

Linux服务器构建与运维管理

从基础到实战（基于 openEuler）

第7章：使用MongoDB实现数据库服务

阮晓龙

13938213680 / ruanxiaolong@hactcm.edu.cn

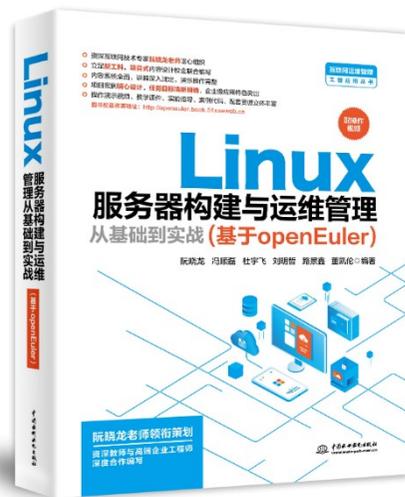
<https://internet.hactcm.edu.cn>
<http://www.51xueweb.cn>

河南中医药大学信息技术学院互联网技术教学团队
河南中医药大学医疗健康信息工程技术研究所

2024.10

提纲

- 了解MongoDB
 - 非关系型数据库
 - CAP、ACID、BASE
 - MongoDB
- 实现MongoDB数据库服务
 - 单机模式的数据库服务
 - 集群模式的数据库服务
- 管理与监控
 - MongoDB Database Tools
 - MongoDB Shell
 - MongoDB Compass
 - Percona Monitoring and Management



1. 了解MongoDB

1.1 非关系型数据库 (No-SQL DBMS)

- 互联网具有数据规模庞大和数据结构动态化等特点：
 - 关系型数据库在处理此类问题时不仅麻烦而且性能方面也达不到要求。
 - 非关系型数据库（简称“NoSQL”）**最初是**为了满足互联网的业务需求而诞生的，其抛弃了关系型数据库的强制一致性和事务等特性。
 - 关系型数据库的特性：
 - 支持复杂查询
 - 支持标准的SQL语言
 - 数据完整性高
 - 非关系型数据库的特性：
 - 存储的伸缩性更强
 - 数据操作的并发性能更强
 - 更容易通过多节点部署提高可用性
 - 数据模型更加灵活



5分钟解释关系型 和非关系型数据库

DB-Engines - Knowledge Base of ...

https://db-engines.com/en/

DB-ENGINES • Hard Real-Time • Embedded • Cloud/OnPrem

extremeDB Data management wherever you need it. [Learn more](#)

English Deutsch

Knowledge Base of Relational and NoSQL Database Management Systems provided by [Redgate Software](#)

Home DB-Engines Ranking Systems Encyclopaedia Blog Sponsors Search Vendor Login

Featured Products: [Neo4j](#) [Datastax Astra](#) [RaimaDB](#) [Milvus](#) [SingleStore](#)

DB-Engines

DB-Engines is an initiative to collect and present information on database management systems (DBMS). In addition to established relational DBMS, systems and concepts of the growing NoSQL area are emphasized.

The DB-Engines Ranking is a list of DBMS ranked by their current popularity. The list is updated monthly.

The most important properties of numerous systems are shown in the overview of database management systems. You can examine the properties for each system, and you can compare them side by side.

In the database encyclopedia terms and concepts on this topic are explained.

DB-Engines is maintained by [Redgate Software](#).

If you have suggestions, questions or general feedback, we would appreciate to get a [message](#) from you.

The latest news

[Are database management tasks slowing you down? Redgate Software wants to hear from you](#)
2 September 2024, Tim Dalton

[DB-Engines now owned by Redgate Software](#)
18 June 2024, Tom Russell

[PostgreSQL is the DBMS of the Year 2023](#)
2 January 2024, Matthias Gelbmann, Paul Andlinger

The most popular database management systems

October 2024	Score
1. Oracle	1309
2. MySQL	1023
3. Microsoft SQL Server	802
4. PostgreSQL	652
5. MongoDB	405

[» more](#)

Recent DB-Engines citations in the news

[Most popular database management systems worldwide 2024](#)
19 June 2024, Statista

[Redgate Software acquires DB-Engines](#)
25 June 2024, Business Weekly

[Redgate Software announces acquisition of DB-Engines](#)
18 June 2024, Cambridge Network

provided by Google News

Upcoming events

[Microsoft SQL Server event](#)
[PASS Data Community Summit](#)
Seattle, Washington
4-8 November 2024

[PostgreSQL event](#)
[PASS Data Community Summit](#)
Seattle, Washington
4-8 November 2024

[» more DBMS events](#)

Featured Products

neo4j

See for yourself how a graph database can make your life easier.
[Use Neo4j online for free.](#)

RAIMA

[RaimaDB](#), embedded database for mission-critical applications. When performance, footprint and reliability matters.
[Try RaimaDB for free.](#)

milvus

Vector database designed for GenAI, fully equipped for enterprise implementation.
[Try Managed Milvus for Free](#)

SingleStore

The data platform to build your intelligent applications.
[Try it free.](#)

DATASTAX ASTRA

Bring all your data to [Generative AI](#) applications with [vector search](#) enabled by the most scalable [vector database](#) available.
[Try for Free](#)

[Present your product here](#)

DB-Engines - Knowledge Base of ...

https://db-engines.com/en/systems

DB-ENGINES Problem: IoT connectivity disruptions. Solution: extremeDB [Learn how](#)

English Deutsch

Knowledge Base of Relational and NoSQL Database Management Systems provided by [Redgate Software](#)

Home DB-Engines Ranking Systems Encyclopaedia Blog Sponsors Search Vendor Login

Featured Products: [Milvus](#) [Neo4j](#) [RaimaDB](#) [SingleStore](#) [Datastax Astra](#)

Database management systems

487 database management systems, 423 currently in the [DB-Engines Ranking](#) and 64 discontinued. Please select a system to see its properties.

1010data 4D 4store Acebase Actian FastObjects Actian NoSQL Database Actian PSQL Actian Vector ActorDB Adabas Aerospike AgensGraph Akiban AlaSQL Algebraix Algolia Alibaba Cloud AnalyticDB for MySQL + Alibaba Cloud AnalyticDB for PostgreSQL Alibaba Cloud Log Service + Alibaba Cloud MaxCompute Alibaba Cloud PolarDB + Alibaba Cloud Table Store Alibaba Cloud TSDB AllegroGraph Altibase Amazon Aurora Amazon CloudSearch Amazon DocumentDB Amazon DynamoDB + Amazon Keyspaces Amazon Neptune Amazon Redshift + Amazon SimpleDB Amazon Timestream Amisa Server AntDB AnzoGraph DB Apache Accumulo Apache Cassandra + Apache Derby Apache Doris Apache Drill Apache Druid Apache Flink Apache HAWQ Apache HBase	GenieDB Geode GeoMesa GeoSpock GigaSpaces Giraph GlobalsDB Google BigQuery + Google Cloud Bigtable Google Cloud Datastore Google Cloud Firestore Google Cloud Spanner Google Search Appliance Graph Engine GraphBase GraphDB + Graphite Greenplum GreptimeDB + GridDB GridGain gStore GT.M H2 H2GIS Hadapt HarperDB Hawkuar Metrics Hazelcast HEAVY.AI Helium Heroic HFSQL HGraphDB Hibari HPE Ezmeral Data Fabric HyperDex HyperGraphDB HyperLevelDB HyperSQL Hypertable Hyrcubd IBM Cloudant IBM Db2 IBM Db2 Event Store IBM Db2 warehouse	ParAccel Percona Server for MongoDB Percona Server for MySQL Perst PieCloudDB + Pinecone PipelineDB PlanetScale PostGIS Postgres-XL PostgreSQL + PouchDB Presto Project Voldemort Prometheus Qdrant Quasardb QuestDB R:BASE Raima Database Manager + Rainstor RaptorDB Rasdaman RavenDB RDF4J RDFox Realm Red Brick Redis + Redland RedStore ReductStore Resin Cache RethinkDB Riak KV Riak TS RisingWave Rizhiyi RocksDB Rockset RRDtool Sadas Engine SAP Adaptive Server SAP Advantage Database Server SAP HANA + SAP IQ
---	---	---

Featured Products

SingleStore

The data platform to build your intelligent applications.
[Try it free.](#)

neo4j

See for yourself how a graph database can make your life easier.
[Use Neo4j online for free.](#)

milvus

Vector database designed for GenAI, fully equipped for enterprise implementation.
[Try Managed Milvus for Free](#)

DATASTAX ASTRA

Bring all your data to [Generative AI](#) applications with [vector search](#) enabled by the most scalable [vector database](#) available.
[Try for Free](#)

[Present your product here](#)

DB-Engines Ranking - popularity x +

https://db-engines.com/en/ranking

DB-ENGINES Problem: IoT connectivity disruptions. Solution: extremeDB Learn how

Knowledge Base of Relational and NoSQL Database Management Systems provided by Redgate Software

Home DB-Engines Ranking Systems Encyclopedia Blog Sponsors Search Vendor Login

Featured Products: [RaimaDB](#) [SingleStore](#) [Neo4j](#) [Milvus](#) [Datastax Astra](#)

Select a ranking

- Complete ranking
- Relational DBMS
- Key-value stores
- Document stores
- Time Series DBMS
- Graph DBMS
- Search engines
- Object oriented DBMS
- RDF stores
- Vector DBMS
- Wide column stores
- Multivalued DBMS
- Spatial DBMS
- Native XML DBMS
- Event Stores
- Content stores
- Navigational DBMS
- Columnar

Special reports

- Ranking by database model
- Open source vs. commercial

Featured Products

Milvus
Vector database designed for GenAI, fully equipped for enterprise implementation. Try Managed Milvus for Free

Datastax Astra
Bring all your data to Generative AI applications with vector search enabled by the most scalable vector database available. Try for Free

Raima
RaimaDB, embedded database for mission-critical applications.

Ranking > Complete Ranking RSS RSS Feed

DB-Engines Ranking

The DB-Engines Ranking ranks database management systems according to their popularity. The ranking is updated monthly.

Read more about the [method](#) of calculating the scores.

423 systems in ranking, October 2024

Rank	Oct 2024	Sep 2024	Oct 2023	DBMS	Database Model	Score	Oct 2024	Sep 2024	Oct 2023
1.	1.	1.	1.	Oracle	Relational, Multi-model	1309.45	+22.85	+48.03	
2.	2.	2.	2.	MySQL	Relational, Multi-model	1022.76	-6.73	-110.56	
3.	3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	802.09	-5.67	-94.79	
4.	4.	4.	4.	PostgreSQL	Relational, Multi-model	652.16	+7.80	+13.34	
5.	5.	5.	5.	MongoDB	Document, Multi-model	405.21	-5.02	-26.21	
6.	6.	6.	6.	Redis	Key-value, Multi-model	149.63	+0.20	-13.33	
7.	7.	11.	7.	Snowflake	Relational	140.60	+6.88	+17.36	
8.	8.	7.	8.	Elasticsearch	Multi-model	131.85	+3.06	-5.30	
9.	9.	8.	8.	IBM Db2	Relational, Multi-model	122.77	-0.28	-12.10	
10.	10.	9.	9.	SQLite	Relational	101.91	-1.43	-23.23	
11.	11.	12.	12.	Apache Cassandra	Wide column, Multi-model	97.61	-1.34	-11.21	
12.	12.	10.	10.	Microsoft Access	Relational	92.15	-1.61	-32.16	
13.	13.	14.	14.	Splunk	Search engine	91.27	-1.75	-1.10	
14.	14.	17.	17.	Databricks	Multi-model	85.60	+1.35	+9.78	
15.	15.	13.	13.	MariaDB	Relational, Multi-model	84.89	+1.45	-14.77	
16.	16.	15.	15.	Microsoft Azure SQL Database	Relational, Multi-model	74.53	+1.58	-6.40	
17.	17.	16.	16.	Amazon DynamoDB	Multi-model	71.85	+1.78	-9.07	
18.	18.	18.	18.	Apache Hive	Relational	52.57	-0.50	-16.61	
19.	19.	20.	20.	Google BigQuery	Relational	51.18	-1.48	-5.39	
20.	20.	21.	21.	FileMaker	Relational	44.40	-0.80	-8.92	
21.	21.	23.	23.	Neo4j	Graph	42.51	-0.17	-5.93	
22.	23.	22.	22.	SAP HANA	Relational, Multi-model	41.21	-0.11	-8.23	
23.	22.	19.	19.	Teradata	Relational, Multi-model	40.69	-0.78	-17.87	
24.	24.	24.	24.	Apache Solr	Search engine, Multi-model	32.98	-1.17	-12.38	
25.	25.	25.	25.	SAP Adaptive Server	Relational, Multi-model	32.40	-0.27	-9.52	
26.	26.	26.	26.	Apache HBase	Wide column	27.19	-0.21	-7.50	
27.	27.	27.	27.	Microsoft Azure Cosmos DB	Multi-model	24.50	-0.47	-9.80	
28.	28.	28.	28.	InfluxDB	Time Series, Multi-model	22.39	+0.27	-7.35	
29.	29.	29.	29.	PostGIS	Spatial, Multi-model	20.56	+0.40	-6.74	
30.	30.	31.	31.	Firebird	Relational	19.02	-0.44	-6.66	
31.	31.	30.	30.	Microsoft Azure Synapse Analytics	Relational	18.72	-0.26	-8.41	
32.	32.	35.	35.	Memcached	Key-value	17.79	+0.95	-3.05	

DB-Engines Ranking - popularity x +

https://db-engines.com/en/ranking-trend

DB-ENGINES Problem: IoT connectivity disruptions. Solution: extremeDB Learn how

Knowledge Base of Relational and NoSQL Database Management Systems provided by Redgate Software

Home DB-Engines Ranking Systems Encyclopedia Blog Sponsors Search Vendor Login

Featured Products: [Datastax Astra](#) [Neo4j](#) [SingleStore](#) [Milvus](#) [RaimaDB](#)

Select a ranking

- Complete ranking
- Relational DBMS
- Key-value stores
- Document stores
- Time Series DBMS
- Graph DBMS
- Search engines
- Object oriented DBMS
- RDF stores
- Vector DBMS
- Wide column stores
- Multivalued DBMS
- Spatial DBMS
- Native XML DBMS
- Event Stores
- Content stores
- Navigational DBMS
- Columnar

Special reports

- Ranking by database model
- Open source vs. commercial

Featured Products

SingleStore
The data platform to build your intelligent applications. Try it free.

Datastax Astra
Bring all your data to Generative AI applications with vector search enabled by the most scalable vector database available. Try for Free

Raima
RaimaDB, embedded database for mission-critical applications.

Ranking > Trend RSS RSS Feed

DB-Engines Ranking - Trend Popularity

The DB-Engines Ranking ranks database management systems according to their popularity.

Read more about the [method](#) of calculating the scores.

DB-Engines Ranking

Score (logarithmic scale)

Legend: Oracle, MySQL, Microsoft SQL Server, PostgreSQL, MongoDB, Redis, Snowflake, Elasticsearch, IBM Db2, SQLite, Apache Cassandra, Microsoft Access, Splunk, Databricks, MariaDB, Microsoft Azure SQL Database, Amazon DynamoDB, Apache Hive, Google BigQuery, FileMaker, Neo4j, SAP HANA, Teradata, Apache Solr, SAP Adaptive Server, Apache HBase, AsterixDB

Click at a system in the legend to hide or show its trend line

Upcoming events

[PostgreSQL event](#)
PASS Data Community Summit
Seattle, Washington
4-8 November 2024

[Microsoft SQL Server event](#)
PASS Data Community Summit
Seattle, Washington
4-8 November 2024

Share this page

[About Us](#) [Advertising and Services](#) [Privacy Policy](#) [Contact](#)

1. 了解MongoDB

1.1 非关系型数据库 (No-SQL DBMS)

- 非关系型数据库是相对于关系型数据库来讲的，不遵循二维数据模型。
- 非关系型数据库具备的通用特点如下：
 - 高性能
 - 分布式
 - 易扩展
 - 不支持事务
- 非关系型数据库具备的适用场景如下：
 - 数据模型比较简单
 - 需要灵活性更强的IT系统
 - 对数据库性能要求较高
 - 不需要高度的数据一致性



1. 了解MongoDB

1.1 非关系型数据库 (No-SQL DBMS)

- 非关系型数据库没有统一架构，通常具有较强的应用场景适应性，不同应用场景下应选用不同的产品。
- 非关系型数据库常见的有键值数据库、列族数据库、文档数据库和图形数据库。

分类	相关产品	应用场景	数据模型	优点	缺点
键值数据库	Redis、Memcached、GaussDB(for Redis)、TcaplusDB	内容缓存、频繁读写	<key,value> 键值对，通过散列表实现	大量操作时性能高	数据无结构化
列族数据库	HBase、Cassandra	分布式数据存储与管理	以列族式存储，将同一列数据存储在一起	查找速度快、复杂性低	功能局限，不支持事务的强一致性
文档数据库	MongoDB、Elasticsearch、GaussDB(for Mongo)、GaussDB(for Influx)	Web 应用、面向文档或半结构化的数据	<key,value>，value 是 JSON 结构的文档	数据结构灵活	缺乏统一查询语法
图形数据库	Neo4j、AllegroGraph	推荐系统、构建关系图谱	图结构	支持复杂的图形算法	复杂性高，只能支持一定的数据规模

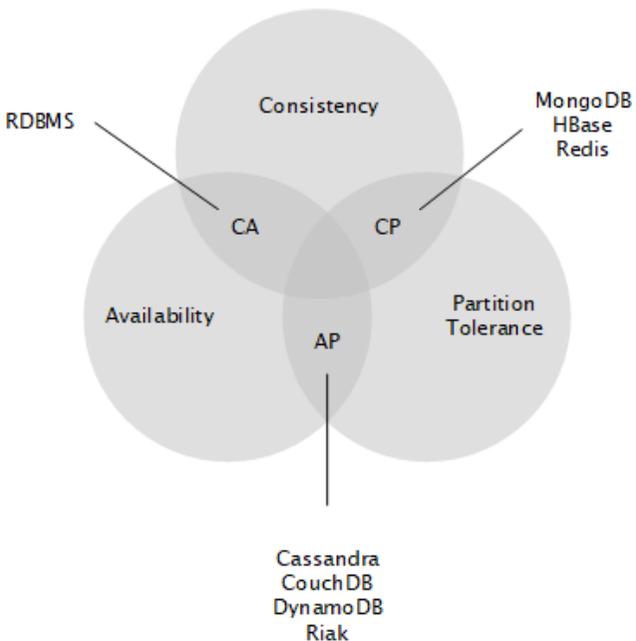


1. 了解MongoDB

1.2 CAP、ACID、BASE

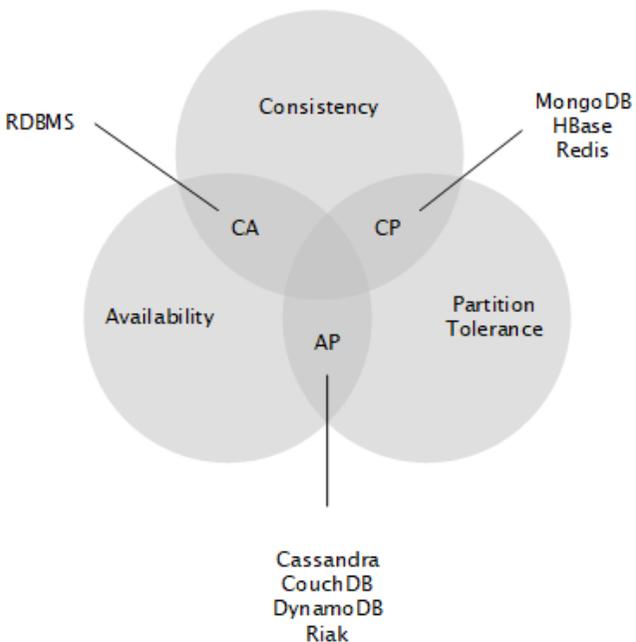
- CAP理论是由Eric Brewer在2001年提出的，指出对于一个分布式计算系统来说，不可能同时满足以下三点。
 - 一致性（Consistency）
 - 一致性是指更新操作成功后，所有节点在同一时间的数据完全一致。
 - 可用性（Availability）
 - 可用性是指用户访问数据时，系统是否能在正常响应时间返回结果。
 - 分区容错性（Partition Tolerance）
 - 分区容错性是指分布式系统在遇到某节点或网络分区故障的时候，仍然能够对外提供满足一致性和可用性的服务。





选择	特征	实例
一致性+可用性	2PC缓存验证	单机数据库、集群数据库、LDAP、xFS
一致性+分区容忍性	乐观锁	分布式系统、分布式锁、大部分的协议
可用性+分区容忍性	冲突解决, 乐观化	Coda (分布式档案系统)、网络缓存、DNS





选择	例子	对应功能分类
一致性+可用性	MySQL	关系型数据库
	Vertica	列存储数据库
一致性+分区容忍性	BigTable	列存储数据库
	HBase	列存储数据库
	MongoDB	文档型数据库
可用性+分区容忍性	Dynamo	键值存储数据库
	Cassandra	列存储数据库



1. 了解MongoDB

1.2 CAP、ACID、BASE

- 关系型数据库支持事务的ACID特性，即原子性、一致性、隔离性和持久性，这4种特性保证在事务过程中数据的正确性，具体描述如下。
 - 原子性 (Atomicity) 。
 - 一个事务的所有操作步骤被看成一个动作，所有的步骤要么全部完成，要么一个也不会完成。如果在事务过程中发生错误，则会回滚到事务开始前的状态，将要被改变的数据库记录不会被改变。
 - 一致性 (Consistency) 。
 - 一致性是指在事务开始之前和事务结束以后，数据库的完整性约束没有被破坏，即数据库事务不能破坏关系数据的完整性及业务逻辑上的一致性。
 - 隔离性 (Isolation) 。
 - 主要用于实现并发控制，隔离能够确保并发执行的事务按顺序一个接一个地执行。通过隔离，一个未完成事务不会影响另外一个未完成事务。
 - 持久性 (Durability) 。
 - 一旦一个事务被提交，它应该持久保存，不会因为与其他操作冲突而取消这个事务。
 - 从事务的4个特性可以看出，关系型数据库要求强一致性，但是这一点在非关系型数据库中是重点弱化的机制。这是因为数据库保持强一致性时，很难保证系统具有横向扩展和可用性的优势，因此，针对分布式数据存储管理只提供了弱一致性的保障。



1. 了解MongoDB

1.2 CAP、ACID、BASE

- BASE是对CAP理论中一致性和可用性进行权衡的结果，核心是无法做到强一致性，但每个应用都可以根据自身的特点，采用适当方式达到最终一致性。
- 一般来说，非关系型数据库都支持BASE原理。
 - 基本可用（Basically Available）。
 - 基本可用指分布式系统在出现故障时，系统允许损失部分可用性，即保证核心功能或者当前最重要功能可用。
 - 软状态（Soft-state）。
 - 软状态允许数据存在中间状态，但不会影响系统的整体可用性，即允许不同节点的副本之间存在暂时的一致情况。
 - 最终一致性（Eventually Consistent）。
 - 最终一致性要求系统中数据副本最终能够一致，而不需要实时保证数据副本一致。最终一致性是BASE原理的核心，也是非关系型数据库的主要特点，通过弱化一致性，提高系统的可伸缩性、可靠性和可用性。
 - 从以上可以看出，关系型数据库与非关系型数据库各有特点，对于数据库的选型应与自己的业务结合，充分考量。



关系型与非关系型数据库分类和特性对比

数据库类型	特征	优点	缺点
关系型数据库	<ol style="list-style-type: none">1、关系型数据库的最大特点就是事务的一致性2、简单来说，关系模型指的就是二维表格模型	<ol style="list-style-type: none">1、容易理解：二维表结构是非常容易理解2、使用方便：支持SQL，可用于复杂查询3、易于维护：丰富的完整性大大降低了数据冗余和数据不一致的概率	<ol style="list-style-type: none">1、为了维护一致性所付出的巨大代价就是其读写性能比较差2、固定的表结构3、高并发读写需求4、海量数据的高效率读写
非关系型数据库	<ol style="list-style-type: none">1、使用键值对存储数据2、分布式3、非关系型数据库严格上不是一种数据库，应该是一种数据结构化存储方法的集合	<ol style="list-style-type: none">1、无需经过sql层的解析，读写性能很高2、基于键值对，数据没有耦合性，易扩展	<ol style="list-style-type: none">1、不提供sql支持，学习和使用成本较高2、无事务处理，附加功能bi和报表等支持也不好

关系型与非关系型数据库各有特点，对于数据库的选型应与自业务结合，充分考量。



1. 了解MongoDB

- ❑ MongoDB是一个表结构自由、开源、可扩展、面向文档的数据库，旨在为Web应用程序提供高性能、高可用且易扩展的数据存储解决方案。
- ❑ MongoDB支持多文档事务、连接查询，是较为接近关系型数据库的非关系型数据库。
- ❑ MongoDB的主要特性：
 - 灵活的数据模型
 - 强大的查询语言
 - 提供多种编程语言的API
 - 易于扩展
 - 支持复制和故障自动转移



mongoDB



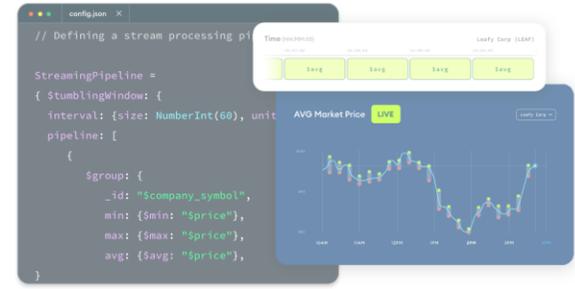
MONGODB ATLAS

深受开发者喜爱。 专为下列场景打造

流媒体数据

不需要单独的数据库来支持事务处理、多样化搜索或生成式 AI。世界上最流行的文档数据库现在也是世界上功能最全面的开发者数据平台。

[免费开始使用](#) [了解详情](#) →



- 49K+** 家客户 → 在 AWS, Azure 和 Google Cloud 等云平台上的可用地区 →
- 115+** 每月开发者加入的数量 →
- 175K+** 使用最多的现代数据库 →
- #1** 使用最多的现代数据库 →

Contact Us

MongoDB 的版本

MongoDB 是什么?

MongoDB 是一个文档数据库，为简化应用程序的开发与扩展而设计。

您可以在以下环境运行 MongoDB:

- [MongoDB Atlas](#) →: 用于云端 MongoDB 部署的完全托管服务
- [MongoDB Enterprise](#) →: 基于订阅、自行管理的 MongoDB 版本
- [MongoDB Community](#) →: source-available、可免费使用以及自行管理的 MongoDB 版本

开启 MongoDB Atlas 之旅



您能做什么

在 MongoDB 中处理您的数据

- 存储和查询您的数据
- 使用聚合转换数据
- 安全访问您的数据
- 部署和扩展数据库

1

部署 MongoDB

在 MongoDB Atlas 用户界面或 Atlas CLI 中快速轻松地创建集群。要了解更多信息，请参阅 MongoDB Atlas 文档中的[创建集群](#)和 Atlas CLI 文档中的[开始使用 Atlas](#)。

对于自托管部署，请参阅 MongoDB 手册中的[复制](#)部分来创建副本集。

2

连接到您的部署

在 MongoDB Atlas 用户界面中访问部署，或者根据 MongoDB 手册中的步骤，使用[驱动程序](#)或 [MongoDB Shell \(mongosh\)](#) 进行连接。

要了解更多信息，请参阅 MongoDB 手册中的[查找连接字符串](#)。

3

插入、查询、更新或删除文档

```
→ atlas setup
? Do you want to setup your Atlas database with default set
→ Y

We are deploying Cluster9876543...
Please store your database authentication access details in
Database User Username: Cluster9876543
Database User Password: abcdef12345
Creating your cluster... [Its safe to 'Ctrl + C']
```

NEW MongoDB 8.0 is available now

MongoDB 8.0

The most popular document database is faster than ever.



36%

faster reads

59%

higher throughput for updates

200%

faster time series aggregations

Up to 50x

faster resharding

New use cases for encrypted data

Queryable Encryption now supports range queries in MongoDB 8.0. This makes sensitive data easier to work with while

Faster, easier horizontal scaling

As applications grow, they scale past the limits of one server. In MongoDB 8.0, scaling is faster and easier, with less cost to get

Ensure resilience with new controls

MongoDB 8.0 now provides greater control for teams to optimize database performance for unpredictable spikes in usage.

MongoDB. 产品 资源 解决方案 公司 定价 简体中文 Support 登录 免费试用

MongoDB Atlas

适用于任何使用场景的开发者数据平台。

[了解更多](#)

向量搜索

统一数据库和向量存储。集成用于 AI 应用程序、聊天机器人、内容生成和高级语义搜索的大型语言模型 (LLM)。

[了解详情](#)

流处理

统一事件驱动的应用程序。使用一致的数据模型和查询界面无缝处理高速数据流。

[了解详情](#)

操作

利用多功能文档数据模型最大限度地提高写入性能。单个查询 API 可满足简单的查找和复杂的分析需求。

[了解详情](#)

事务处理

大规模毫秒级响应，具有灵活的数据模型和丰富的查询（包括二级索引），连接多文档 ACID。

[了解详情](#)

文本搜索

结合使用数据库、搜索引擎和同步功能，获取结果的速度提高 30-50%。启用目录、内容搜索和单一视图。

[了解详情](#)

分析

统一应用程序驱动的分析、实时聚合和转换。优化索引、存储和数据格式。

[了解详情](#)

图表

使用 MongoDB 的图表数据增强应用程序。查找各种模式获取见解。非常适合推荐、欺诈检测和网络。

地理空间

原生 GeoJSON 支持、简单坐标对和专用索引。专为物流、定位服务、空间分析和设计。

[Contact Us](#)

<https://www.mongodb.com/zh-cn/why-use-mongodb>

MongoDB. Products Resources Solutions Company Pricing Eng Support Sign In Try Free

I want to...

Model, build, and interact with data

- [Atlas](#)
Fully managed MongoDB in over 100+ regions on AWS, Google Cloud, and Azure
- [Atlas on AWS](#)
Fully managed MongoDB on AWS

Search, analyze, and visualize data

Migrate to MongoDB

[Run MongoDB](#)

Manage MongoDB programmatically

- [Atlas on Google Cloud](#)
Fully managed MongoDB on Google Cloud
- [Atlas on Microsoft Azure](#)
Fully managed MongoDB on Microsoft Azure
- [Atlas for Government](#)
Fully managed and dedicated MongoDB with FedRAMP Moderate Authorization
- [Community Edition](#)
A Community version of the MongoDB database that can be self-managed and hosted
- [Enterprise Advanced](#)
Self-managed MongoDB for on-premises and private cloud environments
- [Cluster to Cluster Sync](#)
Continuous data sync between separate MongoDB clusters

[Contact Us](#)

1. 了解MongoDB

1.4 MongoDB的数据逻辑结构

- MongoDB的数据逻辑结构与关系型数据库结构比较相似，都是三级存储结构，但它们最大的区别就是MongoDB中的集合是动态模式。

- **文档**：文档是MongoDB存储的元数据，它是由键值对组成的数据结构，其结构类似JSON对象，字段值可以包括其他文档、数组和文档数组，例如：

```
{
  name:'Su'
  age:'26',
  status:'A',
  groups:['news','sports' ]
}
```

- **集合**：MongoDB将文档存储在集合中，集合类似于关系数据库中的表。集合中的文档结构不需要相同，但为了管理方便和数据库的性能，应将相同类型的文档放在统一集合中。
- **上限集合**：集合的大小固定，当其达到最大时会自动覆盖最早插入的数据。
- **数据库**：多个集合组织在一起就是数据库。如表7-1-2所示展示了MongoDB与关系型数据库的逻辑结构对比。



MongoDB与关系型数据库的逻辑结构对比

MongoDB	关系型数据库
文档 (document)	行 (row)
集合 (collection)	表 (table)
数据库 (database)	数据库 (database)



2. 实现MongoDB数据库服务

2.1 单机模式的数据库服务



单机模式的数据库服务

任务目标:

- 安装MongoDB。
- 实现MongoDB服务管理。
- 实现MongoDB服务状态查看。

操作步骤:

- 通过RPM安装
- 配置存储库
- 通过在线方式安装
- 配置服务并测试

操作演示:

Code



2. 实现MongoDB数据库服务

2.2 MongoDB副本集

□ MongoDB集群

- 在MongoDB官方描述中，通常指副本集（Replica Set）或分片集群（Sharded Cluster）。
- 副本集是存储相同数据副本实例，是一组维护相同数据集的实例。
 - 节点（服务器）之间通过自动复制数据，以提供冗余和防止系统故障或计划内维护时停机的保护，确保高可用性。
 - 一个副本集包含多个数据承载节点和一个仲裁器（Arbiter，可选）。
 - 在数据承载节点中，只有一个成员被当作主节点，其他成员皆为从节点。
 - 副本集节点数最好为奇数，成员数最少为3个，不超过50个，最多7个投票成员。
- 分片集群是分布在许多分片（服务器）上的数据集的集合。
 - 在集合级别对数据进行分片，将集合中的文档分布在集群中的分片之间，以实现处理海量数据增长的水平可扩展性和更好的读写操作性能。



2. 实现MongoDB数据库服务

2.2 MongoDB副本集

□ 副本集中的成员

■ MongoDB 副本集中的成员可分为 3 种：主节点（Primary）、从节点（Secondaries）和仲裁器（Arbiter），每种成员都在副本集上起着不同的作用。

□ 主节点。

- 主节点是副本集中唯一能够接收写操作的成员。
- 副本集只能有一个主节点，如果当前的主节点不可用，则通过选举确定新的主节点。

□ 从节点。

- 从节点作为主节点数据集的副本，在副本集中起着数据备份、主节点候选人和负载均衡的作用。
- 尽管客户端无法通过从节点写入数据，但是客户端可以选择从节点读取数据。



2. 实现MongoDB数据库服务

- **副本集的从节点**从功能上可细分为3种属性：
 - 优先级为0的副本集成员（Priority 0 Replica Set Members）
 - 该成员的优先级为0（也作为备用成员），不可选举为主节点，但可投票，可驻留在辅助数据中心或充当冷备用数据库。
 - 隐藏副本集成员（Hidden Replica Set Members）
 - 该成员维护副本集的数据集且拥有选举投票权，但对客户端不可见，通常作数据备份节点。
 - 可阻止应用程序从中读取数据，这使它可以运行需要与正常流量隔离的应用程序。
 - 该成员的优先级为0，具有优先级为0的副本集成员的属性。
 - 延迟副本集成员（Delayed Replica Set Members）
 - 该成员所维护的数据集相对于正常成员总是有一段时间的延迟，保留运行中的“历史”快照，以用于从某些错误（例如意外删除的数据库）中恢复。
 - 该成员既是优先级为0的成员，以防止延迟成员成为主要成员；也是隐藏成员，始终阻止应用程序查看和查询延迟的成员。



2. 实现MongoDB数据库服务

□ 副本集的仲裁器

- 该成员没有数据副本，也不会成为主节点，主要用来选举投票。
- 当副本集的数据节点为偶数时，需要添加一个仲裁节点。
- 仲裁节点因为没有数据，只参与投票，所以仲裁节点需要的资源很少，但官方不建议将仲裁节点部署在副本集的其他节点上。



2. 实现MongoDB数据库服务

2.2 MongoDB副本集

- **副本集的选举。**
 - 副本集通过选举来决定哪个节点为主节点。
 - 以下事件可以触发副本集选举。
 - 向副本集添加新节点
 - 副本集初始化
 - 指定主节点为从节点或副本集重新配置
 - 主节点响应超时（默认10s）
 - 以下因素影响选举。
 - Heartbeats，副本集成员每两秒钟都会向彼此发送一次Heartbeats（类似ping）。
 - 如果某个成员在10s内未响应，则其他成员将其标记为不可访问，
 - 该成员将不能成为主节点或被降低优先级。
 - 优先级。优先级高的成员将优先获取投票权。
 - 票数。得票数最多的成员将成为主节点。



2. 实现MongoDB数据库服务

2.3 集群模式的数据库服务



集群模式的数据库服务

任务目标:

- 实现MongoDB replica set的部署。
- 完成MongoDB replica set的测试。

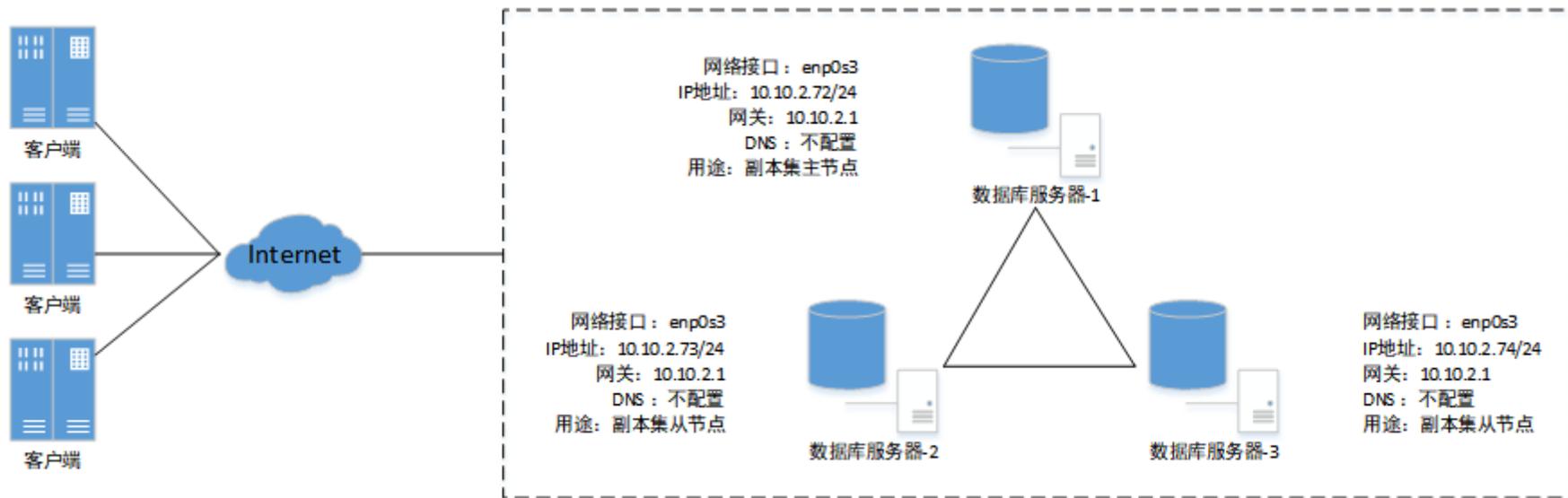
操作步骤:

- 设计方案
- 完成节点部署
- 初始化副本集
- 应用测试

操作演示:



集群模式的数据库服务拓扑结构



本任务配置MongoDB副本集的流程



3. 管理和监控

command-line utilities

**MongoDB
Database
Tools**

管理数据库服务器

**MongoDB
Shell**

操作数据内容

**MongoDB
Compass
(GUI)**

图形化管理软件

**Percona
Monitoring
and
Management**

第三方工具



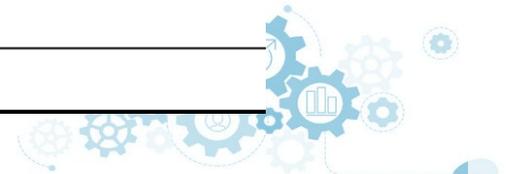
3. 管理和监控

3.1 MongoDB Database Tools

- MongoDB Database Tools是用于MongoDB部署和管理的命令行工具集。
 - 从MongoDB 4.4开始，MongoDB Database Tools与MongoDB服务器分开发布，并使用独立的版本控制，初始版本号为100.0.0。（推荐使用最新版本）
 - MongoDB Database Tools与MongoDB同时发布，有版本兼容性。（`mongodump --version`查看版本）

MongoDB Database Tools工具集列表

名称	类型	描述
mongodump	Binary Import / Export	以二进制格式导出数据
mongorestore	Binary Import / Export	将 mongodump 导出的数据进行转储还原
bsondump	Binary Import / Export	将 BSON 格式转换为 JSON
mongoimport	Data Import / Export	导入数据，支持 JSON、CSV 或 TSV 格式
mongoexport	Data Import / Export	导出数据，支持 JSON、CSV 格式
mongostat	Diagnostic Tools	查看单实例或集群的状态
mongotop	Diagnostic Tools	查看实例的读写耗时
mongofiles	GridFS Tools	从其他成员来看，该成员不可访问



- MongoDB Atlas
- MongoDB Enterprise Advanced
- MongoDB Community Edition
- Tools
 - MongoDB Shell
 - MongoDB Compass (GUI)
 - Atlas CLI
 - Atlas Kubernetes Operator
 - MongoDB CLI for Cloud Manager and Ops Manager
 - MongoDB Cluster-to-Cluster Sync
 - Relational Migrator
 - MongoDB Database Tools**
 - MongoDB Connector for BI
 - App Services CLI
 - Atlas SQL Interface
 - Mobile & Edge



TOOLS

MongoDB Command Line Database Tools Download

The MongoDB Database Tools are a collection of command-line utilities for working with a MongoDB deployment. These tools release independently from the MongoDB Server schedule enabling you to receive more frequent updates and leverage new features as soon as they are available. See the [MongoDB Database Tools](#) documentation for more information.

Version
100.10.0

Platform
Amazon Linux x86_64

Package
tgz

Download Copy link More Options



TOOLS

- MongoDB 数据库工具
 - mongodb
 - mongorestore
 - bsondump
 - mongoimport
 - mongoexport
 - mongostat
 - mongotop
 - mongofiles
 - 安装
 - 日志
 - 100.10.0 更改日志

Docs 主页

MongoDB Database Tools 文档

MongoDB 数据库工具是用于处理 MongoDB 部署的命令行实用程序集合。这些数据库工具包括以下二进制文件：

Binary Import / Export

- [mongodbump](#) 创建 [mongod](#) 数据库内容的二进制导出。
- [mongorestore](#) 将数据从 [mongodbump](#) 数据库转储恢复到 [mongod](#) 或 [mongos](#)
- [bsondump](#) 将 BSON 转储文件转换为 JSON。

Data Import / Export

- [mongoimport](#) 从 [Extended JSON](#)、[CSV](#) 或 [TSV](#) 导出文件导入内容。
- [mongoexport](#) 生成存储在 [mongod](#) 实例中的数据的 [JSON](#) 或 [CSV](#) 导出。

Diagnostic Tools

- [mongostat](#) 提供当前正在运行的 [mongod](#) 或 [mongos](#) 实例状态的整体情况速览。
- [mongotop](#) 概述 [mongod](#) 实例读取和写入数据所花费的时间。

GridFS Tools

- [mongofiles](#) 支持操作存储于 [GridFS](#) 对象的 MongoDB 实例中的文件。

您可以使用 Database Tools 从自托管部署迁移到 [MongoDB Atlas](#)。MongoDB Atlas 是用于云中 MongoDB 部署的完全托管服务。要了解详情，请参阅 [使用 mongorestore 播种](#)。

要了解迁移到 MongoDB Atlas 的所有方法，请参阅 [迁移或导入数据](#)。

版本控制

重要

建议您始终使用 MongoDB Database Tools 的最新版本。要检查 MongoDB Database Tools 的当前版本，请在终端上运行以下命令：

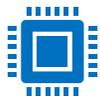
```
mongodbump --version
```

要下载最新版本，请访问 [MongoDB 下载中心](#)。

从 MongoDB 4.4 开始，MongoDB 数据库工具现在与 MongoDB 服务器分开发布，并使用自己的版本控制，初始版本为 100.0.0。以前，这些工具与 MongoDB 服务器一起发布，并使用匹配的版本控制。

3. 管理和监控

3.1 MongoDB Database Tools



```
mongostat <options> <connection-string> <polling interval in seconds>
```

功能:

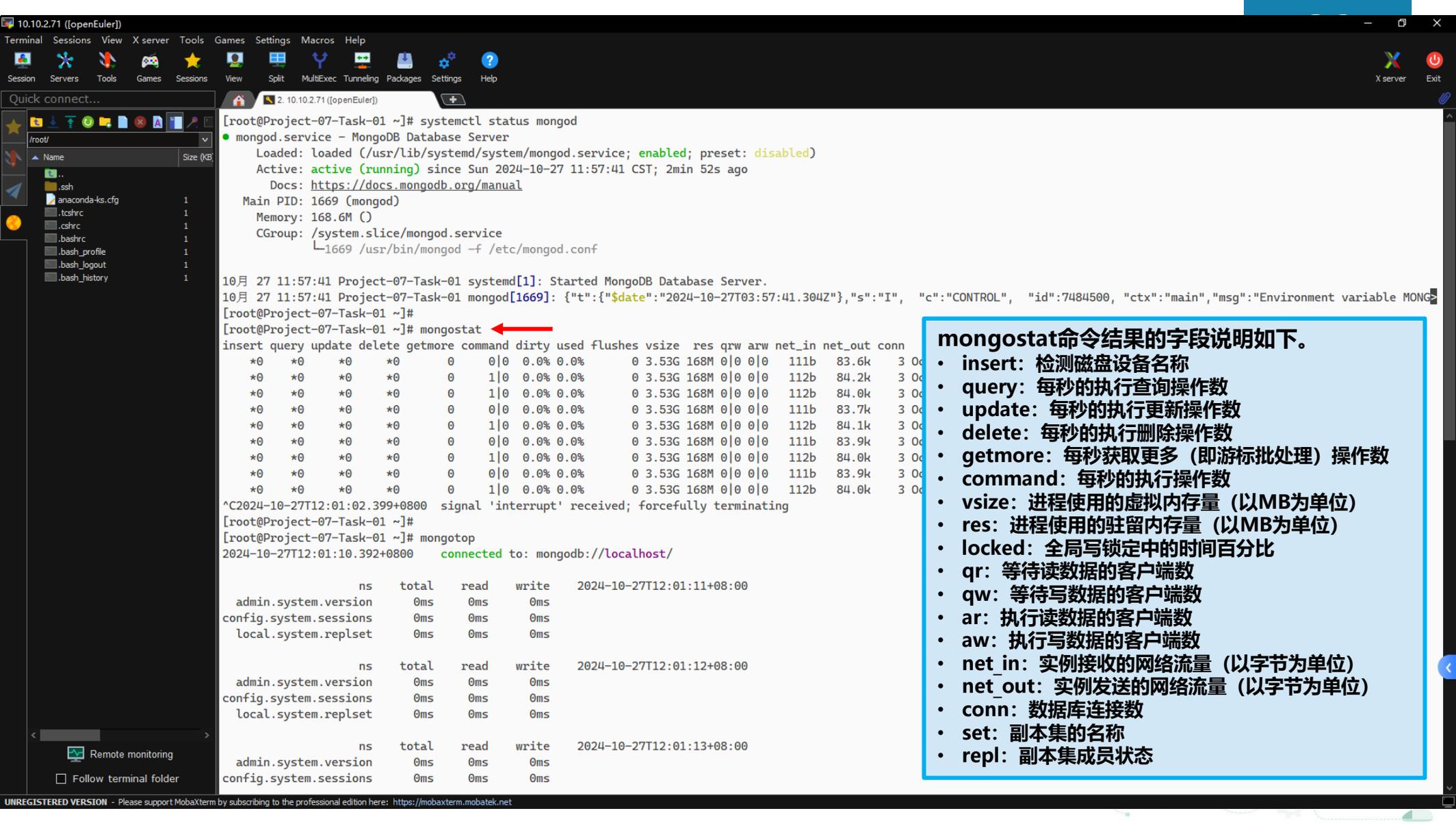
- 根据数据库的操作类型（例如插入、查询、更新、删除等）捕获并返回服务器负载情况。

参数/命令:

- 详细信息查看官方技术手册：
 - <https://www.mongodb.com/zh-cn/docs/database-tools/mongostat/>

主要选项:

- --host=<hostname>:<port>:
 - 数据库服务器主机地址，可使用半角逗号隔开指定多个
- --port=<port>: 数据库服务器主端口号
- --username=<username>, -u=<username>:
 - 数据库连接用户名
- --password=<password>, -p=<password>:
 - 数据库连接密码
- --authenticationDatabase=<dbname>: 认证数据库
- --uri=<connectionString>:
 - 使用连接字符串格式连接，但不可与认证字段同时使用
- --humanReadable=<boolean>: 格式化输出日期等字段
- -O=<field list>:
 - 指定输出字段和字段名称，可使用半角逗号隔开指定多个
- --json: 以JSON格式输出信息
- <sleeptime>: 执行mongostat命令执行的间隔时间



```
[root@Project-07-Task-01 ~]# systemctl status mongod
● mongod.service - MongoDB Database Server
   Loaded: loaded (/usr/lib/systemd/system/mongod.service; enabled; preset: disabled)
   Active: active (running) since Sun 2024-10-27 11:57:41 CST; 2min 52s ago
     Docs: https://docs.mongodb.org/manual
   Main PID: 1669 (mongod)
   Memory: 168.6M ( )
   CGroup: /system.slice/mongod.service
           └─1669 /usr/bin/mongod -f /etc/mongod.conf
```

```
10月 27 11:57:41 Project-07-Task-01 systemd[1]: Started MongoDB Database Server.
10月 27 11:57:41 Project-07-Task-01 mongod[1669]: {"t":{"$date":"2024-10-27T03:57:41.304Z"},"s":"I", "c":"CONTROL", "id":7484500, "ctx":"main","msg":"Environment variable MONG
```

```
[root@Project-07-Task-01 ~]#
[root@Project-07-Task-01 ~]# mongostat ←
insert query update delete getmore command dirty used flushes vsize res qrw arw net_in net_out conn
*0 *0 *0 *0 *0 0 0|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 111b 83.6k 3 0c
*0 *0 *0 *0 *0 0 1|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 112b 84.2k 3 0c
*0 *0 *0 *0 *0 0 1|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 112b 84.0k 3 0c
*0 *0 *0 *0 *0 0 0|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 111b 83.7k 3 0c
*0 *0 *0 *0 *0 0 1|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 112b 84.1k 3 0c
*0 *0 *0 *0 *0 0 0|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 111b 83.9k 3 0c
*0 *0 *0 *0 *0 0 1|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 112b 84.0k 3 0c
*0 *0 *0 *0 *0 0 0|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 111b 83.9k 3 0c
*0 *0 *0 *0 *0 0 1|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 112b 84.0k 3 0c
```

```
^C2024-10-27T12:01:02.399+0800 signal 'interrupt' received; forcefully terminating
[root@Project-07-Task-01 ~]#
[root@Project-07-Task-01 ~]# mongotop
2024-10-27T12:01:10.392+0800 connected to: mongodb://localhost/
```

```
ns total read write 2024-10-27T12:01:11+08:00
admin.system.version 0ms 0ms 0ms
config.system.sessions 0ms 0ms 0ms
local.system.replset 0ms 0ms 0ms

ns total read write 2024-10-27T12:01:12+08:00
admin.system.version 0ms 0ms 0ms
config.system.sessions 0ms 0ms 0ms
local.system.replset 0ms 0ms 0ms

ns total read write 2024-10-27T12:01:13+08:00
admin.system.version 0ms 0ms 0ms
config.system.sessions 0ms 0ms 0ms
```

- mongostat命令结果的字段说明如下。**
- **insert**: 检测磁盘设备名称
 - **query**: 每秒的执行查询操作数
 - **update**: 每秒的执行更新操作数
 - **delete**: 每秒的执行删除操作数
 - **getmore**: 每秒获取更多（即游标批处理）操作数
 - **command**: 每秒的执行操作数
 - **vsize**: 进程使用的虚拟内存量（以MB为单位）
 - **res**: 进程使用的驻留内存量（以MB为单位）
 - **locked**: 全局写锁定中的时间百分比
 - **qr**: 等待读数据的客户端数
 - **qw**: 等待写数据的客户端数
 - **ar**: 执行读数据的客户端数
 - **aw**: 执行写数据的客户端数
 - **net_in**: 实例接收的网络流量（以字节为单位）
 - **net_out**: 实例发送的网络流量（以字节为单位）
 - **conn**: 数据库连接数
 - **set**: 副本集的名称
 - **repl**: 副本集成员状态

3. 管理和监控

3.1 MongoDB Database Tools



mongotop <options> <connection-string> <polling interval in seconds>

功能:

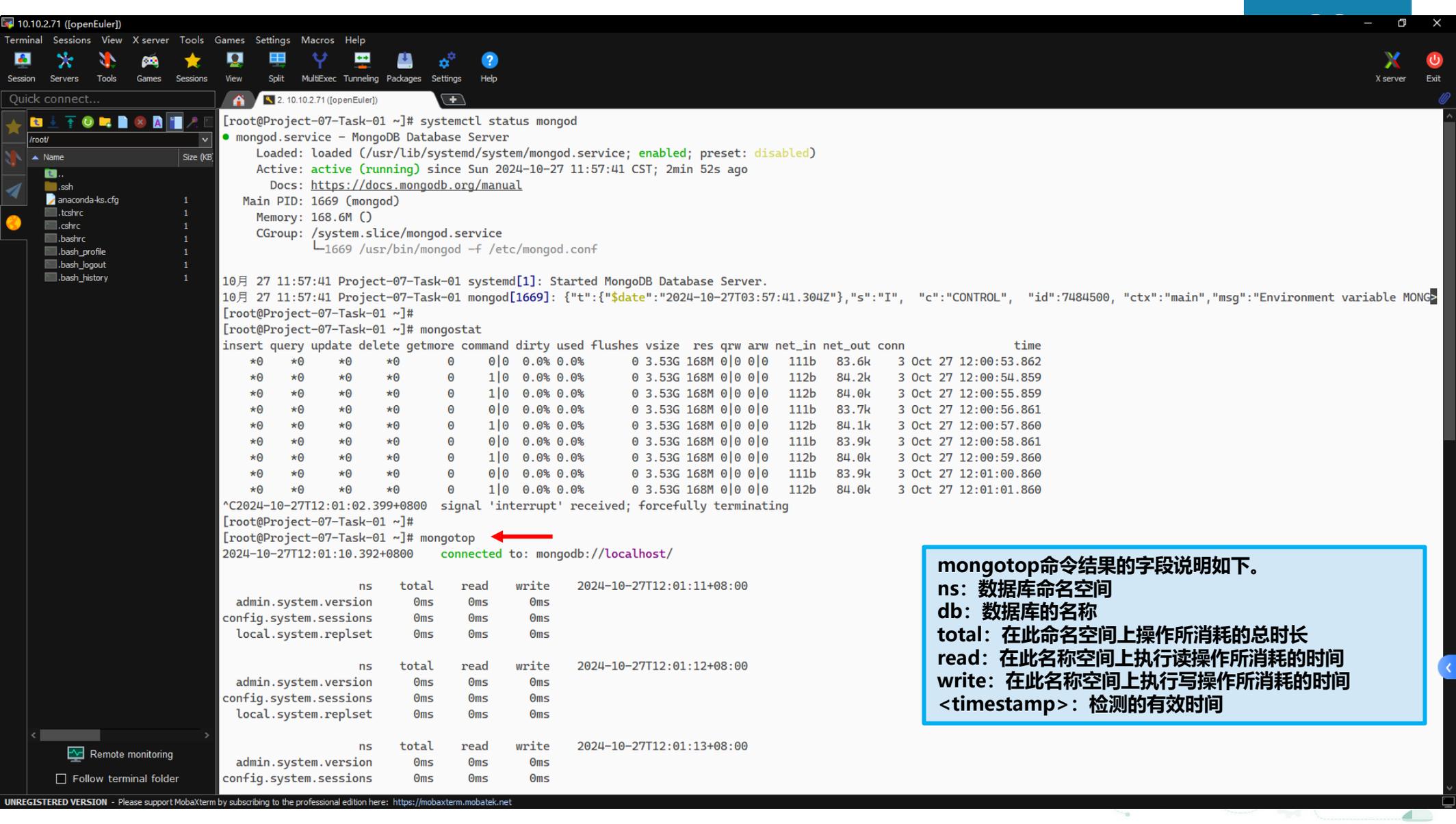
- 根据数据库的操作类型（例如插入、查询、更新、删除等）捕获并返回服务器负载情况。

参数/命令:

- 详细信息查看官方技术手册:
 - <https://www.mongodb.com/zh-cn/docs/database-tools/mongotop/>
- 应用举例:
 - mongotop
 - 没有任何命令行选项运行 mongotop，将连接到在本地主机上运行的 27017 实例，且每秒返回一次值。
 - mongotop 30
 - 为使 mongotop 每 30 秒报告一次，请指定轮询间隔:

主要选项:

- --host=<hostname>:<port>:
 - 数据库服务器主机地址，可使用半角逗号隔开指定多个
- --port=<port>: 数据库服务器主端口号
- --username=<username>, -u=<username>:
 - 数据库连接用户名
- --password=<password>, -p=<password>:
 - 数据库连接密码
- --authenticationDatabase=<dbname>: 认证数据库
- --uri=<connectionString>:
 - 使用连接字符串格式连接，但不可与认证字段同时使用
- --humanReadable=<boolean>: 格式化输出日期等字段
- -O=<field list>:
 - 指定输出字段和字段名称，可使用半角逗号隔开指定多个
- --json: 以JSON格式输出信息
- <sleeptime>: 执行mongostat命令执行的间隔时间



```
[root@Project-07-Task-01 ~]# systemctl status mongod
● mongod.service - MongoDB Database Server
   Loaded: loaded (/usr/lib/systemd/system/mongod.service; enabled; preset: disabled)
   Active: active (running) since Sun 2024-10-27 11:57:41 CST; 2min 52s ago
     Docs: https://docs.mongodb.org/manual
   Main PID: 1669 (mongod)
   Memory: 168.6M ( )
   CGroup: /system.slice/mongod.service
           └─1669 /usr/bin/mongod -f /etc/mongod.conf
```

```
10月 27 11:57:41 Project-07-Task-01 systemd[1]: Started MongoDB Database Server.
10月 27 11:57:41 Project-07-Task-01 mongod[1669]: {"t":{"$date":"2024-10-27T03:57:41.304Z"},"s":"I", "c":"CONTROL", "id":7484500, "ctx":"main","msg":"Environment variable MONG
```

```
[root@Project-07-Task-01 ~]#
[root@Project-07-Task-01 ~]# mongostat
insert query update delete getmore command dirty used flushes vszize  res qrw arw net_in net_out conn                time
*0 *0 *0 *0 *0 0 0|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 111b 83.6k 3 Oct 27 12:00:53.862
*0 *0 *0 *0 *0 0 1|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 112b 84.2k 3 Oct 27 12:00:54.859
*0 *0 *0 *0 *0 0 1|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 112b 84.0k 3 Oct 27 12:00:55.859
*0 *0 *0 *0 *0 0 0|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 111b 83.7k 3 Oct 27 12:00:56.861
*0 *0 *0 *0 *0 0 1|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 112b 84.1k 3 Oct 27 12:00:57.860
*0 *0 *0 *0 *0 0 0|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 111b 83.9k 3 Oct 27 12:00:58.861
*0 *0 *0 *0 *0 0 1|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 112b 84.0k 3 Oct 27 12:00:59.860
*0 *0 *0 *0 *0 0 0|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 111b 83.9k 3 Oct 27 12:01:00.860
*0 *0 *0 *0 *0 0 1|0 0.0% 0.0% 0 3.53G 168M 0|0 0|0 112b 84.0k 3 Oct 27 12:01:01.860
```

```
^C2024-10-27T12:01:02.399+0800 signal 'interrupt' received; forcefully terminating
[root@Project-07-Task-01 ~]#
[root@Project-07-Task-01 ~]# mongotop ←
2024-10-27T12:01:10.392+0800 connected to: mongodb://localhost/
```

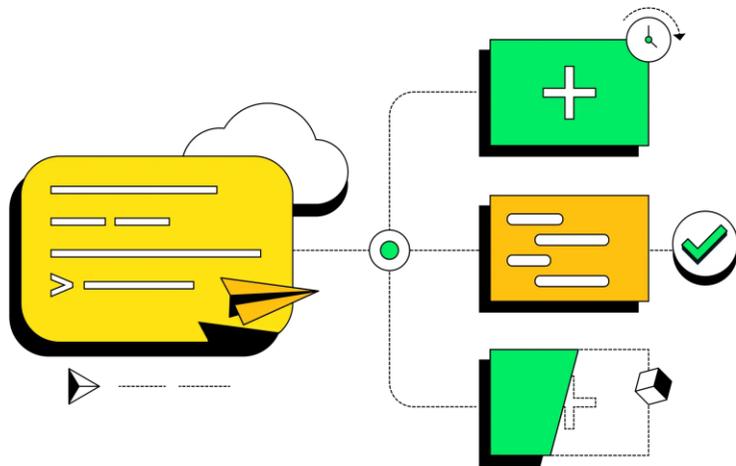
ns	total	read	write	2024-10-27T12:01:11+08:00
admin.system.version	0ms	0ms	0ms	
config.system.sessions	0ms	0ms	0ms	
local.system.replset	0ms	0ms	0ms	
ns	total	read	write	2024-10-27T12:01:12+08:00
admin.system.version	0ms	0ms	0ms	
config.system.sessions	0ms	0ms	0ms	
local.system.replset	0ms	0ms	0ms	
ns	total	read	write	2024-10-27T12:01:13+08:00
admin.system.version	0ms	0ms	0ms	
config.system.sessions	0ms	0ms	0ms	

mongotop命令结果的字段说明如下。
ns: 数据库命名空间
db: 数据库的名称
total: 在此命名空间上操作所消耗的总时长
read: 在此名称空间上执行读操作所消耗的时间
write: 在此名称空间上执行写操作所消耗的时间
<timestamp>: 检测的有效时间

3. 管理和监控

3.2 MongoDB Shell

- MongoDB Shell (mongosh)
 - 是一个JavaScript 和 Node.js REPL环境。
 - 用于与Atlas、本地或其他远程主机上的MongoDB实例进行交互操作。
- 使用MongoDB Shell可以：
 - 连接到MongoDB
 - 执行增删改查和聚合操作
 - 管理MongoDB
 - 管理数据库和集合
 - 管理服务器
 - 管理用户和角色
 - 使用脚本实现自动化运维



MongoDB Atlas

MongoDB Enterprise Advanced

MongoDB Community Edition

Tools

MongoDB Shell

MongoDB Compass (GUI)

Atlas CLI

Atlas Kubernetes Operator

MongoDB CLI for Cloud
Manager and Ops Manager

MongoDB Cluster-to-Cluster
Sync

Relational Migrator

MongoDB Database Tools

MongoDB Connector for BI

App Services CLI

Atlas SQL Interface

Mobile & Edge



TOOLS

MongoDB Shell Download

MongoDB Shell is the quickest way to connect to (and work with) MongoDB. Easily query data, configure settings, and execute other actions with this modern, extensible command-line interface – replete with syntax highlighting, intelligent autocomplete, contextual help, and error messages.

Compatibility Note: Red Hat Enterprise Linux (RHEL) 7, Amazon Linux 2, SUSE Linux Enterprise Server (SLES) 12, and Ubuntu 18.04 support is deprecated and might be removed in a later mongosh release.

Note: MongoDB Shell is an open source (Apache 2.0), standalone product developed separately from the MongoDB Server.

Learn more

Version

2.3.2

Platform

Linux x64

Package

tgz

Download



Copy link

More Options

- Docs 主页
- MongoDB Shell
 - 安装
 - 连接
 - 配置
 - 运行命令
 - 执行 CRUD 操作
 - 运行聚合管道
 - 客户端字段级加密 (Client-Side Field Level Encryption)
 - 编写脚本
 - Snippets
 - 参考
 - 版本说明
 - 帮助

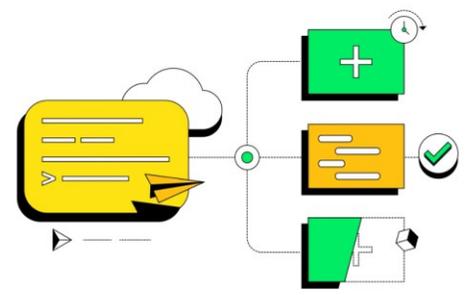
搜索 MongoDB 文档

反馈

欢迎使用 MongoDB Shell (mongosh)

MongoDB Shell (mongosh) 是一个 JavaScript 和 Node.js REPL 环境，用于与 Atlas、本地或其他远程主机上的 MongoDB 部署进行交互。使用 MongoDB Shell 测试查询，并与 MongoDB 数据库中的数据

下载 mongosh



您能做什么

从 Shell 访问 MongoDB

- 连接至 MongoDB
- 执行增删改查和聚合操作
- 管理
- 使用脚本实现自动化

1 查找连接字符串

查找连接字符串。连接字符串各不相同，具体情况取决于要连接到的部署类型。

了解如何查找 Atlas 的连接字符串。

或连接到自托管部署。

2 连接至 MongoDB

使用连接字符串连接到一个 MongoDB 部署。

如下连接字符串将连接到一个 Atlas 部署：

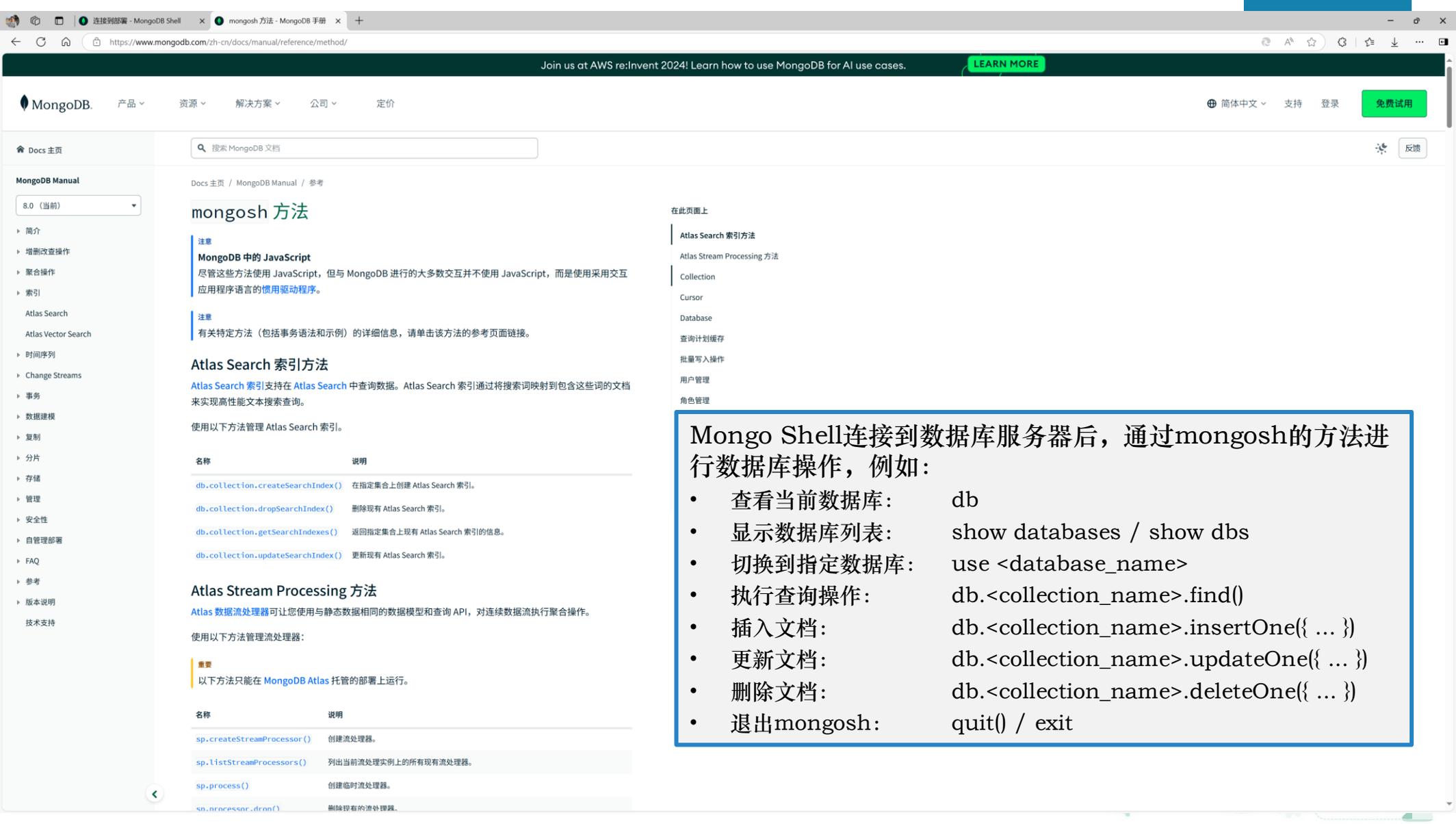
```
mongosh "mongodb+srv://mycluster.abcd1.mongodb.net/m
```

```
~$ mongosh "mongodb+srv://cluster0.ab123.mongodb.net/myFirstDatabase" --username user
Enter password: *****
Current Mongosh Log ID: 6142707ff2a7e240da5ade56
Connecting to:      mongodb+srv://cluster0.ab123.mongodb.net/myFirstDatabase
Using MongoDB:      4.4.8
Using Mongosh:      1.0.6

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

Atlas atlas-cej3pr-shard-0 [primary] myFirstDatabase> show dbs
sample_airbnb      57.2 MB
sample_analytics   9.54 MB
sample_geospatial  1.1 MB
sample_mflix       59 MB
sample_restaurants 6.91 MB
sample_supplies    1.02 MB
sample_training    46.6 MB
sample_weatherdata 2.72 MB
admin              336 KB
local              6.76 GB

Atlas atlas-cej3pr-shard-0 [primary] myFirstDatabase>
```



- MongoDB Manual
- 8.0 (当前)
- 简介
- 增删改查操作
- 聚合操作
- 索引
 - Atlas Search
 - Atlas Vector Search
- 时间序列
- Change Streams
- 事务
- 数据建模
- 复制
- 分片
- 存储
- 管理
- 安全性
- 自我管理部署
- FAQ
- 参考
- 版本说明
- 技术支持

Docs 主页 / MongoDB Manual / 参考

mongosh 方法

注意
MongoDB 中的 JavaScript
尽管这些方法使用 JavaScript，但与 MongoDB 进行的大多数交互并不使用 JavaScript，而是使用采用交互应用程序语言的惯用驱动程序。

注意
有关特定方法（包括事务语法和示例）的详细信息，请单击该方法的参考页面链接。

Atlas Search 索引方法

Atlas Search 索引支持在 Atlas Search 中查询数据。Atlas Search 索引通过将搜索词映射到包含这些词的文档来实现高性能文本搜索查询。

使用以下方法管理 Atlas Search 索引。

名称	说明
<code>db.collection.createSearchIndex()</code>	在指定集合上创建 Atlas Search 索引。
<code>db.collection.dropSearchIndex()</code>	删除现有 Atlas Search 索引。
<code>db.collection.getSearchIndexes()</code>	返回指定集合上现有 Atlas Search 索引的信息。
<code>db.collection.updateSearchIndex()</code>	更新现有 Atlas Search 索引。

Atlas Stream Processing 方法

Atlas 数据流处理器可让您使用与静态数据相同的数据模型和查询 API，对连续数据流执行聚合操作。

使用以下方法管理流处理器：

重要
以下方法只能在 MongoDB Atlas 托管的部署上运行。

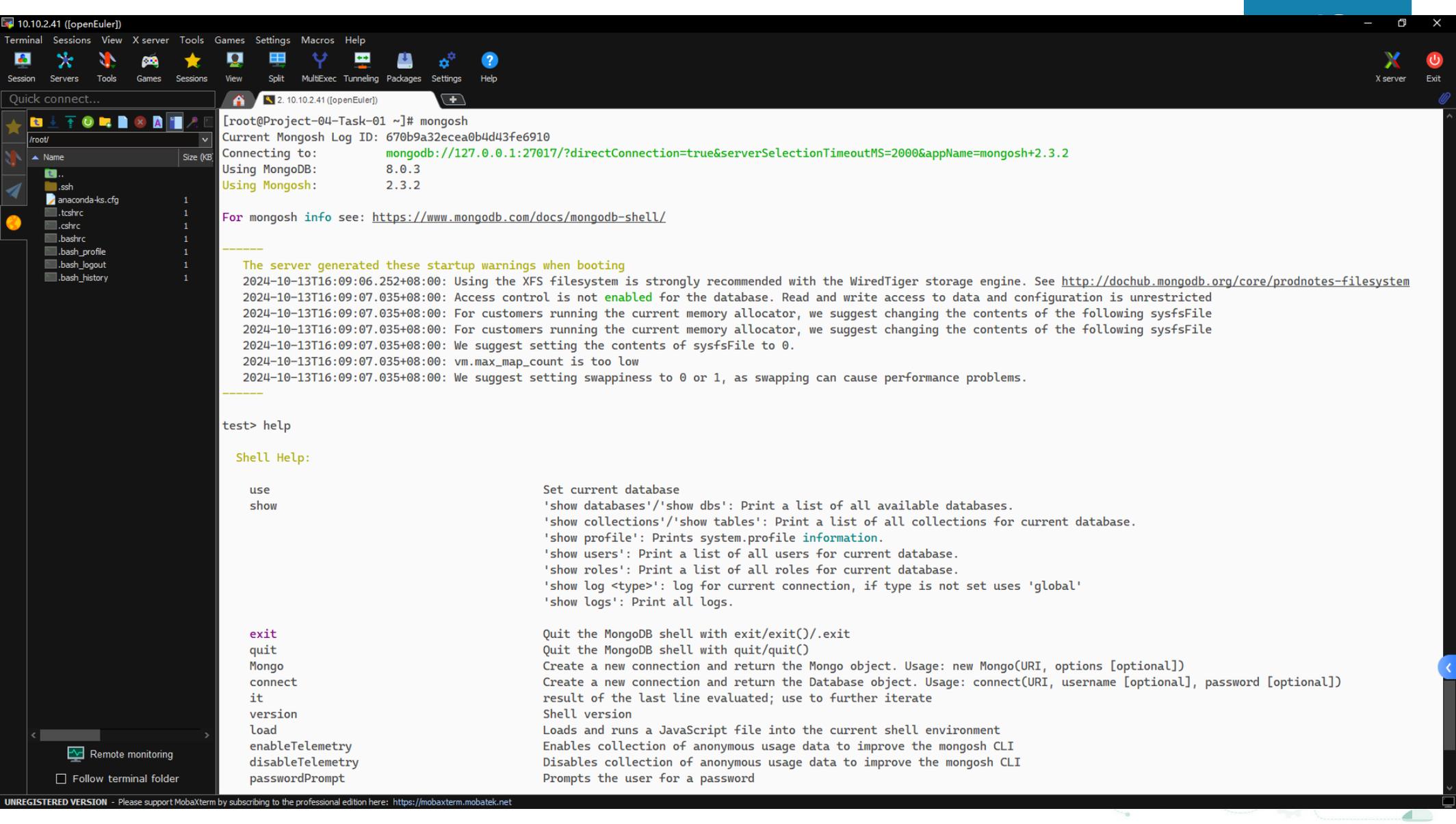
名称	说明
<code>sp.createStreamProcessor()</code>	创建流处理器。
<code>sp.listStreamProcessors()</code>	列出当前流处理实例上的所有现有流处理器。
<code>sp.process()</code>	创建临时流处理器。
<code>sp.processor.dron()</code>	删除现有的流处理器。

在此页面上

- Atlas Search 索引方法
- Atlas Stream Processing 方法
- Collection
- Cursor
- Database
- 查询计划缓存
- 批量写入操作
- 用户管理
- 角色管理

Mongo Shell连接到数据库服务器后，通过mongosh的方法进行数据库操作，例如：

- 查看当前数据库：`db`
- 显示数据库列表：`show databases / show dbs`
- 切换到指定数据库：`use <database_name>`
- 执行查询操作：`db.<collection_name>.find()`
- 插入文档：`db.<collection_name>.insertOne({ ... })`
- 更新文档：`db.<collection_name>.updateOne({ ... })`
- 删除文档：`db.<collection_name>.deleteOne({ ... })`
- 退出mongosh：`quit() / exit`



```
[root@Project-04-Task-01 ~]# mongosh
Current Mongosh Log ID: 670b9a32ecea0b4d43fe6910
Connecting to:      mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+2.3.2
Using MongoDB:     8.0.3
Using Mongosh:     2.3.2
```

For mongosh info see: <https://www.mongodb.com/docs/mongosh-shell/>

The server generated these startup warnings when booting

```
2024-10-13T16:09:06.252+08:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
2024-10-13T16:09:07.035+08:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
2024-10-13T16:09:07.035+08:00: For customers running the current memory allocator, we suggest changing the contents of the following sysfsFile
2024-10-13T16:09:07.035+08:00: For customers running the current memory allocator, we suggest changing the contents of the following sysfsFile
2024-10-13T16:09:07.035+08:00: We suggest setting the contents of sysfsFile to 0.
2024-10-13T16:09:07.035+08:00: vm.max_map_count is too low
2024-10-13T16:09:07.035+08:00: We suggest setting swappiness to 0 or 1, as swapping can cause performance problems.
```

test> help

Shell Help:

use	Set current database
show	'show databases'/'show dbs': Print a list of all available databases. 'show collections'/'show tables': Print a list of all collections for current database. 'show profile': Prints system.profile information. 'show users': Print a list of all users for current database. 'show roles': Print a list of all roles for current database. 'show log <type>': log for current connection, if type is not set uses 'global' 'show logs': Print all logs.
exit	Quit the MongoDB shell with exit/exit()/exit
quit	Quit the MongoDB shell with quit/quit()
Mongo	Create a new connection and return the Mongo object. Usage: new Mongo(URI, options [optional])
connect	Create a new connection and return the Database object. Usage: connect(URI, username [optional], password [optional])
it	result of the last line evaluated; use to further iterate
version	Shell version
load	Loads and runs a JavaScript file into the current shell environment
enableTelemetry	Enables collection of anonymous usage data to improve the mongosh CLI
disableTelemetry	Disables collection of anonymous usage data to improve the mongosh CLI
passwordPrompt	Prompts the user for a password

3. 管理和监控

3.3 MongoDB Compass(GUI)

- ❑ MongoDB Compass是MongoDB官方提供的一款可视化数据库管理工具。
 - 支持多种操作系统，包括Windows、macOS和Linux。
- ❑ MongoDB Compass的主要功能：

功能	详情
数据管理	创建、查看、修改和删除数据库、数据表、视图、列、索引等
索引管理	创建、删除索引
数据聚合	创建和执行聚合管道
数据导入导出	支持从 SQL、CSV、XML 等格式文件导入数据 支持以 CSV、XML、PDF、SQL 等格式导出数据
监控	监控数据库服务器的流量、连接、进程、查询统计、数据库变量状态、主机状态等
集群管理	查看集群状态、查看集群成员、添加集群成员
统一身份验证	支持 Kerberos、LDAP 和 x.509 身份验证
文档模型分析	提供对指定集中文档的字段和值的分析



3. 管理和监控

3.3 MongoDB Compass(GUI)

□ MongoDB Compass的版本:

- Compass: 完整版的MongoDB Compass, 具有所有功能
- Compass Readonly: 只允许读取操作
- Compass Isolated: 除了连接MongoDB服务器外, 不发起任何网络请求

	Compass	Compass Readonly	Compass 隔离
通过完整的 CRUD 功能与文档、collection和数据库进行交互	√		√
创建并执行查询和聚合管道	√	√	√
创建和删除索引	√		√
使用可视化解释计划查看和优化查询性能	√	√	√
Kerberos、LDAP 和 x.509 身份验证	√	√	√
模式分析	√	√	√
实时服务器统计信息	√	√	√
创建、删除和编辑文档验证规则	√		√
错误报告和数据使用情况collection	√	√	
Automatic updates	√	√	
嵌入式 Shell 支持	√		√



MongoDB Atlas

MongoDB Enterprise Advanced

MongoDB Community Edition

Tools

MongoDB Shell

MongoDB Compass (GUI)

Atlas CLI

Atlas Kubernetes Operator

MongoDB CLI for Cloud
Manager and Ops Manager

MongoDB Cluster-to-Cluster
Sync

Relational Migrator

MongoDB Database Tools

MongoDB Connector for BI

App Services CLI

Atlas SQL Interface

Mobile & Edge



TOOLS

MongoDB Compass Download (GUI)

Easily explore and manipulate your database with Compass, the GUI for MongoDB. Intuitive and flexible, Compass provides detailed schema visualizations, real-time performance metrics, sophisticated querying abilities, and much more.

Please note that MongoDB Compass comes in three versions: a **full version** with all features, a **read-only version** without write or delete capabilities, and an **isolated edition**, whose sole network connection is to the MongoDB instance.

For more information, see our [documentation pages](#).

Compass

The full version of MongoDB Compass, with all features and capabilities.

Readonly Edition

This version is limited strictly to read operations, with all write and delete capabilities removed.

Isolated Edition

This version disables all network connections except the connection to the MongoDB instance.

Learn more

Version

1.44.5 (Stable)

Platform

Windows 64-bit (10+)

Package

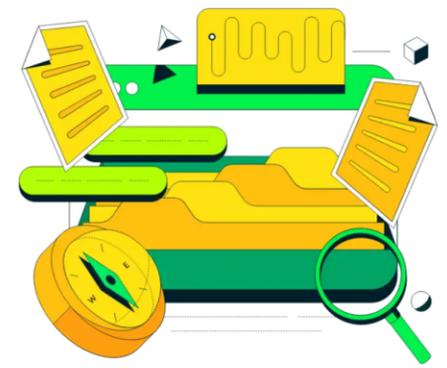
exe

什么是 MongoDB Compass?

MongoDB Compass 是一款功能强大的 GUI，用于在可视环境中查询、聚合和分析 MongoDB 数据。

Compass 可免费使用并提供源代码，并且可以在 macOS、Windows 和 Linux 上运行。

[下载 Compass](#) [查看安装说明](#)



- MongoDB Compass
- Overview
- 下载并安装
- 连接
- 自定义
- 主屏幕
- 与您的数据交互
- 导入和导出数据
- 嵌入式 MongoDB Shell
- 创建聚合管道
- 故障排除
- 键盘快捷键
- FAQ
- 了解详情
- 版本说明
- 提交反馈

您能做什么

直观地探索您的数据

浏览 Compass 可以帮助您完成的一些任务，例如通过易于导航的界面导入和管理数据。

- 导入数据
- 查询您的数据
- 创建聚合管道
- 在 shell 中运行命令**

1 连接到您的部署
 连接到 MongoDB Atlas 上托管的 MongoDB 部署，或您自己的计算机上本地托管的部署。
[要了解更多信息，请参阅“连接到 MongoDB”](#)

2 在 MongoDB Shell 中处理数据
 使用 Compass 中的嵌入式 MongoDB Shell 在交互式 JavaScript 环境中控制数据。

```

MongoDB Data
use sample_weatherdata
use sample_weatherdata
db.data.find( { "demoPoint.value" : { $lte: 600 } }, { type: 1, demoPoint: 1 }, limit: 5 )
( { "_id" : ObjectId("5553996c4082c7f15319801c"),
  type: "PM-2.5",
  demoPoint: { value: 3.5, quality: "1" } }
  { "_id" : ObjectId("5553996c4082c7f15319801c"),
  type: "PM-2.5",
  demoPoint: { value: 3.5, quality: "1" } }
  { "_id" : ObjectId("5553996c4082c7f15319801c"),
  type: "PM-2.5",
  demoPoint: { value: 3.5, quality: "1" } }
  { "_id" : ObjectId("5553996c4082c7f15319801c"),
  type: "PM-2.5",
  demoPoint: { value: 4.5, quality: "1" } }
  { "_id" : ObjectId("5553996c4082c7f15319801c"),
  type: "PM-2.5",
  demoPoint: { value: 4.5, quality: "1" } }
  )
  
```

3. 管理和监控

3.3 MongoDB Compass(GUI)



使用MongoDB Compass管理数据库服务器

任务目标:

- 完成MongoDB Compass的安装，并对MongoDB数据库服务器进行管理。

操作步骤:

- 获取MongoDB Compass
- 安装并配置
- 连接到MongoDB
- 数据管理和操作
- 监控数据库服务器

操作演示:



Compass

My Queries

CONNECTIONS (1)

Search connections

- 10.10.2.71
 - admin
 - config
 - local

10.10.2.71

10.10.2.71

Sort by Database Name

Open MongoDB shell

Create database

Refresh

View

Database Name	Storage size	Collections	Indexes
admin	57.34 kB	0	3
config	24.58 kB	0	2
local	36.86 kB	3	1

3. 管理和监控

3.4 Percona Monitoring and Management

- Percona Monitoring and Management（简称“PMM”）是一款开源数据库监控和管理工具软件。
 - 支持MongoDB、MySQL、MariaDB、PostgreSQL等常用数据库。
 - PMM内置监控指标和性能分析模型，并具有可视化监控分析功能。
 - PMM依据监控数据可提供性能、安全性和配置建议。

- 官方的描述：
 - An open source database monitoring, observability, and management tool.
 - Monitor, manage, and improve the performance of your databases no matter where they are located or deployed.



Database Monitoring Tools for M x +

https://www.percona.com/software/database-tools/percona-monitoring-and-management

PERCONA Solutions Resources Community About



Percona Monitoring and Management

An open source database monitoring, observability, and management tool

[INSTALL NOW](#) [EXPLORE DOCUMENTATION](#)

Monitor, manage, and improve the performance of your databases no matter where they are located or deployed.

Percona Monitoring and Management (PMM) is an open source database observability, monitoring, and management tool for use with MySQL, PostgreSQL, MongoDB, and the servers on which they run. It enables you to view node- to single-query performance metrics for all of your databases in a single place. With Query Analytics, you can quickly locate costly and slow-running queries to address bottlenecks. Additionally, Percona Advisors equip you with performance, security, and configuration recommendations that help you keep your databases performing at their best. Alerting and management features like backup, restore, and built-in open source Private DBaaS are designed to increase the velocity of your IT team.

Monitor and troubleshoot
Monitor your database performance with customizable dashboards and real-time alerting. Learn more about

Fix Bottlenecks
Zoom in and drill down on database performance, from node to single-query levels. Quickly locate and fix



Percona Monitoring and Management x Database Monitoring Tools for M x +

https://www.percona.com/software/database-tools/percona-monitoring-and-management

PERCONA Solutions Resources Community About Blog [Contact us](#)

Monitor and troubleshoot

Monitor your database performance with customizable dashboards and real-time alerting. Learn more about Percona Alerting and technology-specific dashboards.

[MYSQL MONITORING](#) [MONGODB MONITORING](#) [QUERY ANALYTICS](#)
[POSTGRESQL MONITORING](#) [PERCONA ALERTING](#)

Fix Bottlenecks

Zoom in and drill down on database performance, from node to single-query levels. Quickly locate and fix bottlenecks and slow-running queries with Query Analytics.

Optimize

Use automated insights and recommendations for security posture, configuration, and performance to uncover problems before they become larger issues.

[PERCONA ADVISORS](#)

Backup and Restore

Back up your critical data with zero downtime and minimal performance impact. Schedule various types of backups (hot, incremental, physical) and restore databases up to a specific moment with Point-in-Time Recovery.

[BACKUP & RESTORE](#)

Supported Technologies and Dashboards

[MySQL](#) [MongoDB](#) [PostgreSQL](#) [Amazon RDS and Amazon Aurora](#) [ProxySQL](#) [Percona XtraDB Cluster](#)

Percona Monitoring and Management provides actionable performance data for MySQL variants, including Percona Server for MySQL, Percona XtraDB



- Percona Monitoring and Management
- Welcome
- Quickstart guide
- Setting up
- Get started
- How to
- Details
- DBaaS
- FAQ
- Release notes
- Reference
- Download PDF

Welcome

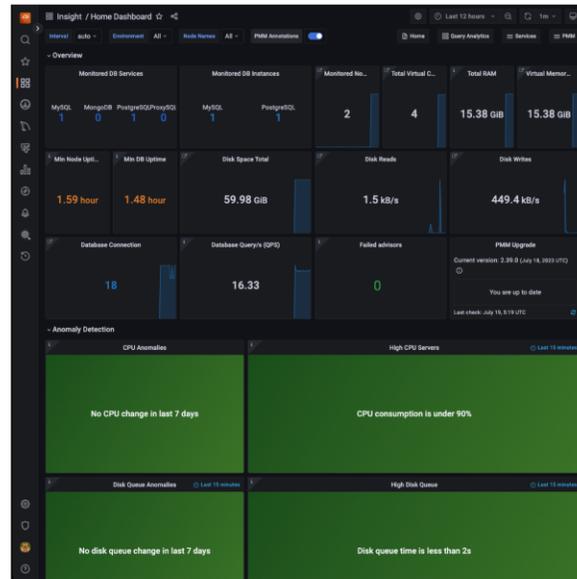
Percona Monitoring and Management (PMM) is an open-source database monitoring, management, and observability solution for MySQL, PostgreSQL, and MongoDB.

It allows you to observe the health of your database systems, explore new patterns in their behavior, troubleshoot them and perform database management operations no matter where they are located on-prem or in the cloud.

- PMM **collects** thousands of out-of-the-box performance **metrics** from databases and their hosts.
- The PMM [web UI](#) **visualizes data** in [dashboards](#).
- Additional features include [advisors for database health assessments](#).

This is the documentation for the latest release, **PMM 2.43.1** ([Release Notes](#)).

Here's how the home page looks on our [free, live demo system](#).



PMM runs in the cloud, on-prem, or across hybrid platforms. It's supported by our [legendary expertise](#) in open source databases, and by a vibrant developer and user [community](#).

- Table of contents
- Start here
- Read more
 - Links to popular sections
 - For System Administrators
 - For Users
- Get expert help

- Percona Monitoring and Management
- Welcome
- Quickstart guide
- Setting up
- Get started
- How to
- Details
- DBaaS
- FAQ
- Release notes
- Reference
- Download PDF

Get started with PMM

To get up and running with Percona Monitoring and Management (PMM) in no time, install PMM on Bare Metal/Virtual using the Easy-install script for Docker.

This is the simplest and most efficient way to install PMM.

1 Alternative installation options

Prerequisites

Before you start installing PMM, verify that your system meets the compatibility requirements.

1 Verify system compatibility

Install PMM

The Easy-install script only runs on Linux-compatible systems. To use it, run the command with `sudo` privileges or as `root`:

- 1 Download and install PMM using `cURL` or `wget`:

```

cURL      wget
curl -fsSL https://www.percona.com/get/pmm | /bin/bash

```

- 2 After the installation is complete, log into PMM with the default `admin:admin` credentials.

1 What's happening under the hood?

Connect database

Once PMM is set up, choose the database or the application that you want it to monitor:

- MySQL
- PostgreSQL
- MongoDB
- ProxySQL
- HAProxy

To connect a self-hosted MySQL database:

- 1 Create database account for PMM using the following command example. This creates a database user with name `pmm`, password `<your_password>`, and the necessary permissions:

```

CREATE USER 'pmm'@'127.0.0.1' IDENTIFIED BY '<your_password>' WITH MAX_USER_CONNECTIONS 10;
GRANT SELECT, PROCESS, REPLICATION CLIENT, RELOAD, BACKUP_ADMIN ON *.* TO 'pmm'@'127.0.0.1';

```

- 2 To optimize server-side resources, install PMM Client via Package Manager on the database node:

3. 管理和监控

3.4 Percona Monitoring and Management



使用Percona Monitoring and Management管理数据库服务器

任务目标:

- 部署PMM，并实现对MongoDB、MySQL Server的监控和运维管理。

操作步骤:

- 部署PMM
- 监控和管理MongoDB
- 监控和管理MySQL Server
- 理解监控数据
- 理解优化建议

操作演示:



网络与信息系统智能运维 课程体系学习平台

本课程体系由
河南中医药大学信息技术学院建设

课程体系学习平台由河南中医药大学医疗健康信息
工程技术研究所开发与技术保障

网络与信息系统智能运维课程体系学习平台
<https://internet.hactcm.edu.cn>

互联网运维管理工程应用丛书
<http://www.51xueweb.cn>



扫码学习
并获取课程资源

