's age: 53 years Clinical history: The patient has a family history of breast cancer, with the patient's mother diagnosed at the age of 48. Findings: Left Cranio-Caudal (L-CC) view: BI-RADS score 6 - Highly suggestive of malignancy. Left Medio-Lateral Oblique (L-MLO) view: BI-RADS score 4 - Suspicious abnormality, additional imaging and/or biopsy recommended. Right Cranio-Caudal (R-CC) view: BI-RADS score 5 - Highly suggestive of malignancy. Right Medio-Lateral Oblique (R-MLO) view: BI-RADS score 5 - Highly suggestive of malignancy. Impression: The

Conclusions

recommended for optimal management. Urgent follow-up is encouraged due to the high suspicion of malignancy and the patient's family history of breast cancer.

Search score: 0.493284

52 years old
Report Date: 9/20/2023, 9:59:28 AM[Ask a Question](#)**Clinical notes**

Patient's age: 52 years old Clinical History: The patient has a family history of breast cancer. BI-RADS scores: - Left Craniocaudal (L-CC): BI-RADS 2 - Left Mediolateral Oblique (L-MLO): BI-RADS 5 - Right Craniocaudal (R-CC): BI-RADS 5 - Right Mediolateral Oblique (R-MLO): BI-RADS 5 Impressions: 1. Left CC: No significant abnormalities identified, BI-RADS 2. 2. Left MLO: Suspicious findings for malignancy, BI-RADS 5. 3. Right CC: Suspicious findings for

Conclusions

steps. Urgent evaluation and expedited follow-up are strongly recommended, given the patient's family history of breast cancer and the high BI-RADS scores on the affected views. Early detection and intervention are crucial for optimal patient outcomes. Further evaluation with additional imaging modalities and potential biopsy will help determine the nature of the suspicious findings and guide appropriate treatment decisions. The patient should be informed of the



In conclusion

MongoDB **Search** and **Vector Search** in Enterprise Server enable developers to build **full-text and semantic search** features directly in self-hosted MongoDB, without external engines, simplifying development, reducing complexity, meeting strict regulatory requirements, and accelerating innovation in self-managed environments.



Are you ready to disrupt
industries and unleash the **power**
of software and **data**?

Learn more about MongoDB Industry Solutions





THANK YOU

MongoDB Search & Vector Search for use with Enterprise Server



Rodrigo Leal
Retail Industry Solutions Principal
rodrigo.leal@mongodb.com

