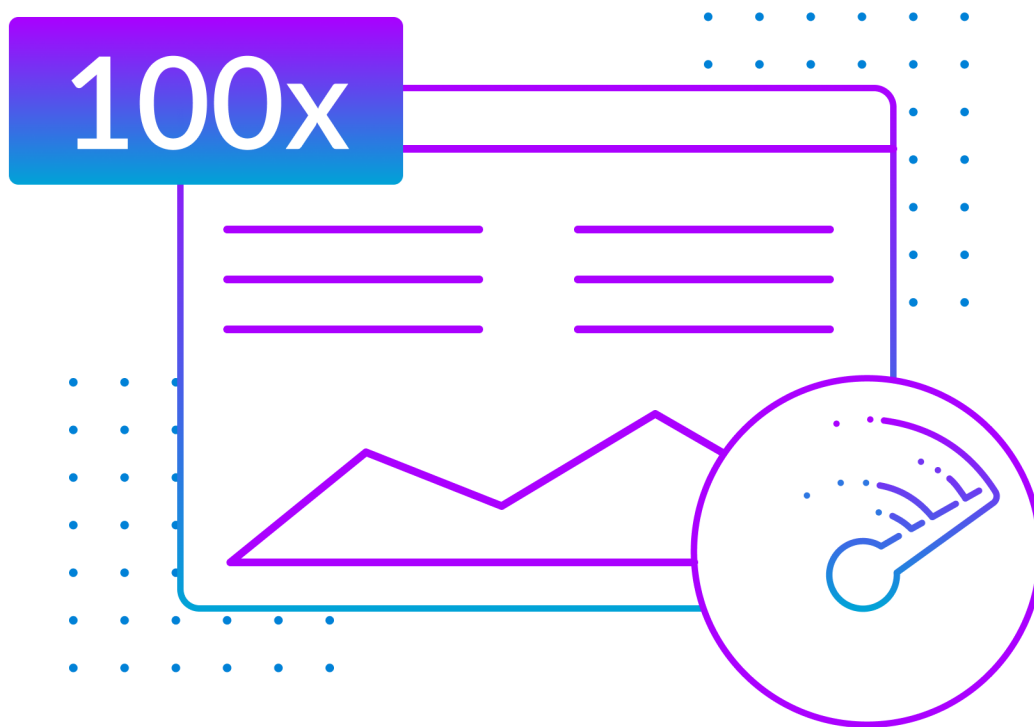


SingleStore Kai™: Driving 100x Faster Analytics for Your MongoDB® applications



SingleStore is not a MongoDB® partner. SingleStore Kai™ is an API that offers compatibility and enhanced performance for applications built on MongoDB®.



Table of Contents

Table of Contents	2
SingleStore Kai™ Overview	2
Why We Built SingleStore Kai™	3
Introducing SingleStore Kai™	4
Best of Both Worlds: Bringing the MongoDB® Developer Experience Together with the Benefits of SingleStoreDB	7
How We Built SingleStore Kai™	9
1. The MongoDB® proxy	9
2. Accessing collections and tables	10
3. New native replication for MongoDB®	12
Adoption Models for SingleStore Kai™	13
Augmenting MongoDB®	13
Replacing MongoDB®	14
Build Your Next Intelligent Application on SingleStoreDB	16
How to Get Started with SingleStore Kai™	16
Summary	18

SingleStore Kai™ Overview

[SingleStore Kai™](#) is an API to boost analytics on your MongoDB® applications by 100x without compromising transactional capabilities. SingleStore Kai allows developers to run blazing-fast analytics on JSON, without having to change queries or refactor your application code written for MongoDB®. This feature is [now available in public preview](#) and is a game-changer for app developers, enabling you to take advantage of both SQL and MongoDB® APIs in a single database engine to power fast, real-time applications.

Why We Built SingleStore Kai™

MongoDB® is one of the most popular and widely adopted databases for storing and processing JSON data. It is commonly utilized for its document-oriented data model, thanks to its simplicity and efficiency in handling JSON/BSON data formats. However being a document database, MongoDB® is not performant enough or designed to power real-time analytics on JSON, which modern applications demand. It has both functional and architectural limitations when it comes to analytics — especially around performing complex or fast analytics to power modern, interactive applications.

So today, customers invariably have to “flatten” complex JSON data and arrays, transform and ETL (or move) data from MongoDB® to other relational databases like Elastic, Rockset or Snowflake to perform analytics on the data.

This usually means normalizing data and re-writing MongoDB® queries for new analytical applications — a process that is cumbersome, error-prone and takes massive effort. And even then, with all the effort, it fails to deliver fast analytics and results in reduced data fidelity, increased latencies and data movement, and rising costs.

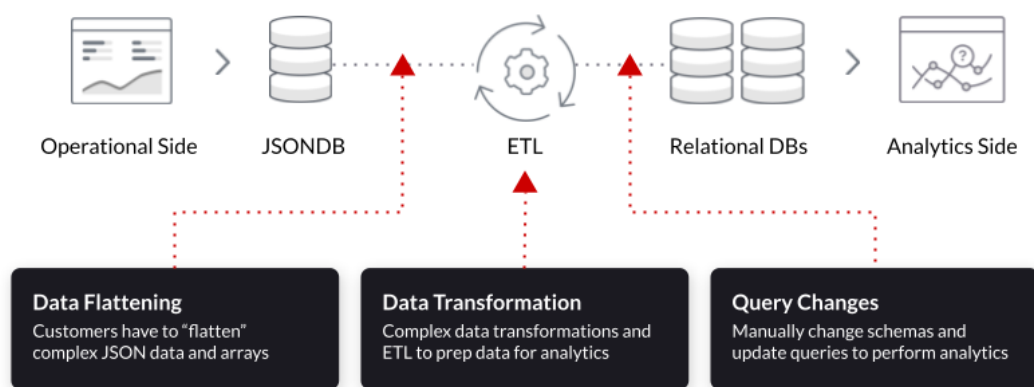


Figure 1. An example of a common workaround to power analytics on JSONDB

Apart from the analytical challenges, MongoDB® is not easy to scale out, requiring extensive expertise and efforts to scale using their shard key implementation. It is also a monolithic server where the compute-to-database relationship is one:many, and does not let app developers easily isolate their read workloads from write workloads. Finally, it provides little-to-no support for storing and running algorithms for vectors — which is a key capability for building and scaling modern generative AI applications.

Introducing SingleStore Kai™

SingleStore Kai™ is a fast, easy and powerful API to drive up to 100x faster analytics on your MongoDB® applications — without any query changes, application migration or data transformations. The API essentially enables you to power applications with real-time analytics on collections of JSON documents in SingleStoreDB using the same MongoDB® commands. See a quick [overview video](#) of SingleStore Kai™, or check out the [SingleStore Kai™ demo](#).

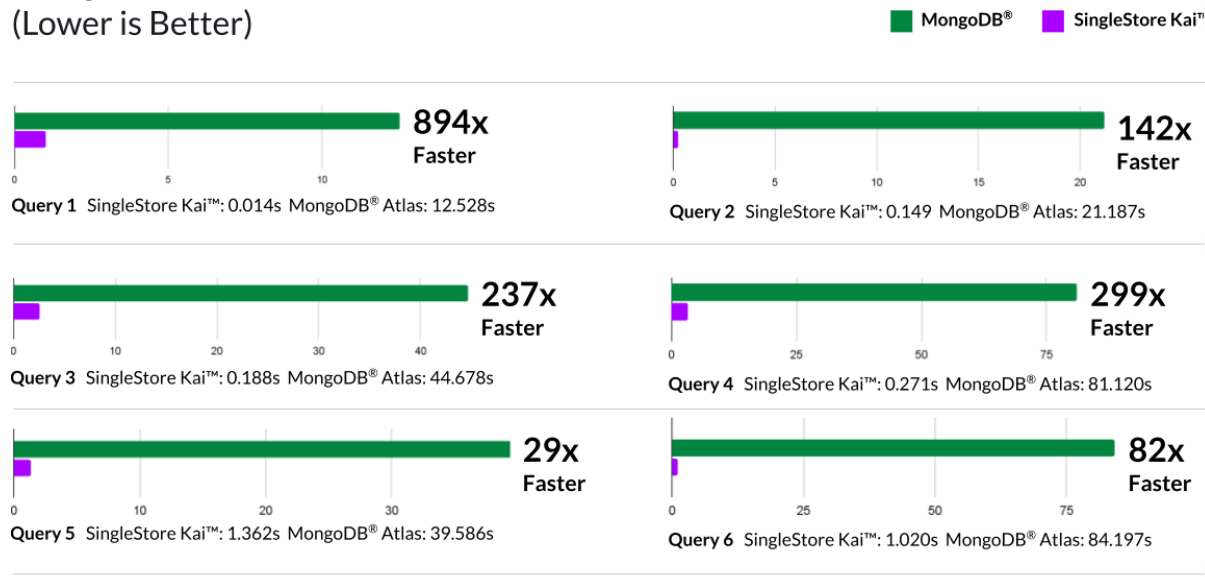
Some of the key benefits of SingleStore Kai™ include:

i) Turbocharge Analytics on JSON

With SingleStore Kai™, you can drive 100-1,000x faster analytics on JSON data for your MongoDB® applications. The API proxy translates MongoDB® queries into SQL statements that are executed by SingleStoreDB, driving lightning-fast analytics for your applications — something you can explore further in our [latest benchmarks](#) comparing analytical performance between SingleStoreDB and MongoDB.

Spoiler alert: SingleStoreDB with SingleStore Kai™ performed orders of magnitude (20-800x) faster for analytical queries compared to MongoDB® (see the following diagram).

Analytical Performance Benchmarks (Lower is Better)



We also ran the ClickBench analytics performance benchmark where 43 queries were executed over a data set of about 100 million records — and found that 80% of the queries were executed in half the time with SingleStore Kai™, compared to MongoDB®. And, 30% of the queries were at least 100x faster, with few queries even reaching over 1,000x faster. You can read more about our [internal and industry benchmarks here](#).

ii) Zero Code Changes & Data Transformations

Beyond the sheer speed boost, one of the most impressive aspects of SingleStore Kai™ is that developers no longer need to perform complex or manual query changes — and do not have to perform any data transformations or schema migrations.

You can continue to use the same MongoDB® commands without having to normalize, flatten or move data to a specialized analytical database to power super-fast analytics for your applications.

Moreover, you don't need to define a schema to ingest and index data. Data is indexed and transformed into a compressed columnstore format in real time, and lets you run any kind of complex SQL query while getting instant results. Often, little or no changes are required to existing applications — just change the connection string to point to SingleStoreDB. Check out this full breakdown of [how it works](#).

iii) Continue to Leverage MongoDB® Tools & Drivers

With SingleStore Kai™, developers can continue to use the same MongoDB® APIs, tools, drivers, endpoints, skill sets and ecosystem they are most familiar with. By supporting the MongoDB® wire protocol, SingleStore allows MongoDB® clients to communicate with a SingleStoreDB cluster — meaning developers who are familiar with MongoDB® can easily power real-time analytics on SingleStoreDB, without having to learn an entirely new set of tools or APIs.

You can stick with familiar methods not only for defining the business logic of your applications, but also for data definition, data manipulation (CRUD operations) and some database administration (status information).

iv) Best of Both Worlds (NoSQL + SQL)

SingleStore Kai™ empowers you to bring together the best of both worlds — including the schema flexibility and simplicity of a JSON document store, together with the speed, efficiency and complex analytical capabilities that only a relational SQL database can provide. That means you can use both a MongoDB® API and a SQL API to power your applications with analytics.

v) AI/Vector Functionality for JSON

The new era of generative AI requires real-time analytics on all data, including JSON collections. SingleStoreDB [supports vectors and fast vector similarity search](#) using `dot_product` and `euclidean_distance` functions. And with the launch of SingleStore Kai™, developers can now utilize the vector and AI capabilities on JSON collections within MongoDB® — powering use cases like semantic search, image recognition, similarity matching and more. To learn more check out this [demo: Vectors on JSON](#).

vi) Simplified Architecture & No Additional Costs

With the addition of SingleStore Kai™, SingleStoreDB becomes one of the leading databases available today, unifying fast transactions and analytics across two of the most widely used data APIs — MongoDB® and SQL. The result? You can simplify your overall architecture, streamline your data operations and leverage the benefits of both for your app development.

Best of all, SingleStore Kai™ is available at no additional cost, and is part of the Standard edition of SingleStoreDB Cloud.

Here is a quick breakdown of some common challenges and pain points with MongoDB® — and the benefits of using SingleStore Kai™:

Feature	MongoDB® Challenges/Pain Points	Benefits of SingleStore Kai™
Analytics	Slow performance for analytics	100 - 1,000x faster analytics
Scale-out	Cumbersome shard management	Automatic data distribution across any number of leaf nodes
Compute: Database relationship	One:many	Many:many
Vector data and functions	No support for vectors	Vector functions supported (dot_product and euclidean_distance functions)
APIs supported	MongoDB®	MySQL & MongoDB® APIs

Best of Both Worlds: Bringing the MongoDB® Developer Experience Together with the Benefits of SingleStoreDB

SQL is by far the most powerful language and industry standard for querying structured data, and is widely used in business intelligence and analytics applications. MongoDB®, on the other hand, is well-suited for storing and querying semi-structured data, making it popular for use cases like document management, content management, etc. What if you could bring these two together in a single engine to power modern applications?

SingleStore Kai™ is a big step in that direction.

For example, if you have an application that currently uses MongoDB® as its backend operational database, you can now power 100x faster analytics and utilize all the goodness of SingleStoreDB by simply [changing the connection string](#) of your application — all by pointing to a SingleStoreDB cluster instead of a MongoDB® cluster. You can then utilize the same MongoDB® commands and tools to interact with your SingleStoreDB cluster as you would with MongoDB®.

This compatibility also means developers can take advantage of the MongoDB® ecosystem and toolbox — which includes a range of tools and libraries for working with MongoDB® data — in SingleStoreDB. For example, you can use the MongoDB® shell to execute queries against your SingleStoreDB cluster, or use MongoDB® Compass to visually explore and analyze your data within SingleStoreDB.

Integrating The Best of Both Worlds (NoSQL + SQL)



You can also use the MongoDB® VC Code extension, or third-party components (like Prisma) that have a MongoDB® integration. You can find a non-exhaustive list of all the tools that we have tested and provided support for in our [SingleStore Kai™ documentation](#).

Now, the interoperability between SQL and Mongo is really compelling because as a developer, you do not have to choose between SQL or NoSQL (MongoDB® query language). Instead, you can now effectively utilize both languages with a single engine to power your applications. Furthermore, this feature enables organizations to take advantage of the strengths of both SQL and MongoDB®. When teams with different skill sets (SQL and NoSQL) are able to work with the same database to power applications, it leads to faster application development, better performance and more efficient data management.

And more importantly, all of SingleStoreDB's innate architectural and performance advantages — including [ultra-fast ingestion](#) with Pipelines, a horizontal scale-out architecture, highly performant columnstore for analytics, SIMD vectorization and compilation of queries to machine code — all work well for processing MongoDB® aggregation pipelines, just as it does for SQL.

All of this in addition to improvements in JSON as part of SingleStore 8.0 launch, where we introduced up to [400x faster seeks on JSON](#) by enabling data to be stored in a compressed format. And when specific rows need to be retrieved, only the retrieved data needs to be decompressed. To learn more, check out this [SingleStore Kai™ demo](#).

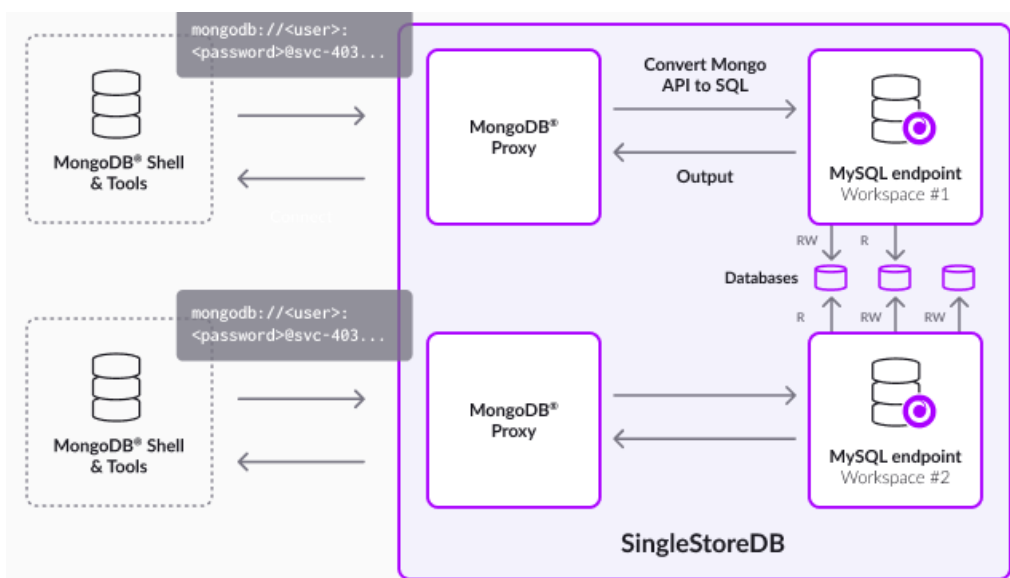
How We Built SingleStore Kai™

1. The MongoDB® proxy

The underlying architecture for SingleStore Kai™ is innovative, yet simple and efficient. As mentioned previously, we have developed a MongoDB® proxy that is MongoDB® wire-protocol compatible, effectively converting MongoDB® queries and operations into SQL commands that can be processed in the most optimal way by SingleStoreDB.

SingleStore Kai™ runs MongoDB® CRUD (create/update/delete) commands and aggregation pipelines (analytical queries) by using the MongoDB® Proxy server that essentially sits between the client application and SingleStoreDB. As indicated in the following diagram, the proxy is able to receive MongoDB® commands, generate SQL equivalents executed against SingleStoreDB, read the resulting rowsets and return the results as collections of BSON documents back to the MongoDB® client application.

SingleStore Kai™ provides compatibility and coverage that is already higher than other leading Mongo API-compatible databases, such as Azure CosmosDB and AWS DocumentDB, for both transactional and analytical commands. To learn more about our detailed compatibility and mapping, refer to our documentation page [here](#).



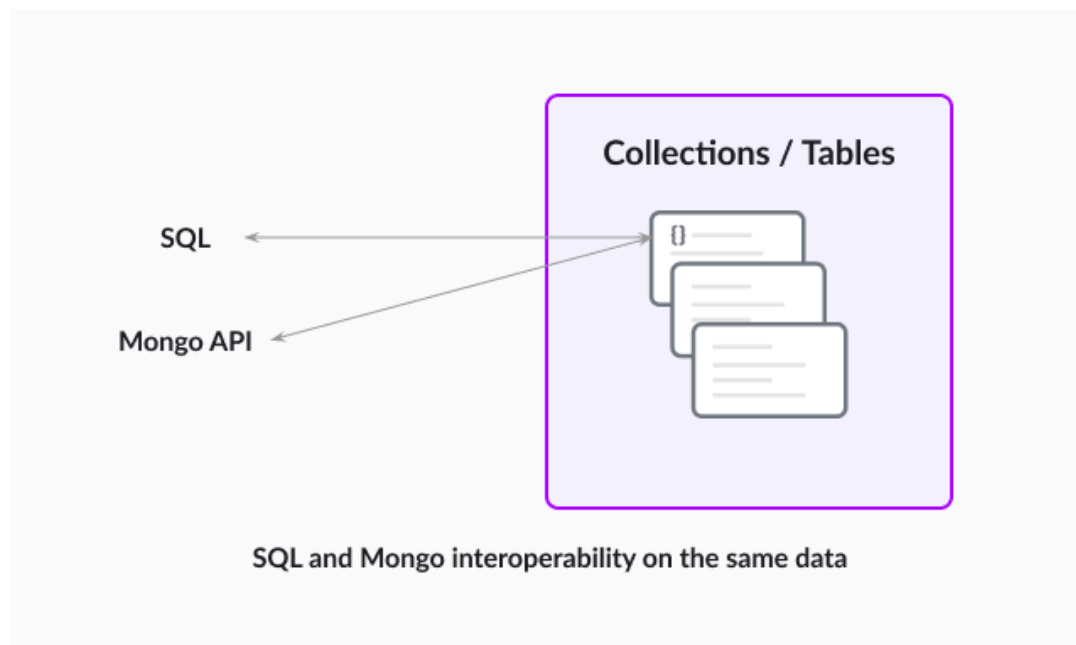
In terms of deployment, SingleStore Kai™ is enabled at a [workspace](#) level. Every MongoDB® API-enabled workspace has a dedicated MongoDB® proxy and endpoint provisioned. Furthermore, a database may be attached to multiple MongoDB® API-enabled workspaces simultaneously, allowing diverse workloads to operate on shared data.

In terms of security, the Mongo API-enabled workspaces already support [AWS Private Link](#) connections for enhanced security, reduced network latency and improved reliability.

Finally, in terms of performance tax, the Mongo proxy requires minimal processing power and does not negatively impact the query performance, as evidenced in the benchmarks.

2. Accessing collections and tables

SingleStoreDB provides interoperability between collections and tables, which is unique in the industry as teams with either SQL and MongoDB® skills can work on the same data — without duplication — to power their applications.



Data ingested through SingleStore Kai™ is stored in a collection. A collection is simply represented as a table with one JSON column for the `_id` and one JSON column for the rest.

You can also extract certain fields from that JSON column using features like [persisted computed columns](#) to improve analytical performance. For example, you can index a specific field in a collection using the MongoDB® `createIndex` command, which will create an index with a persisted computed column within the same collection.

The MongoDB® proxy can also access and execute transactions on a table. The same information can be surfaced as a flat JSON document when reading the table using SingleStore Kai™. Take a look at the following example:

Table created via SQL

```
CREATE TABLE example_table (  
  date_column DATE,  
  int_column INT,  
  float_column1 FLOAT,  
  float_column2 FLOAT);
```

```
INSERT INTO example_table  
(date_column, int_column,  
float_column1, float_column2) VALUES  
(  
  ('2022-01-01', 1, 1.1, 2.2),  
  ('2022-01-02', 2, 2.2, 3.3),  
  ('2022-01-03', 3, 3.3, 4.4),  
  ('2022-01-04', 4, 4.4, 5.5),  
  ('2022-01-05', 5, 5.5, 6.6);
```

Table read via SingleStore Kai™

Query
`db.example_table.find( { int_column: 2 })`

Result
[
 {
 date_column: ISODate("2022-01-02T00:00:00.000Z"),
 int_column: 2,
 float_column1: 2.2,
 float_column2: 3.3
 }
]

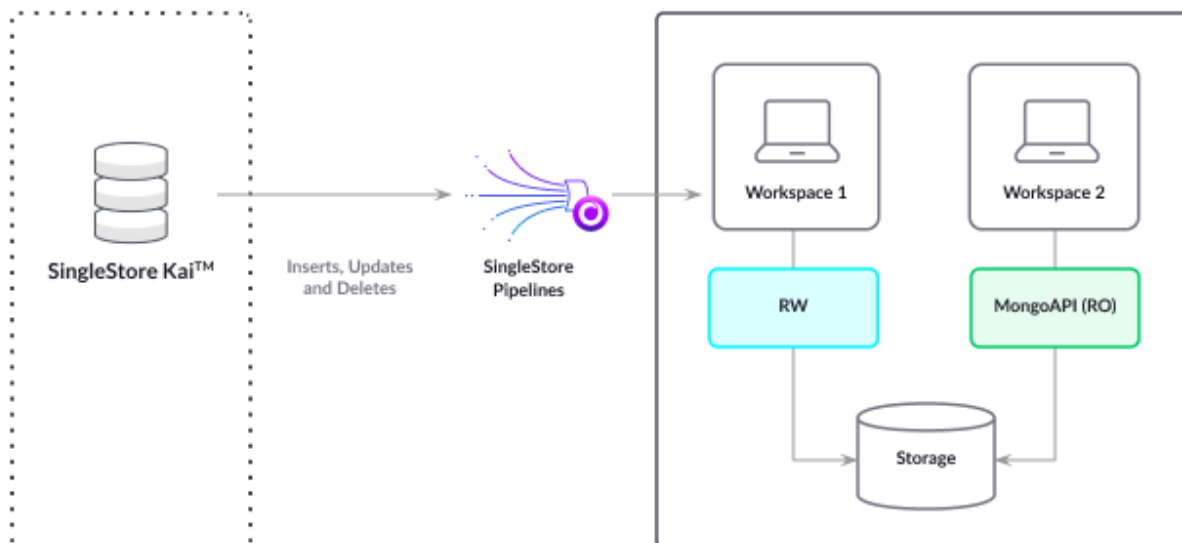
Inversely, you can also use SQL to read, write and ingest data into collections.

3. New native replication for MongoDB®

Finally, when it comes to getting data in, we have also introduced a service (currently in private preview) to quickly and easily migrate — or replicate — your collections from MongoDB® into SingleStoreDB at no additional cost. SingleStore's new data replication solution for MongoDB® provides customers the ability to do one-time replication (snapshot) loads and/or real-time CDC (Change Data Capture) from MongoDB® collections into SingleStoreDB.

This solution, based on [SingleStore Pipelines](#), is natively integrated with the SingleStoreDB engine, enabling customers to easily and effectively replicate/move data to quickly get started with the MongoDB® API. Not to mention, it provides superior [ingestion performance at high speeds](#).

Here is quick overview of the replication service:



Beyond the native tooling for replication previously mentioned, SingleStore also has partnerships with leading data integration firms like [Sapper](#) and [Arcion](#) that have proven expertise in integrating data from a variety of sources — including MongoDB® — to SingleStoreDB, with a fully managed user experience.

Adoption Models for SingleStore Kai™

When it comes to adoption models for SingleStore Kai™, you have a few options based on their specific business goals. You can either:

- Augment your existing MongoDB® instance with SingleStoreDB
- Replace your MongoDB® instance with SingleStoreDB
- Build a net-new application with blazing-fast analytics using SingleStoreDB

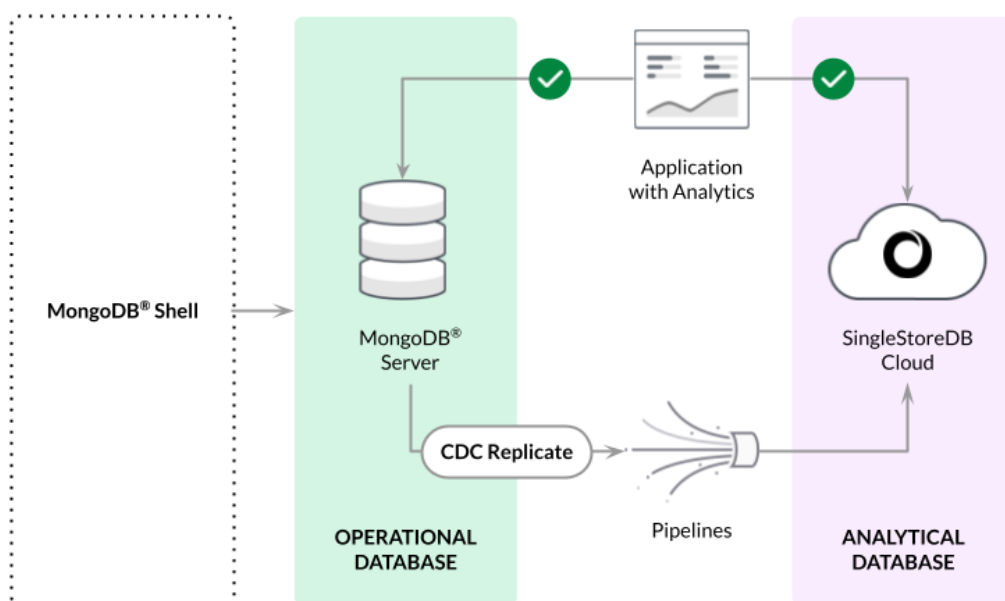
Let's dive a little deeper into the implementation details.

Augmenting MongoDB®

If you are primarily experiencing issues around running aggregates or performing fast, complex analytical queries using MongoDB®, then augmenting MongoDB® with SingleStoreDB (with Kai) might be an easy and effective option. In this augmentation model, MongoDB® continues to be the operational database, while SingleStoreDB becomes the analytical engine.

Collections of JSON that require analytics are replicated into SingleStoreDB (using our new MongoDB® replication tool), and the application runs ultra-fast analytical queries against SingleStoreDB. The key difference here is that no changes to your MongoDB® queries, data flattening or transformations are required. Just change the connection strings to point your analytical queries to SingleStoreDB, and this makes it simple, easy, efficient and blazing fast. You can even build additional logic with SQL or MongoDB® over your collections and tables.

Augment Your MongoDB® Server



This can be done in four easy steps:

1. Select the collections that require analytics
2. Replicate these collections to SingleStoreDB using the replication tool
3. Change Connection Strings
4. Run the same MongoDB® queries against SingleStoreDB — and watch them speed up to 100x faster

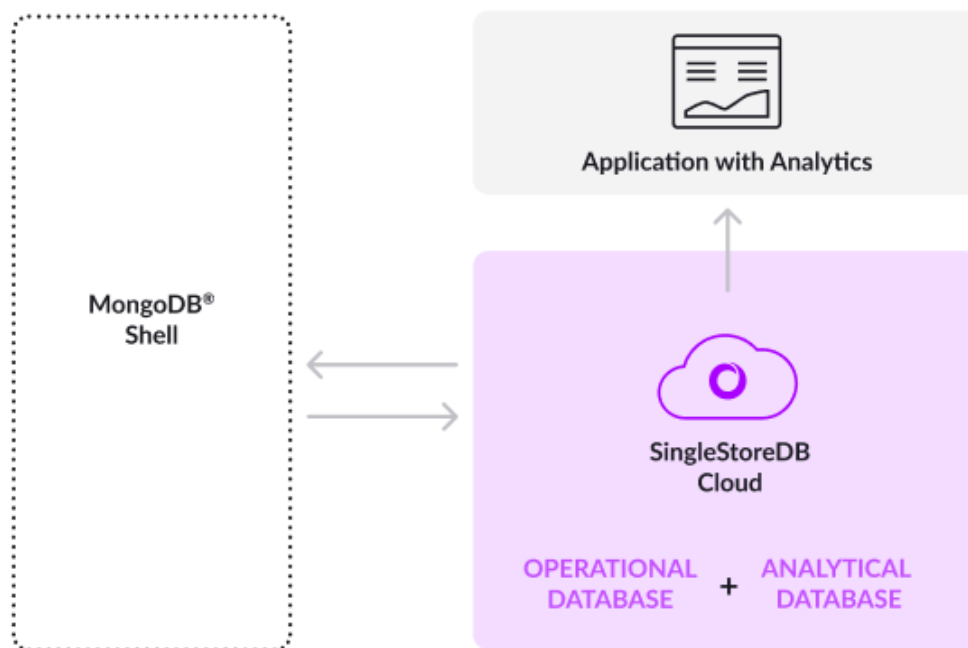
Replacing MongoDB®

Now, if your primary goal is to perform fast analytics while minimizing data replication, SingleStoreDB can replace MongoDB® as your primary database, becoming the sole database for both operations and analytics. In this case, you don't need to change the application code, and can continue to use the same MongoDB® CRUD operations and queries (but point them to SingleStoreDB).

The coverage of our API is in line with similar offerings from AWS DocumentDB or Azure Cosmos DB.

You can take advantage of all of the SingleStoreDB capabilities described earlier — including ultra-fast ingestion, a single universal store for transactions and analytics, horizontal scale-out architecture, highly performant columnstore for analytics, SIMD vectorization and compilation of queries to machine code —and continue to use SingleStoreDB as the single unified database for transactions and analytics.

In this case, you will move all collections and data into SingleStoreDB, running all operations against it as the primary database and analytical engine. And as with the other scenarios, you can continue to use the same drivers and tools, and utilize the same MongoDB® queries against SingleStoreDB.



Build Your Next Generative AI Application on SingleStoreDB

This is ideal for customers looking to build their next intelligent, generative AI application but want to utilize the best of what SQL and NoSQL offer to power those apps. As mentioned previously, with SingleStore Kai™, you do not have to choose between SQL or MongoDB® (NoSQL). Instead, you can now effectively utilize the best of both worlds within a single engine to power your applications.

More importantly, the trend of using vector engines to [build generative AI applications](#) has reached a tipping point with the advent of ChatGPT. With the ease of vectorizing pictures, sounds, videos or texts and implementing matching algorithms at scale, there has never been a better time to adopt this approach.

SingleStoreDB supports [vectors and vector similarity](#) search using dot_product (for cosine similarity) and euclidean_distance functions. These functions are used by our customers to power applications including facial recognition, image matching, visual product/photo search and text-based semantic search.

We not only support vector functions through SQL but now through SingleStore Kai™ as well. So MongoDB® developers can now call, process and run vector functions in SingleStoreDB using the API, which makes building intelligent applications simpler and easier than ever.

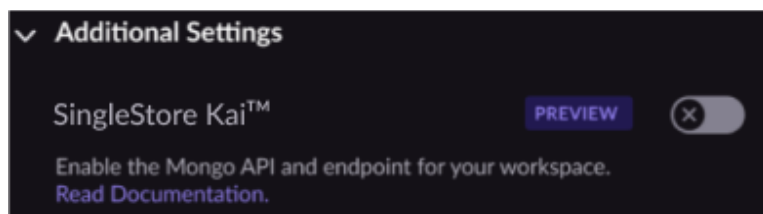
With SingleStoreDB, you can take your applications to the next level and unlock the full potential of vectorization technology in MongoDB®.

How to Get Started with SingleStore Kai™

SingleStore Kai™ is now available in public preview with no additional cost as part of our [SingleStoreDB Cloud](#) offering.

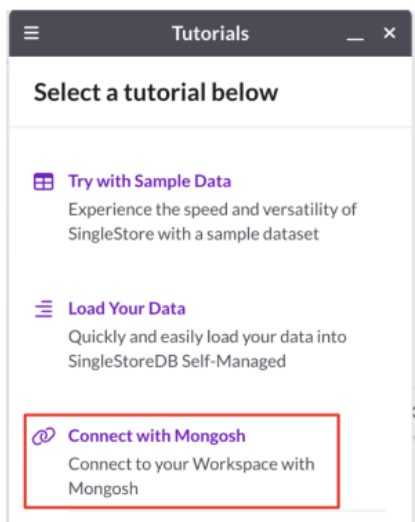
We currently support SingleStore Kai™ across 10 AWS regions globally, and plan to roll out that capability to Azure and GCP in the next few months.

If you are new to SingleStore, you can start your free trial [here](#). If you are an existing customer or already have access to SingleStoreDB, start by creating a new workspace in an AWS region, and make sure that “SingleStore Kai™ MongoDB® API” is enabled in the Advanced Settings.



In addition to the [SingleStore Kai™ documentation pages](#) and [SQrL](#) (our expert bot trained over SingleStore content), you will get a few additional resources:

- A guide on how to connect to Mongo using Mongosh.



- One sample notebook with PyMongo to quickly import sample collections from MongoDB® Atlas into SingleStoreDB
- A replication notebook template to copy or quickly replicate your data from your Mongo cluster to SingleStoreDB

Summary

Modern analytical applications need to drive fast analytics on JSON to deliver interactive, engaging user experiences (in-app analytics). SingleStore Kai™ provides a simple, easy and ridiculously fast API that can drive up to 100-1,000x faster analytics for your MongoDB® applications – without extensive query changes or modifications to your application.

And, developers can easily augment (or replace) their MongoDB® applications with SingleStoreDB to dramatically improve performance and scalability, and utilize both SQL and NoSQL to power their applications. The service is available to try out for free and is now available as part of SingleStoreDB Cloud.

For more information:

- Learn more about [SingleStore Kai™](#)
- [Try SingleStore Kai™](#) free, and get \$600 in free credits
- [Watch a demo](#)
- See the latest [benchmarks for SingleStore Kai™](#)
- Dig deeper into [documentation](#)
- [Schedule a deep dive](#) with one of our Solution Engineers