

Isaac Araya Solano | Carnet: 2018151703

Resumen #4

MongoDB Architecture Guide

Introduction

Old tools are becoming obsolete since they cannot handle the amounts of data that are being created nowadays. That is the reason why NoSQL databases have been growing over the past decade.

The Document Model

MongoDB was launched in 2009 as a completely new class of general purpose database. It retains the best from SQL and NoSQL DBs while providing a technology foundation competent enough to meet the demands of modern apps. Instead of storing info as tables, it can be stored as JSON, making it easy to modify the structure of the docs and store any kind of data.

Improved support for time series data

Time series data is the one that represents changes over time. It is critical for modern apps, specially for IoT, stock trading, clickstreams and social media. Due to MongoDB's flexibility, it's been used for these type of databases for years.

Advantages of Document Data Model

Intuitive: faster and easier for developers

Documents in the DB map directly to objects in your code, making them more natural to work with. With this model there is no need to decompose data across tables or run complex joins.

Flexible Schema: dynamically adapt to change

A document's schema is dynamic and self describing, so it does not need to be predefined in the database. This allows developers to adapt to change quickly and easily since you can modify the structure at any time.

Universal: JSON documents are everywhere

JSON is lightweight and language-independent, so it's easy to work with and has become an standard for data communication and storage. You can store the data in the structure that your app needs. Mongo stores the data in BSON that is the Binary representation of JSON.

Working with Document Data

The MongoDB Query API is comprehensive and expressive. Ad hoc queries, indexing, and real-time aggregations provide powerful ways to access, group, transform, and analyze data.

The MongoDB Aggregation Pipeline allows you to transform and analyze data. The most basic pipeline stages provide filters that operate like queries and document transformations that modify the form of the output document.

With ACID transactions, you can maintain the same all-or-nothing and snapshot isolation guarantees as with relational databases. With strong data consistency, MongoDB eliminates the application complexity imposed by eventually consistent NoSQL systems.

MongoDB goes beyond many other databases, when it comes to act on data in real time, with features such as Change Streams, which automatically detects and notifies consuming applications of any data modifications in the database.

To further accelerate developer productivity, MongoDB provides native drivers for popular programming languages and frameworks. Supported drivers include Java, JavaScript, C#/.NET, Go, Python, PHP, Scala, Rust, and more.

Distributed Architecture: Scalable, Resilient, and Mission Critical

Through replica sets and native sharding, MongoDB enables you to scale out your applications with always-on availability.

Availability and data protection with replica sets

MongoDB replica sets enable you to create up to 50 copies of your data. They are predominantly designed for resilience. If a primary node suffers an outage or is taken down for maintenance, the MongoDB cluster will automatically elect a replacement in a few seconds.

Replica sets are elected by sophisticated algorithms. Mongo provides a lot of services and protocols in order to protect your data and transactions.

Scale Up, Out, and Across Storage Tiers

Like most databases, you can scale MongoDB vertically by moving to larger or smaller instance sizes. As a distributed system, MongoDB can perform a rolling restart of the replica set, enabling you to move between different instances without application downtime.

Native sharding also allows you to seamlessly scale out your database across multiple nodes and without adding complexity to the app. Shards can be added or removed at any time. Mongo automatically rebalance data across the shards.

MongoDB native sharding gives you the following options:

- **Ranged sharding:** Documents are partitioned across shards according to the shard key value.
- **Hashed sharding:** Documents are distributed according to an MD5 hash of the shard key value.
- **Zoned sharding:** This allows developers to define specific rules governing data placement in a sharded cluster

Privacy and Security

MongoDB includes extensive capabilities to defend, detect, and control access to data:

- **Authentication:** MongoDB offers a strong challenge-response mechanism based on SCRAM-256, along with integration to enterprise security infrastructure, including LDAP, Windows Active Directory, Kerberos, x.509 certificates, and AWS IAM.
- **Authorization:** Role-based access control (RBAC) enables you to configure granular permissions for a user or application based on the privileges they need to do their jobs.
- **Auditing:** For regulatory compliance, security administrators can use MongoDB's native audit log to record all database activity and changes.
- **Network isolation:** For users running fully managed databases in MongoDB Atlas, user data and underlying systems are fully isolated from other users. Database resources are associated with a user group, which is contained in its own virtual private cloud (VPC). Access can be granted only by IP whitelisting or VPC peering.
- **Encryption everywhere:** MongoDB data can be encrypted while in motion across the network, while in use in the database, and while at rest, whether on disk or in backups.

With client-side field-level encryption (FLE), you have access to some of the most advanced data protection controls anywhere.

One Platform for All Your Workloads

MongoDB Atlas application data platform delivers a cohesive and integrated set of data and database services. Key characteristics of MongoDB's application data platform include:

- A data plane with the ability to support a wide variety of application types
- A unified and consistent experience for developers, data analysts, data scientists, and critical supporting functions such as operations teams, security teams, and data engineers
- Global, multi-cloud data distribution to support data residency requirements and provide deployment flexibility
- Transparent data movement between services and automated data life-cycle management

Real-Time Analytics

Atlas Database supports read-only analytics nodes that can handle more intensive analytics queries. By configuring the read preference, you can direct analytics queries to these nodes without affecting operational workloads.

Atlas Search

Allows for easy creation of fast full-text search capabilities on MongoDB data without additional APIs or technologies. It uses Apache Lucene and offers features such as fuzzy search, autocomplete, and custom scoring.

Atlas Data Lake

Allows you to analyze data in Amazon S3 using the MongoDB Query API, without the need to set up or manage infrastructure. It parallelizes operations and optimizes workloads, utilizing compute in the region

closest to your data. It also supports federated queries, enabling you to combine and analyze data across S3 and your Atlas database clusters.

Atlas Charts

Atlas Charts is a data visualization service that allows you to create charts and dashboards from MongoDB data in a drag-and-drop interface. It can read from analytics or secondary nodes without affecting operational workloads and supports sharing and embedding. It can use data from Atlas Database deployments or Atlas Data Lake.

Realm Sync

Realm Sync enables bidirectional data sync between Atlas Database clusters and Realm, an open-source mobile database. Realm's SDKs allow developers to access data stored in MongoDB Atlas and interact with the platform's set of services directly from the client.

GraphQL

With MongoDB, you can automatically create a JSON schema and use GraphQL to enable a single endpoint for querying your MongoDB collections. This allows you to build performant applications and easily secure them with built-in authentication and data access control.

Event-Driven Architecture

Atlas Triggers automatically execute functions in real-time in response to changes in the database or user authentication events, or at preset intervals.

MongoDB for Mission-Critical Applications in Your Data Center

MongoDB Enterprise Advanced is a package of advanced software, support, certifications, and other services that allows you to run MongoDB on your own self-managed infrastructure. MongoDB Enterprise Server is a version of the database software that includes advanced features such as an in-memory storage engine, advanced security options, comprehensive auditing, and an encrypted storage engine. MongoDB Ops Manager simplifies the administration tasks associated with running MongoDB on premises or in a private cloud. The MongoDB Enterprise Operator for Kubernetes automates and manages MongoDB clusters on Kubernetes. The MongoDB Connector for BI lets you use MongoDB as a data source for your existing SQL-based BI and analytics platforms.

Run MongoDB for Free With Tools From Us

MongoDB Community Server is the free, source-available version of the database software.

- **MongoDB Compass:** GUI for MongoDB
- **Cloud Manager:** the cloud-based management platform that enables you to deploy, monitor, back up, and scale MongoDB.
- **Connectors:** The MongoDB Connector for Apache Spark lets you use all of Spark's libraries to analyze MongoDB data using various APIs.