

MongoDB

From Wikipedia, the free encyclopedia

MongoDB (from *humongous*) is a cross-platform document-oriented database. Classified as a NoSQL database, MongoDB eschews the traditional table-based relational database structure in favor of JSON-like documents with dynamic schemas (MongoDB calls the format BSON), making the integration of data in certain types of applications easier and faster. Released under a combination of the GNU Affero General Public License and the Apache License, MongoDB is free and open-source software.

First developed by the software company 10gen (now MongoDB Inc.) in October 2007 as a component of a planned platform as a service product, the company shifted to an open source development model in 2009, with 10gen offering commercial support and other services.^[1] Since then, MongoDB has been adopted as backend software by a number of major websites and services, including Craigslist, eBay, Foursquare, SourceForge, Viacom, and The New York Times among others. As of 2014, MongoDB was the most popular NoSQL database system.^[2]

MongoDB



Developer(s)	MongoDB Inc.
Initial release	2009
Stable release	3.0.3 / 12 May 2015
Development status	Active
Written in	C++, JavaScript, C
Operating system	Cross-platform
Available in	English
Type	Document-oriented database
License	GNU AGPL v3.0 (drivers: Apache license)
Website	www.mongodb.org (https://www.mongodb.org)

Contents

- 1 Licensing and support
- 2 Main features
- 3 Criticisms
- 4 Language support
- 5 Management and graphical front-ends
 - 5.1 Third-Party GUI tools
- 6 Popularity
- 7 Production deployments
- 8 See also
- 9 References
- 10 Bibliography
- 11 External links

Licensing and support

MongoDB is available for free under the GNU Affero General Public License.^[3] The language drivers are available under an Apache License. In addition, MongoDB Inc. offers proprietary licenses for MongoDB.^[1]

Main features

Some of the features include:^[4]

Document-oriented

Instead of taking a business subject and breaking it up into multiple relational structures, MongoDB can store the business subject in the minimal number of documents. For example, instead of storing title and author information in two distinct relational structures, title, author, and other title-related information can all be stored in a single document called Book, which is much more intuitive and usually easier to work with.^[5]

Ad hoc queries

MongoDB supports search by field, range queries, regular expression searches. Queries can return specific fields of documents and also include user-defined JavaScript functions.

Indexing

Any field in a MongoDB document can be indexed (indices in MongoDB are conceptually similar to those in RDBMSes). Secondary indices are also available.

Replication

MongoDB provides high availability with replica sets.^[6] A replica set consists of two or more copies of the data. Each replica set member may act in the role of primary or secondary replica at any time. The primary replica performs all writes and reads by default. Secondary replicas maintain a copy of the data on the primary using built-in replication. When a primary replica fails, the replica set automatically conducts an election process to determine which secondary should become the primary. Secondaries can also perform read operations, but the data is eventually consistent by default.

Load balancing

MongoDB scales horizontally using sharding.^[7] The user chooses a shard key, which determines how the data in a collection will be distributed. The data is split into ranges (based on the shard key) and distributed across multiple shards. (A shard is a master with one or more slaves.)

MongoDB can run over multiple servers, balancing the load and/or duplicating data to keep the system up and running in case of hardware failure. Automatic configuration is easy to deploy, and new machines can be added to a running database.

File storage

MongoDB can be used as a file system, taking advantage of load balancing and data replication features over multiple machines for storing files.

This function, called GridFS,^[8] is included with MongoDB drivers and available with no difficulty for development languages (see "Language Support" for a list of supported languages). MongoDB exposes functions for file manipulation and content to developers. GridFS is used, for example, in plugins for NGINX^[9] and lighttpd.^[10] Instead of storing a file in a single document, GridFS divides a file into parts, or chunks, and stores each of those chunks as a separate document.^[11]

In a multi-machine MongoDB system, files can be distributed and copied multiple times between machines transparently, thus effectively creating a load-balanced and fault-tolerant system.

Aggregation

MapReduce can be used for batch processing of data and aggregation operations. The aggregation

framework enables users to obtain the kind of results for which the SQL GROUP BY clause is used.

Server-side JavaScript execution

JavaScript can be used in queries, aggregation functions (such as MapReduce), and sent directly to the database to be executed.

Capped collections

MongoDB supports fixed-size collections called capped collections. This type of collection maintains insertion order and, once the specified size has been reached, behaves like a circular queue.

Criticisms

In some failure scenarios where an application can access two distinct MongoDB processes, but these processes cannot access each other, it is possible for MongoDB to return stale reads. In this scenario it is also possible for MongoDB to acknowledge writes that will be rolled back.^[12]

Prior to November 2012, MongoDB's default consistency model ("write concern") acknowledged writes as soon as they had entered the client's outgoing queue,^[13] meaning that the default setup was vulnerable to client crashes.

MongoDB uses a readers-writer lock that allows concurrent read access to a database but exclusive write access to a single write operation. Before version 2.2, this lock was implemented on a per-mongod basis. Since version 2.2, the lock has been implemented at the database level.^[14] Since version 2.8, which later become 3.0,^[15] pluggable storage engines were introduced.^[16] Based on the storage engine the lock has been implemented on collection or on document level (document level is the entity of the isolation on write operations). With versions prior to that version "2.8", one approach to increase concurrency is to use sharding.^[17] In some situations, reads and writes will yield their locks. If MongoDB predicts a page is unlikely to be in memory, operations will yield their lock while the pages load. The use of lock yielding expanded greatly in 2.2.^[18]

Another criticism is related to the limitations of MongoDB when used on 32-bit systems.^[19] In some cases, this was due to inherent memory limitations.^[20] MongoDB recommends 64-bit systems and that users provide sufficient RAM for their working set. Some users encounter issues when their working set exceeds available RAM and the system encounters page faults. Compose, a provider of managed MongoDB infrastructure, recommends a scaling checklist for large systems.^[21]

Additionally, MongoDB does not support collation-based sorting and is limited to byte-wise comparison via memcmp,^[22] which will not provide correct ordering for many non-English languages^[23] when used with a Unicode encoding.

Language support

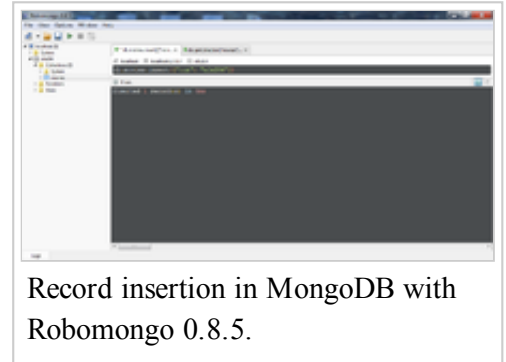
MongoDB has official drivers for a variety of popular programming languages and development environments.^[24] There are also a large number of unofficial or community-supported drivers for other programming languages and frameworks.^[25]

Management and graphical front-ends

Third-Party GUI tools

There is an active and growing community of developers building third-party rich GUI tools for the MongoDB. Some relevant examples (listed in alphabetical order):

- 3T MongoChef, cross-platform MongoDB GUI
- BI Studio, Business Intelligence frontend for MongoDB.
- Database Master, web-based client software, supports RDMS
- Fang of Mongo, web-based UI, built with Django and jQuery
- FusionReactor, Java based Application Monitor for MongoDB
- Futon4Mongo, a clone of the CouchDB-Futon-Web-Interface for MongoDB
- Humongous.io, Web based GUI for MongoDB
- mms, Mongo Management Studio, cross-platform and web-based GUI
- Mongo3, ruby-based GUI
- MongoHub, a native OS-X-application for MongoDB management
- NoSQL Manager for MongoDB, a MS Windows GUI application for MongoDB management with Shell
- Opricot, browser-based MongoDB-Shell, implemented in PHP
- Robomongo, Shell-centric cross-platform MongoDB management tool
- SlamData, enables running SQL queries on a MongoDB database
- UMongo (JMongoBrowser), cross-platform Management-GUI, implemented in Java



Record insertion in MongoDB with Robomongo 0.8.5.

Popularity

As of June 2015, MongoDB is the fifth most popular type of database management system, and the most popular for document stores.^[2]

Production deployments

Large-scale deployments of MongoDB are tracked by MongoDB Inc. Notable users of MongoDB include:

- Amadeus IT Group uses MongoDB for its back-end software.
- The Compact Muon Solenoid at CERN uses MongoDB as the primary back-end for the *Data Aggregation System* for the Large Hadron Collider.^[26]
- eBay uses MongoDB in the search suggestion and the internal Cloud Manager *State Hub*.^[27]
- Foursquare deploys MongoDB on Amazon AWS to store venues and user check-ins into venues.^[28]
- LinkedIn uses MongoDB as their backend DB.
- MetLife uses MongoDB for "The Wall", a customer service application providing a "360-degree view" of MetLife customers.^[29]
- SAP uses MongoDB in the SAP PaaS.^[30]
- Shutterfly uses MongoDB for its photo platform. As of 2013, the photo platform stores 18 billion photos uploaded by Shutterfly's 7 million users.^{[31][32]}
- Sophos uses MongoDB in their cloud security technology.
- Sourceforge uses MongoDB for its back-end storage pages.^[33]

- Talentica Software uses MongoDB in Ad-serving platform for one of their client.

See also

- V8 (JavaScript engine)
- Node.js
- SpiderMonkey (software)
- Rhino (JavaScript engine)
- Server-side scripting
- MEAN, a solutions stack using MongoDB as the database
- HyperDex, a NoSQL database providing the MongoDB API with stronger consistency guarantees
- OrientDB, a Document-Graph Multi-model database

References

1. "10gen embraces what it created, becomes MongoDB Inc." (<http://gigaom.com/2013/08/27/10gen-embraces-what-it-created-becomes-mongodb-inc/>). *Gigaom*. Retrieved 27 August 2013.
2. "Popularity ranking of database management systems" (<http://db-engines.com/en/ranking>). *db-engines.com*. Solid IT. Retrieved 16 June 2015.
3. The MongoDB NoSQL Database Blog, The AGPL (<http://blog.mongodb.org/post/103832439/the-agpl>)
4. MongoDB Developer Manual (<http://www.mongodb.org/display/DOCS/Manual>)
5. Data Modeling for MongoDB (<http://www.technicspub.com/product.sc?jsessionid=2C874BF93727F3F4751DDFF7B0105B0B.m1plqscsfapp04?productId=73&categoryId=1>)
6. [1] (<http://docs.mongodb.org/manual/core/replication-introduction/>)
7. [2] (<http://docs.mongodb.org/manual/sharding/>)
8. GridFS article on MongoDB Developer's Manual (<http://www.mongodb.org/display/DOCS/GridFS>)
9. NGINX plugin for MongoDB source code (<http://github.com/mdirolf/nginx-gridfs>)
10. lighttpd plugin for MongoDB source code (<http://bitbucket.org/bwmcadams/lighttpd-gridfs/src/>)
11. Expertstown - MongoDB overview (<http://www.expertstown.com/mongodb-overview/>)
12. Call me maybe: MongoDB stale reads (<https://aphyr.com/posts/322-call-me-maybe-mongodb-stale-reads>)
13. "Default Write Concern Change" (<http://docs.mongodb.org/manual/release-notes/drivers-write-concern/>). *MongoDB Release Notes*. Retrieved April 17, 2014.
14. FAQ Concurrency - How Granular Are Locks (<http://docs.mongodb.org/manual/faq/concurrency/#how-granular-are-locks-in-mongodb>)
15. Eliot Horowitz (2015-01-22). "Renaming Our Upcoming Release to MongoDB 3.0" (http://www.mongodb.com/blog/post/renaming-our-upcoming-release-mongodb-30?_ga=1.2592691.2020943709.1424661872). Retrieved 2015-02-23.
16. [3] (<http://blog.mongodb.org/post/102461818738/announcing-mongodb-2-8-0-rc0-release-candidate-and>) - MongoDB 2.8 release
17. FAQ Concurrency - How Does Sharding Affect Concurrency (<http://docs.mongodb.org/manual/faq/concurrency/#how-does-sharding-affect-concurrency>)
18. FAQ Concurrency - Do Operations Ever Yield the Lock (<http://docs.mongodb.org/manual/faq/concurrency/#does-a-read-or-write-operation-ever-yield-the-lock>)
19. 32-bit Limitations (<http://blog.mongodb.org/post/137788967/32-bit-limitations>)
20. Does Everybody Hate MongoDB (<https://blog.serverdensity.com/does-everyone-hate-mongodb/>)
21. [4] (<https://blog.compose.io/mongodb-scaling-to-100gb-and-beyond/>)
22. "memcmp" (<http://en.cppreference.com/w/c/string/byte/memcmp>). *cppreference.com*. 31 May 2013. Retrieved 26 April 2014.
23. MongoDB Jira ticket 1920 (<https://jira.mongodb.org/browse/SERVER-1920>)
24. "MongoDB Drivers and Client Libraries" (<http://www.mongodb.org/display/DOCS/Drivers>). *Mongodb.org*. Retrieved 2013-07-08.

Retrieved 2015-07-08.

25. "Community Supported Drivers" (<http://docs.mongodb.org/ecosystem/drivers/community-supported-drivers/>). Mongodb.org. Retrieved 2014-07-09.
26. Holy Large Hadron Collider, Batman! (<http://blog.mongodb.org/post/660037122/holy-large-hadron-collider-batman>)
27. MongoDB at eBay (<http://www.slideshare.net/mongodb/mongodb-at-ebay>)
28. Experiences Deploying MongoDB on AWS (<http://www.10gen.com/presentations/experiences-deploying-mongodb-aws>)
29. "Metlife uses nosql for customer service" (<http://www.informationweek.com/software/information-management/metlife-uses-nosql-for-customer-service/240154741>). Information Week. Retrieved 8 November 2014.
30. The Quest to Understand the Use of MongoDB in the SAP PaaS (<http://scn.sap.com/people/richard.hirsch/blog/2011/09/30/the-quest-to-understand-the-use-of-mongodb-in-the-sap-paas>)
31. Real World NoSQL: MongoDB at Shutterfly (<http://gigaom.com/2011/01/28/real-world-nosql-mongodb-at-shutterfly/>)
32. Here's How We Think Of Shutterfly's Stock Value (<http://seekingalpha.com/article/1241641-heres-how-we-think-of-shutterflys-stock-value>)
33. Scaling SourceForge with MongoDB (<http://www.oscon.com/oscon2010/public/schedule/detail/13669>)

Bibliography

- Hoberman, Steve (June 1, 2014), *Data Modeling for MongoDB* (1st ed.), Technics Publications, p. 226, ISBN 978-1-935504-70-2
- Banker, Kyle (March 28, 2011), *MongoDB in Action* (1st ed.), Manning, p. 375, ISBN 978-1-935182-87-0
- Chodorow, Kristina; Dirolf, Michael (September 23, 2010), *MongoDB: The Definitive Guide* (1st ed.), O'Reilly Media, p. 216, ISBN 978-1-4493-8156-1
- Pirtle, Mitch (March 3, 2011), *MongoDB for Web Development* (1st ed.), Addison-Wesley Professional, p. 360, ISBN 978-0-321-70533-4
- Hawkins, Tim; Plugge, Eelco; Membrey, Peter (September 26, 2010), *The Definitive Guide to MongoDB: The NoSQL Database for Cloud and Desktop Computing* (1st ed.), Apress, p. 350, ISBN 978-1-4302-3051-9

External links

- Official website (<http://www.mongodb.org/>)

Retrieved from "https://en.wikipedia.org/w/index.php?title=MongoDB&oldid=668579827"

Categories: Free database management systems | Document-oriented databases

| Distributed computing architecture | Structured storage | NoSQL | Database-related software for Linux
| 2009 software

-
- This page was last modified on 25 June 2015, at 06:31.
 - Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.