



---

---

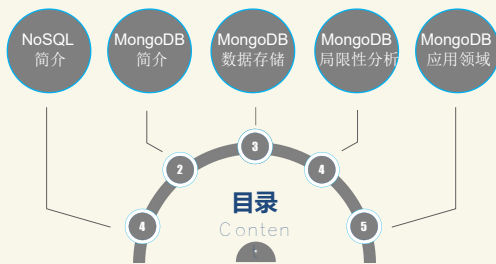
---

---

---

---

---



---

---

---

---

---

---

---

1 NoSQL简介 (1)

NoSQL简介 MongoDB简介 数据存储 局限性分析 应用领域

> NoSQL是Not Only SQL的缩写，而不是Not SQL，它不一定遵循传统数据库的一些基本要求，如SQL标准、ACID属性、表结构等。相比传统数据库，叫它分布式数据管理系统更贴切，数据存储被简化更灵活，重点被放在了分布式数据管理上。

> 大数据的性能要求的背景需求：  
High performance —高并发读写，高并发、实时动态数据查询和修改。  
Huge Storage —海量数据的高效存储和访问，海量用户信息的高效、实时存储和查询。  
High Scalability , High Availability —高可扩展性和高可用性，需要拥有快速横向扩展能力、提供不间断服务。

> 关系数据库处理大数据的弱势：  
面对高并发读写需求，数据库读写压力大，硬盘IO无法承受；  
面对海量数据，数据库存储记录有限，SQL查询效率低；  
横向扩展艰难，无法通过快速增加服务器节点升级和维护系统。

<

>

---

---

---

---

---

---

---

1 NoSQL 简介 (2)

NoSQL 简介MongoDB 简介数据存储局限性分析应用领域

数据库结构相对简单，数据间无关联性，这自然就带来了很高的读写性能，且易扩展。

灵活的数据模型，可以随时存储自定义的数据格式。而在关系数据库里，大数据量的表，增加字段比较麻烦。

通过复制模型实现高可用性，可以分布部署在低廉的PC集群上。

支持动态增加、删除服务器节点，随时控制硬件投入成本。

NoSQL处理大数据的优势

<

>

---

---

---

---

---

---

---

---

2 MongoDB 简介 (1)

NoSQL 简介MongoDB 简介数据存储局限性分析应用领域

类型	部分代表	特点
key-value 存储	Dynamo Tokyo Cabinet Berkeley DB Voldemort MemcacheDB Redis LevelDB	可以通过key快速查询到其value/list。value是最灵活及易扩展的数据结构，一般来说，存储不管value的格式，照单全收。灵活性和复杂性是双刃剑：一方面可以任意组织文档的结构，另一方面查询需求会变得比较复杂
列存储	BigTable Hbase Cassandra Hypertable	源自Google的BigTable，按列存储。方便存储结构化和半结构化数据，方便数据压缩，针对对某一列或者某几列的查询有非常大的IO优势
文档存储	MongoDB CouchDB Riak	一般用类json的格式存储，存储的内容是文档型的。这样也就有机会对某些字段建立索引，而且可以实现关系数据库的某些复杂查询功能
图存储	Neo4j FlockDB HyperGraphDB	数据并非对等的，关系型的存储或者键值对的存储，可能都不是最好的存储方式，图存储是图形关系的最佳存储

<

>

---

---

---

---

---

---

---

---

2 MongoDB 简介 (2)

NoSQL 简介MongoDB 简介数据存储局限性分析应用领域

> MongoDB简介

一个NoSQL中，功能最丰富的分布式文件存储数据库。

它支持的数据结构非常松散，是类似JSON的BSON格式，因此可以存储比较复杂的数据模型。

它支持的查询语言非常强大，其语法有点类似于面向对象的查询语言，而且还支持索引等功能。

> MongoDB主要特点

高性能、易部署、易使用，存储数据方便。

模式自由，支持动态查询、完全索引、文档内嵌查询。

面向文档，以K/V形式存储数据，Key用于唯一标识，而Value则可以是各种复杂的数据类型。

支持主/从服务器间的数据复制和故障恢复。

自动分片，以支持云级别的服务伸缩性，可动态添加、删除额外的服务器。

<

>

---

---

---

---

---

---

---

---

2 MongoDB简介 (3)

NoSQL简介MongoDB简介数据存储局限性分析应用领域

>

文档是MongoDB的核心概念：  
an ordered set of keys with associated values. 有序的键值对  
同很多语言中的map, hash, dictionary自然对应  
{ "greeting" : "Hello, world!" }, 一个键值对  
[ { "greeting" : "Hello, world!" , "foo" : 3} , 两个键值对  
Python字典、 Perl/Ruby中的Hash等不关心次序，但MongoDB是区别次序的。  
一个Document中各个键值对数据类型可以不一样。Key，当然只能是字符串，但是唯一的。  
MongoDB，是类型敏感和大小写敏感的。  
每个文档有一个特别的Key " \_id" ，相当于rowID

<

>

---

---

---

---

---

---

---

---

3 MongoDB数据存储 (1)

NoSQL简介MongoDB简介数据存储局限性分析应用领域

>

Binary JSON ( Binary Serialized Document Notation ) 是一种类JSON二进制形式的存储格式，支持内嵌的文档对象和数组对象 包括基本JSON数据类型和JSON没有的数据类型  
布尔ture | false  
整数123  
浮点12.3  
字符串"hello world"  
数组列表或一套值可以以array表示 [, "apple", "banana", "pear"]  
日期以毫秒计，不保存时区 用new Date()来声明  
时间戳  
对象ID 是一个独特的12字节ID 用new ObjectId()来声明  
内嵌文档 { "name" : "Joe", "age" : 13, "phone" : { "home" : "123", "moblie" : "456" } }  
RegExp正则表达式 /a-f/

<

>

---

---

---

---

---

---

---

---

3 MongoDB数据存储 (2)

NoSQL简介MongoDB简介数据存储局限性分析应用领域

>

MongoDB每个数据库都有自己的独立文件，如果开启 directoryperdb 选项，则每个库的文件会单独放在一个文件夹里  
每个库由一个名字空间文件和多个数据文件组成，名字空间文件以.ns结尾，数据文件按照数据量大小由0开始增长，第一个数据文件为64M，翻倍增长直至2G  
.ns文件保存数据库所有名字空间，每一个集合、索引都将占用一个命名空间以保存元数据。默认16M最大为2G，启动时可通过nssize参数选项更改ns文件大小以增加可创建集合及索引个数  
使用预分配数据文件方式来保证写入性能的稳定（可用-noprealloc关闭）。预分配在后台进行，并且每个预分配的文件都用0进行填充  
数据库启动后会产生一个6bytes的mongd.lock数据文件，用于防止同一个实例多次运行，非正常退出后需要删除此文件才能重启成功

<

>

---

---

---

---

---

---

---

---

3 MongoDB数据存储 (3)

NoSQL简介MongoDB简介数据存储局限性分析应用领域

- 名字空间文件保存的是一个hash table，保存了每个名字空间的信息，其内容为存储信息元数据，包括其大小，块数，第一块位置，最后一块位置，被删除的块的链表以及索引信息
- 每个数据文件被分成多个数据块(Extent)，用双向链表连接。头部包含了块的元数据：自己的位置，上一个和下一个块的位置以及块中第一条和最后一条记录的位置指针。剩下部分存储具体的数据记录，也是通过双向链接来进行连接
- 每条记录中头部包含记录元数据，如：头部长度，块中的位置偏移，上一条和下一条记录的位置。剩余是具体数据内容，如果padding > 1则实际长度大于数据内容本身长度
- 内存管理采用OS的MMAP内存映射机制，把数据文件映射到内存中，读操作可以起到缓存作用，写操作可由内存转化成顺序磁盘写入
- 操作系统虚拟内存管理器会托管所有磁盘操作，这样MongoDB内存管理实现会很简单，缺点是人工没有办法控制占多大内存

<

>

---

---

---

---

---

---

---

---

3 MongoDB局限性分析

NoSQL简介MongoDB简介数据存储局限性分析应用领域

- 在32位系统上，不支持大于2.5G的数据。
- 单个文档大小限制为 16 M
- 锁粒度太粗，MongoDB使用的是一把全局的读写锁
- 不支持join操作和事务机制，这个确实是非MongoDB要做的领域
- 对内存要求比较大，至少要保证热数据（索引，数据及系统其它开销）都能装进内存
- 用户权限方面比较弱，将机器部署在安全的内网环境中，尽量不要用权限，2.0后有改进
- 对于数组型的数据操作不够丰富

<

>

---

---

---

---

---

---

---

---

5 MongoDB应用领域

NoSQL简介MongoDB简介数据存储局限性分析应用领域

- 网站数据：MongoDB非常适合实时的插入，更新与查询，并具备网站实时数据存储所需的复制及高度伸缩性。
- 缓存：由于性能很高，MongoDB也适合作为信息基础设施的缓存层。在系统重启之后，由MongoDB搭建的持久化缓存层可以避免下层的数据源 过载。
- 大尺寸，低价值的数据：使用传统的关系型数据库存储一些数据时可能会比较昂贵，在此之前，很多时候程序员往往会选择传统的文件进行存储。
- 高伸缩性的场景：MongoDB非常适合由数十或数百台服务器组成的数据库。MongoDB对MapReduce引擎的内置支持。
- 用于对象及JSON数据的存储：MongoDB的BSON数据格式非常适合文档化格式的存储及查询。

<

>

---

---

---

---

---

---

---

---

↑

参考资料

【1】张恩,张广弟,兰磊.基于MongoDB的海量空间数据存储和并行[D]. 地理空间信息, 2014, 12(1): 46-48.

【2】申德荣,于戈,王习特,等.支持大数据管理的NoSQL系统研究综述[J]. 软件学报,2013, 24(8): 1786-1803

【3】邱铮. MongoDB 在煤炭行业 GIS 数据存储方面的应用[J].煤炭技术, 2013, 32(4): 201-202

【4】邱儒琼, 郑丽娜, 李兵. 基于MongoDB的电子地图瓦片数据存储和服务研究[J].地理空间信息, 2014, 10 (6): 155-157.

【5】李浩松, 朱欣焰, 李京伟, 等. WebGIS 空间数据分布式缓存技术研究 [J]. 武汉大学学报: 信息科学版, 2005, 30 (12):1092-1095.

【6】邵庆林, 陈柏堃. 使用文档型数据库 MongoDB 深度挖掘天气现象数据内在价值[J]. 计算机与网络, 2012, 38(16): 67-69.

【7】刘一梦. 基于 MongoDB 的云数据管理技术的研究与应用[D]. 北京: 北京交通大学, 2012. 8-11.

<

>

---

---

---

---

---

---

---