

MongoDB 与 MySQL 的适用场景：

MongoDB 的适用场景为：数据不是特别重要（例如通知，推送这些），数据表结构变化较为频繁，数据量特别大，数据的并发性特别高，数据结构比较特别（例如地图的位置坐标），这些情况下用 MongoDB，其他情况就还是用 MySQL，这样组合使用就可以达到最大的效率。

1.如果需要将 MongoDB 作为后端 db 来代替MySQL使用，即这里 MySQL 与 MongoDB 属于平行级别，那么，这样的使用可能有以下几种情况的考量：

(1)MongoDB 所负责部分以文档形式存储，能够有较好的代码亲和性，json 格式的直接写入方便。（如日志之类）

(2)从 data models 设计阶段就将原子性考虑于其中，无需事务之类的辅助。开发用如 nodejs 之类的语言来进行开发，对开发比较方便。

(3)MongoDB 本身的 failover 机制，无需使用如 MHA 之类的方式实现。

2.将 MongoDB 作为类似 Redis, memcache 来做缓存db，为 MySQL 提供服务，或是后端日志收集分析。考虑到 MongoDB 属于 No SQL 型数据库，SQL 语句与数据结构不如 MySQL 那么亲和，也会有很多时候将 MongoDB 做为辅助MySQL而使用的类 Redis memcache 之类的缓存db来使用。亦或是仅作日志收集分析。

MongoDB 有一个最大的缺点，就是它占用的空间很大，因为它属于典型空间换时间原则的类型。那么它的磁盘空间比普通数据库会浪费一些，而且到目前为止它还没有实现在线压缩功能，在 MongoDB 中频繁的进行数据增删改时，如果记录变了，例如数据大小发生了变化，这时候容易产生一些数据碎片，出现碎片引发的结果，一个是索引会出现性能问题。另外一个就是在一定的时间后，所占空间会莫名其妙地增大，所以要定期把数据库做修复，定期重新做索引，这样会提升MongoDB 的稳定性和效率。

1.MySQL 来自女儿的名字； MongoDB 来自 humongous 2.MySQL 使用 Table/Row/Column； MongoDB 使用 Collection/Document 3.MySQL 需要指定 table 的 schema； MongoDB的 collection 的每个 document 的 schema 可以自由修改 4.MySQL 支持 join； MongoDB 没有 join 5.MySQL 使用 SQL 语言； MongoDB 使用类似 JavaScript 的函数

命令对比 MongoDB 与 MySQL 命令对比 传统的关系数据库一般由数据库 (database)、表 (table)、记录 (record) 三个层次概念组成，MongoDB 是由数据库 (database)、集合 (collection)、文档对象 (document) 三个层次组成。MongoDB对于关系型数据库里的表，但是集合中没有列、行和关系概念，这体现了模式自由的特点。

MongoDB (文档型数据库)：提供可扩展的高性能数据存储 1、基于分布式文件存储 2、高负载情况下添加更多节点，可以保证服务器性能 3、将数据存储为一个文档 MongoDB 与 MySQL 的比较 1、稳定性 2、索引，索引放在内存中，能够提升随机读写的性能。如果索引不能完全放在内存，一旦出现随机读写比较高的时候，就会频繁地进行磁盘交换，MongoDB 的性能就会急剧下降 3、占用的空间很大，因为它属于典型空间换时间原则的类型。那么它的磁盘空间比普通数据库会浪费一些，而且到目前为止它还没有实现在线压缩功能，在 MongoDB 中频繁的进行数据增删改时，如果记录变了，例如数据大小发生了变化，这时候容易产生一些数据碎片，出现碎片引发的结果，一个是索引会出现性能问题，另外一个就是在一定的时间后，所占空间会莫名其妙地增大，所以要定期把数据库做修复，定期重新做索引，这样会提升MongoDB 的稳定性和效率。在最新的版本里，它已经在实现在线压缩，估计应该在 2.0版左右，应该能够实现在线压缩，可以在后台执行现在repair DataBase 的一些操作。如果那样，就解决了目前困扰我们的大问题。 4、MongoDB 对数据间的事务关系支持比较弱 5、运维不方便 MongoDB 相对于 MySQL 的优势

1. 适合那些对数据库具体数据格式不明确或者数据库数据格式经常变化的需求模型，而且对开发者十分友好。
2. 自带一个分布式文件系统，可以很方便地部署到服务器集群上。MongoDB 里有一个Shard的概念，就是为了服务器分片使用的。每增加一台Shard，MongoDB 的插入性能也会以接近倍数的方式增长，磁盘容量也很

可以很方便地扩充。

3. 自带了对map-reduce运算框架的支持，这也很方便进行数据的统计。类似于group by MongoDB 与 MySQL 命令对比 传统的关系数据库一般由数据库 (database)、表 (table)、记录 (record) 三个层次概念组成，MongoDB 是由数据库 (database)、集合 (collection)、文档对象 (document) 三个层次组成。MongoDB 对于关系型数据库里的表，但是集合中没有列、行和关系概念，这体现了模式自由的特点。

MongoDB 和 Redis 的区别：简介 MongoDB 更类似 MySQL，支持字段索引、游标操作，其优势在于查询功能比较强大，擅长查询 JSON 数据，能存储海量数据，但是不支持事务。

MySQL 在大数据量时效率显著下降，MongoDB 更多时候作为关系数据库的一种替代。

内存管理机制 Redis 数据全部存在内存，定期写入磁盘，当内存不够时，可以选择指定的 LRU 算法删除数据。

MongoDB 数据存在内存，由 linux 系统 mmap 实现，当内存不够时，只将热点数据放入内存，其他数据存在磁盘。

支持的数据结构 Redis 支持的数据结构丰富，包括 hash、set、list 等。MongoDB 数据结构比较单一，但是支持丰富的数据表达，索引，最类似关系型数据库，支持的查询语言非常丰富。

性能 二者性能都比较高，应该说都不会是瓶颈。

可靠性 二者均支持持久化。

集群 MongoDB 集群技术比较成熟，Redis 从 3.0 开始支持集群。

不适用的场景 Ø 需要使用复杂 sql 的操作 Ø 事务性系统

下面是 MongoDB 和 Redis 的对比图：

MySQL 与 Redis 的区别：

MySQL 是持久化存储，存放在磁盘里面，检索的话，会涉及到一定的 IO，为了解决这个瓶颈，于是出现了缓存，比如现在用的最多的 memcached(简称 mc)。首先，用户访问 mc，如果未命中，就去访问 MySQL，之后像内存和硬盘一样，把数据复制到 mc 一部分。Redis 和 mc 都是缓存，并且都是驻留在内存中运行的，这大大提升了高数据量 web 访问的访问速度。然而 mc 只是提供了简单的数据结构，比如 string 存储；Redis 却提供了大量的数据结构，比如 string、list、set、hashset、sorted set 这些，这使得用户方便了好多，毕竟封装了一层实用的功能，同时实现了同样的效果，当然用 Redis 而慢慢舍弃 mc。内存和硬盘的关系，硬盘放置主体数据用于持久化存储，而内存则是当前运行的那部分数据，CPU 访问内存而不是磁盘，这大大提升了运行的速度，当然这是基于程序的局部化访问原理。推理到 Redis + MySQL，它是内存+磁盘关系的一个映射，MySQL 放在磁盘，Redis 放在内存，这样的话，web 应用每次只访问 Redis，如果没有找到的数据，才去访问 MySQL。然而 Redis + MySQL 和内存+磁盘的用法最好是不同的。前者是内存数据库，数据保存在内存中，当然速度快。后者是关系型数据库，功能强大，数据访问也就慢。像 memcache，MongoDB，Redis，都属于 No SQL 系列。

不是一个类型的东西，应用场景也不太一样，还是要看你的需求来决定。

作者：喵星人Xcode 来源：CSDN 原文：<https://blog.csdn.net/CatStarXcode/article/details/79513425> 版权声明：本文为博主原创文章，转载请附上博文链接！