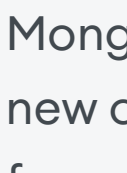


MongoDB Announces New Time Series Capabilities Coming for MongoDB 6.0

Try MongoDB Atlas for Free Today



MongoDB

June 7, 2022 | Updated: November 21, 2022

[#Time Series](#) [#MongoDB World](#)

MongoDB 6.0 will come with a wealth of exciting new features for customers, including new capabilities for managing time series data. Notable improvements include support for secondary indexes on time series measurements, which makes it easier for users to index data that varies over time. We've also made read performance improvements for sort operations.

These enhancements come as part of a steady drumbeat of new [time series](#) capabilities MongoDB has added since the release of MongoDB 5.0. Over the past year, we've released significant new capabilities that make building and running applications that rely on time series data easier for developers. These features mean developers can manage time series data with greater ease, faster speed, and lower cost than ever before.

MongoDB supports time series data for the full data lifecycle, including ingestion, storage, querying, analysis, visualization, and online archiving or automated expiration as data ages. In MongoDB 6.0, extensive support for time series data will be available to all MongoDB customers, not just MongoDB Atlas customers who have opted to participate in the [Rapid Release program](#).

Keep reading to learn more about what time series data is, discover what we've done with time series collections since MongoDB 5.0, and preview what's coming next.

The Time Series Landscape

Time series data is a sequence of measurements over a period of time with common metadata. Managing time series workloads is required across a variety of industries. Examples include sensor readings for manufacturing; vehicle-tracking device logs for transportation and logistics; data from consumer IoT devices, such as smart watches; customer interaction data in e-commerce; and financial transactions data for the securities and cryptocurrency industries.

Time series data is everywhere, and companies need to collect and analyze this data to understand what is happening right now in their businesses, and to assess future needs quickly.

Time series workloads typically have the following attributes:

- Inserts arrive in batches that are sequentially ordered by time
- Data sets are typically append-only
- Older data can be deleted or archived
- Queries and aggregations are typically applied to data within a specified time range

With these unique qualities come a variety of challenges for developers looking to build applications that leverage time series data. One is data volume: Time series workloads can generate data many times per second, making storage capacity (and associated costs) a significant concern. Another is continuity of data. Gaps in time series data – for example, when sensors go offline – can make analyzing the data significantly more difficult.

Time series workloads across numerous industries have rapidly increased. As a result, MongoDB has significantly invested in advancing our capabilities in this space and empowering developers to build best-in-class applications using time series data on MongoDB.

Time Series in MongoDB 5.0

In MongoDB 5.0, we added:

Time series collections: Since MongoDB's earliest days, developers have used our platform to store time series data. But doing so efficiently required careful, expert data modeling that wasn't achievable for every organization. Starting with MongoDB 5.0, customers gained access to a new type of collection in MongoDB specifically for time series data that can sit side by side next to other collections in a MongoDB database. Within a time series collection, writes are organized so that data from the same source is stored in the same block alongside other data points from a similar period of time. Data in the time series collections is stored in an optimized columnar format ordered by time, which powers performance at scale – drastically reducing storage and I/O for read operations.

Visualization of time series data in MongoDB Charts: Many organizations working with time series data want to analyze it to diagnose issues and predict trends that affect their business. With time series data in MongoDB Charts, customers got instant visibility into trends in their time series data.

Window functions and temporal expressions: The MongoDB Query API expanded with new queries for time series data – specifically, window functions for query operations on related data and temporal expressions to help users uncover hidden patterns quickly.

Secondary indexing on metadata: MongoDB added support for creating secondary indexes on time series metadata. While the default time series indexes already supported queries on time and metadata as a whole, this new capability allowed users to create secondary indexes on specific metadata subfields required for more efficient query execution.

Time Series in MongoDB 5.1

In MongoDB 5.1, we added:

Sharding for time series collections: With the release of MongoDB 5.1, time series collections could take advantage of MongoDB's native sharding to horizontally distribute massive data sets. This improves throughput and cardinality-handling. Nodes are co-located with data producers to support local write operations and to enforce data sovereignty controls.

Multi-delete operations: Support was added for multi-delete operations against a time series collection's metadata field. Although most time series collections are append-only, organizations need to be able to delete data to accommodate when customers invoke their right to erasure. The option to execute multi-delete operations gives developers and administrators an easy way to comply with these types of modern data privacy regulations.

Atlas Online Archive support in Preview: One common challenge with time series is that the rapid proliferation of data can lead to rising storage costs. With Atlas Online Archive support for time series collections, users can define archiving policies to automatically move aged data out of the database and into lower-cost, fully managed cloud object storage. Users simply define their own archiving policy and Atlas handles all the data movement for them. This allows users to retain all of their time series data for analysis or compliance purposes while also lowering costs.

Broader platform support for time series: MongoDB released broader platform support for time series data, including the ability to create time series collections directly from Atlas Data Explorer, [MongoDB Compass](#), or the [MongoDB for VS Code Extension](#).

Time Series in MongoDB 5.2

In MongoDB 5.2, we added:

Columnar compression: With the addition of columnar compression for time series, organizations are able to dramatically reduce their storage footprint. This new capability means that time series data in BSON is significantly compressed in time series buckets before undergoing even further compression for storage on disk. Columnar compression leads to a 10 to 30x improvement in WiredTiger cache usage; it also significantly improves query performance by reducing disk I/O since more data can fit in memory.

New aggregation operators: The release of MongoDB 5.2 included new operators and enhancements to improve aggregation performance and reduce the number of aggregation stages needed to unlock insights. These new operators are valuable for analyzing a variety of types of data, including time series:

- The `$top` and `$bottom` operators allow users to compute the top and bottom N elements of a dataset and return related fields within the same query without requiring complex logic.
- `$maxN`, `$minN`, and accumulators such as `$firstN`, `$lastN`, which return elements while taking into account the current order of documents in a dataset.

Time Series in MongoDB 5.3

In MongoDB 5.3, we added:

Densification and gap-filling: It is common for time series data to be uneven (for example, when a sensor goes offline and several readings are missed). In order to perform analytics and ensure the correctness of results, however, the data being analyzed needs to be continuous. Densification and gap-filling allow users to better handle missing time series data by creating additional data points to compensate for missing values.

What to Expect With Time Series in MongoDB 6.0

In MongoDB 6.0, we have added the following features for time series:

Secondary indexes on measurements: MongoDB customers will be able to create a secondary or compound index on any field in a time series collection. This enables geo-indexing (for example, tracking changes over time on a fleet of vehicles or equipment). These new index types also provide improved read performance.

Read performance improvements for sort operations: MongoDB 6.0 comes with optimizations to last point queries, which let the user find the most recent entry in a time series data collection. The query executes a distinct scan looking only for the last point rather than executing a scan of the full collection. Also included in this release will be a feature that enables users to leverage a clustered index or secondary index to perform sort operations on time and metadata fields in most scenarios in a more optimized way.

What's Next

Our work on time series will not stop with MongoDB 6.0. We will continue to empower developers to build best-in-class applications using time series data on MongoDB. In future releases, expect to hear about cluster-to-cluster replication for time series data, features to enhance scalability for time series data on MongoDB, and much more.

Sign up for the MongoDB Atlas Dedicated Tier (M10+) and choose the MongoDB 6.0.0 Release Candidate to get all the latest innovations for time series workloads on MongoDB.



← Previous

Working Together: MongoDB's Partner of the Year Awards

Organizations face more pressure than ever to modernize, to innovate, and to become more data-driven....

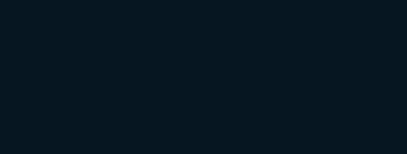
June 7, 2022

Next →

How the NFSA is Using MongoDB Atlas and AI to Make Aussie Culture Accessible

Where can you find everything from facts about Kylie Minogue, to more than 6,000 Australian home movies,...

May 14, 2024



About

Careers

Investor Relations

Legal Notices

Privacy Notices

Security Information

Trust Center

Support


Contact Us


Customer Portal

Atlas Status


Customer Support


Social


 GitHub

 Stack Overflow

 LinkedIn

 YouTube

 X

 Twitch

 Facebook

 Contact Us