

# 好功夫

博客园

首页

新随笔

联系

订阅

管理

随笔 - 29

文章 - 0

评论 - 0

## memcached、redis、mongodb之间的区别

转载自：

<https://www.cnblogs.com/tuyile006/p/6382062.html>

### >>Memcached

**Memcached的优点：**

Memcached可以利用多核优势，单实例吞吐量极高，可以达到几十万QPS（取决于key、value的字节大小以及服务器硬件性能，日常环境中QPS高峰大约在4-6w左右）。适用于最大程度扛量。支持直接配置为session handle。

**Memcached的局限性：**

只支持简单的key/value数据结构，不像Redis可以支持丰富的数据类型。  
无法进行持久化，数据不能备份，只能用于缓存使用，且重启后数据全部丢失。  
无法进行数据同步，不能将MC中的数据迁移到其他MC实例中。  
Memcached内存分配采用Slab Allocation机制管理内存，value大小分布差异较大时会造成内存利用率降低，并引发低利用率时依然出现踢出等问题。需要用户注重value设计。

### >>Redis

**Redis的优点：**

支持多种数据结构，如 string（字符串）、list(双向链表)、dict(hash表)、set(集合)、zset(排序set)、hyperloglog（基数估算）  
支持持久化操作，可以进行aof及rdb数据持久化到磁盘，从而进行数据备份或数据恢复等操作，较好的防止数据丢失的手段。  
支持通过Replication进行数据复制，通过master-slave机制，可以实时进行数据的同步复制，支持多级复制和增量复制，master-slave机制是Redis进行HA的重要手段。  
单线程请求，所有命令串行执行，并发情况下不需要考虑数据一致性问题。  
支持pub/sub消息订阅机制，可以用来进行消息订阅与通知。  
支持简单的事务需求，但业界使用场景很少，并不成熟。

**Redis的局限性：**

Redis只能使用单线程，性能受限于CPU性能，故单实例CPU最高才可能达到5-6wQPS每秒（取决于数据结构，数据大小以及服务器硬件性能，日常环境中QPS高峰大约在1-2w左右）。  
支持简单的事务需求，但业界使用场景很少，并不成熟，既是优点也是缺点。  
Redis在string类型上会消耗较多内存，可以使用dict（hash表）压缩存储以降低内存耗用。

Mc和Redis都是Key-Value类型，不适合在不同数据集之间建立关系，也不适合进行查询搜索。比如redis的keys pattern这种匹配操作，对redis的性能是灾难。

### >>mongoDB

mongoDB 是一种文档性的数据库。先解释一下文档的数据库，即可以存放xml、json、bson类型系那个的数据。

这些数据具备自述性（self-describing），呈现分层的树状数据结构。redis可以用hash存放简单关系型数据。

*mongoDB 存放json格式数据。*

*适合场景：事件记录、内容管理或者博客平台，比如评论系统。*

#### 1.mongodb持久化原理

mongodb与mysql不同，mysql的每一次更新操作都会直接写入硬盘，但是mongo不会，做为内存型数据库，数据操作会先写入内存，然后再会持久化到硬盘中去，那么mongo是如何持久化的呢  
mongodb在启动时，专门初始化一个线程不断循环（除非应用crash掉），用于在一定时间周期内来从defer队列中获取要持久化的数据并写入到磁盘的journal(日志)和mongofile(数据)处，当然因为它不是在用户添加记录时就写到磁盘上，所以按mongodb开发者说，它不会造成性能上的损耗，因为看过代码

公告

昵称： 好功夫  
园龄： 1年3个月  
粉丝： 0  
关注： 1  
+加关注

2019年9月						
日	一	二	三	四	五	六
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5
6	7	8	9	10	11	12

搜索

找找看

谷歌搜索

常用链接

我的随笔  
我的评论  
我的参与  
最新评论  
我的标签

随笔档案

2018年11月(16)  
2018年10月(2)  
2018年8月(2)  
2018年7月(5)  
2018年6月(2)  
2018年5月(2)

阅读排行榜

- python json处理、集合操作、函数定义、文件读写(1168)
- python字典、list、字符串常见处理方法、元组、文件读写(272)
- yaml文件、装饰器、unittest(220)
- memcached、redis、mongodb之间的区别(97)

5. python之: map、filter、random、datetime, 写日志, 发邮件, 操作mysql、redis, MD5加密, 写excel(61)

发现, 当进行CUD操作时, 记录(Record类型)都被放入到defer队列中以供延时批量 (groupcommit) 提交写入, 但相信其中时间周期参数是个要认真考量的参数, 系统为90毫秒, 如果该值更低的话, 可能会造成频繁磁盘操作, 过高又会造成系统宕机时数据丢失过。

## 2.什么是NoSQL数据库? NoSQL和RDBMS有什么区别? 在哪些情况下使用和不使用NoSQL数据库?

NoSQL是非关系型数据库, NoSQL = Not Only SQL。

关系型数据库采用的结构化的数据, NoSQL采用的是键值对的方式存储数据。

在处理非结构化/半结构化的大数据时; 在水平方向上进行扩展时; 随时应对动态增加的数据项时可以优先考虑使用NoSQL数据库。

在考虑数据库的成熟度; 支持; 分析和商业智能; 管理及专业性等问题时, 应优先考虑关系型数据库。

## 3.MySQL和MongoDB之间最基本的区别是什么?

关系型数据库与非关系型数据库的区别, 即数据存储结构的不同。

## 4.MongoDB的特点是什么?

(1) 面向文档 (2) 高性能 (3) 高可用 (4) 易扩展 (5) 丰富的查询语言

## 5.MongoDB支持存储过程吗? 如果支持的话, 怎么用?

MongoDB支持存储过程, 它是javascript写的, 保存在db.system.js表中。

## 6.如何理解MongoDB中的GridFS机制, MongoDB为何使用GridFS来存储文件?

GridFS是一种将大型文件存储在MongoDB中的文件规范。使用GridFS可以将大文件分隔成多个小文档存放, 这样我们能够有效的保存大文档, 而且解决了BSON对象有限制的问题。

## 7.为什么MongoDB的数据文件很大?

MongoDB采用的预分配空间的方式来防止文件碎片。

## 8.当更新一个正在被迁移的块 (Chunk) 上的文档时会发生什么?

更新操作会立即发生在旧的块 (Chunk) 上, 然后更改会在所有权转移前复制到新的分片上。

## 9.MongoDB在A:{B,C}上建立索引, 查询A:{B,C}和A:{C,B}都会使用索引吗?

不会, 只会在A:{B,C}上使用索引。

## 10.如果一个分片 (Shard) 停止或很慢的时候, 发起一个查询会怎样?

如果一个分片停止了, 除非查询设置了“Partial”选项, 否则查询会返回一个错误。如果一个分片响应很慢, MongoDB会等待它的响应。

# >>Redis、Memcache和MongoDB的区别

从以下几个维度, 对redis、memcache、mongodb 做了对比,

### 1、性能

都比较高, 性能对我们来说应该都不是瓶颈

总体来讲, TPS方面redis和memcache差不多, 要大于mongodb

### 2、操作的便利性

memcache数据结构单一

redis丰富一些, 数据操作方面, redis更好一些, 较少的网络IO次数

mongodb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

### 3、内存空间的大小和数据量的大小

redis在2.0版本后增加了自己的VM特性, 突破物理内存的限制; 可以对key value设置过期时间 (类似memcache)

memcache可以修改最大可用内存, 采用LRU算法

mongodb适合大数据量的存储, 依赖操作系统VM做内存管理, 吃内存也比较厉害, 服务不要和别的服务在一起

### 4、可用性 (单点问题)

对于单点问题,

redis, 依赖客户端来实现分布式读写; 主从复制时, 每次从节点重新连接主节点都要依赖整个快照, 无增量复制, 因性能和效率问题,

所以单点问题比较复杂; 不支持自动sharding, 需要依赖程序设定一致hash 机制。

一种替代方案是, 不用redis本身的复制机制, 采用自己做主动复制 (多份存储), 或者改成增量复制的方式 (需要自己实现), 一致性问题 and 性能的权衡

Memcache本身没有数据冗余机制，也没必要；对于故障预防，采用依赖成熟的hash或者环状的算法，解决单点故障引起的抖动问题。

mongoDB支持master-slave,replicaset（内部采用paxos选举算法，自动故障恢复）,auto sharding机制，对客户端屏蔽了故障转移和切分机制。

## 5、可靠性（持久化）

对于数据持久化和数据恢复，

redis支持（快照、AOF）：依赖快照进行持久化，aof增强了可靠性的同时，对性能有所影响

memcache不支持，通常在做缓存,提升性能；

MongoDB从1.8版本开始采用binlog方式支持持久化的可靠性

## 6、数据一致性（事务支持）

Memcache 在并发场景下，用cas保证一致性

redis事务支持比较弱，只能保证事务中的每个操作连续执行

mongoDB不支持事务

## 7、数据分析

mongoDB内置了数据分析的功能(mapreduce),其他不支持

## 8、应用场景

redis：数据量较小的更性能操作和运算上

memcache：用于在动态系统中减少数据库负载，提升性能;做缓存，提高性能（适合读多写少，对于数据量比较大，可以采用sharding）

MongoDB:主要解决海量数据的访问效率问题



好功夫

关注 - 1

粉丝 - 0

+加关注

0

0

« 上一篇: [转载: awk详解](#)

» 下一篇: [Appium定位方式总结](#)

posted @ 2018-10-21 21:47 好功夫 阅读(96) 评论(0) 编辑 收藏

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问](#) 网站首页。

【推荐】超50万C++/C#源码：大型实时仿真组态图形源码

【推荐】零基础轻松玩转华为云产品，获赠礼加返百元大礼

【推荐】天翼云开学季，学生必备云套餐，每月仅需9块9

【推荐】华为云文字识别资源包重磅上市，1元万次限时抢购

【福利】git pull && cherry-pick 博客园&华为云百万代金券

### 相关博文:

- [redis, memcached, mongodb之间的区别](#)
- [Redis、MongoDB及Memcached的区别](#)
- [Redis、MongoDB及Memcached的区别](#)
- [Redis、MongoDB及Memcached的区别Redis（内存数据库）](#)
- [redis和mongodb的比较](#)