

MongoDB Architecture Guide

The Document Model

MongoDB was launched in 2009 as a general-purpose database and quickly established itself as one of the most popular databases among developers. This database replaces tables of relational databases with documents that store data as a JSON (JavaScript Object Notation). This lets Mongo store any type of data, and the structure can easily be modified. The document data model have the advantages of being intuitive, flexible and universal.

Intuitive

Documents directly map to the objects in code, so they are natural to work with. The following example of a JSON document in MongoDB demonstrates how a customer object is modeled in a single document structure with related data embedded as subdocuments and arrays. This approach collapses what would otherwise be seven separate parent-child tables linked by foreign keys in a relational database.

```
{
  "_id": ObjectId("5ad88534e3632e1a35a58d00"),
  "name": {
    "first": "John",
    "last": "Doe"
  },
  "address": [
    {
      "location": "work",
      "address": {
        "street": "16 Hatfields",
        "city": "London",
        "postal_code": "SE1 8DJ"
      },
      "geo": {
        "type": "Point",
        "coord": [51.5065752, -0.109081]
      }
    },
    {
      "location": "home",
      "address": {
        "street": "16 Hatfields",
        "city": "London",
        "postal_code": "SE1 8DJ"
      },
      "geo": {
        "type": "Point",
        "coord": [51.5065752, -0.109081]
      }
    }
  ],
  "phone": [
    { "location": "work",
      "number": "+44-1234567890" },
    { "location": "home",
      "number": "+44-1234567890" }
  ],
  "dob": ISODate("1977-04-01T05:00:00Z"),
  "retirement_fund": NumberDecimal("1292815.75")
}
```

Flexible

In a document database, the schema is dynamic and self-describing, so there's no need to predefine it in the database. This allows for greater flexibility and the ability to continuously integrate new application functionality without having to perform disruptive schema migrations. Additionally, schema validation can be used to enforce rules governing the structure of documents and apply data governance standards to a

document schema. Overall, using a document database like MongoDB offers benefits such as greater flexibility, easier schema management, and improved data governance.

Universal

JSON is lightweight and language-independent, and it allows for flexible data structuring, including rich objects, key-value pairs, tables, geospatial and time series data, and graph structures. MongoDB stores data as JSON documents in BSON, which extends the JSON representation to include additional types such as int, long, date, floating point, and decimal128. This makes it easier for applications using MongoDB to reliably process, sort, and compare data. Overall, the use of JSON and BSON in MongoDB enables greater flexibility and reliability in data communication and storage.

Working With Document Data

The MongoDB Query API is comprehensive and expressive, with powerful ad hoc queries, indexing, and real-time aggregations. The Aggregation Pipeline allows for efficient data transformation and analysis. ACID transactions and strong data consistency guarantees ensure reliability and eliminate the complexity of eventually consistent NoSQL systems. MongoDB also offers features such as Change Streams for reactive event-driven data pipelines. The MongoDB Query API is easy to use and eliminates the need for cumbersome ORM abstraction layers, with native drivers available for popular programming languages and frameworks. Overall, MongoDB offers a comprehensive and efficient solution for data querying, transformation, and analysis with features that address the differences between databases.

Distributed Architecture

With up to 50 copies of data that can be provisioned across separate nodes, data centers, and geographic regions, replica sets offer resilience against primary node outages or maintenance. MongoDB's cluster will automatically elect a replacement in a few seconds, switching over client connections and retrying any failed operations. The replica set election process is controlled by sophisticated algorithms based on an extended implementation of the Raft consensus protocol. Replica sets can also be used to scale read operations and intelligently route queries to the closest node, minimizing read latency. Additionally, MongoDB's write concern ensures write operations propagate to a majority of replicas in a cluster for greater data durability. Replica sets are fully tunable, with custom write concerns that target specific members of a replica set, deployed locally and in remote regions, for reduced risk of data loss in the event of a complete regional failure. Overall, MongoDB replica sets offer comprehensive data protection, resilience, and scalability features that can be tailored to meet specific needs.

Scale Up, Out, and Across Storage Tiers

MongoDB can perform a rolling restart of the replica set to move between different instances without application downtime. With native sharding, MongoDB can scale out the database across multiple nodes to handle write-intensive workloads and growing data sizes. MongoDB offers different sharding policies for distributing data according to query patterns or data placement requirements, including ranged sharding, hashed sharding, and zoned sharding. Additionally, MongoDB Atlas Online Archive enables tiered scaling, automatically tiering aged data out of the database and into cloud object storage for more economical data storage without losing access to the data.

Overall, MongoDB provides comprehensive scaling options for adapting to evolving workload demands and growing data sizes.

Privacy and Security

MongoDB provides extensive capabilities for defending, detecting, and controlling access to data in response to increasing governmental oversight of privacy and data security. These capabilities include authentication mechanisms, integration with enterprise security infrastructure, and authorization based on role-based access control (RBAC). MongoDB's native audit log records all database activity and changes for regulatory compliance. Network isolation is available for users running fully managed databases in MongoDB Atlas. Encryption is available for data in motion across the network, in use in the database, and at rest on disk or in backups. MongoDB's client-side field-level encryption (FLE) allows for selective encryption of individual document fields with their own keys, and decryption is handled seamlessly on the client side. FLE provides advanced data protection controls, including compliance with privacy regulations such as GDPR and CCPA, and ciphertext rendering of encrypted fields on the server, making them unreadable to any party that does not have both client access and the necessary decryption keys. Overall, MongoDB provides comprehensive capabilities for securing data and complying with privacy regulations.

One Platform

The MongoDB Atlas is an application data platform that offers a cohesive and integrated set of data and database services. It streamlines how teams work with data in the context of building software and systems that deliver real-time experiences to both end customers and internal users. It provides a data plane with the ability to support a wide variety of application types that can be independently developed, deployed, and evolved to address a wide variety of use cases. It also provides a unified and consistent experience for developers, data analysts, data scientists, and critical supporting functions such as operations teams, security teams, and data engineers. The Atlas Database provides a general-purpose database for modern applications with flexible document data model, rich and expressive query API, and native support for both vertical and horizontal scaling. It is available in more than 80 regions across AWS, Google Cloud Platform, and Azure, and offers best-in-class infrastructure and database automation that ensures continuous availability, elastic scalability, and compliance with the most demanding data security and privacy standards. Additionally, it offers complementary services that allow organizations to support a wide range of application and real-time analytics data workloads.

The MongoDB Atlas application data platform offers several services to streamline working with data and building real-time experiences. It includes features like a read-only analytics node, Atlas Search for full-text search capabilities, Atlas Data Lake for analyzing cloud object storage, Atlas Charts for data visualization, and Realm Sync for data synchronization between Atlas Database clusters and Realm. It also provides GraphQL for highly performant applications, event-driven architecture using functions, and several deployment options like global and multi-cloud clusters, fully managed backups, and fine-grained monitoring.