## redis和hbase的区别

### 读写性能：

HBase写快读慢，HBase的读取时长通常是几毫秒，而Redis的读取时长通常是几十微秒。性能相差非常大。

数据类型：（推荐学习：Redis视频教程）  
  
HBase和Redis都支持KV类型。但是Redis支持List、Set等更丰富的类型。

### 数据量：

Redis支持的数据量通常受内存限制，而HBase没有这个限制，可以存储远超内存大小的数据。

### 部署难易：

HBase部署需要依赖hadoop、zookeeper等服务，而Redis的部署非常简单。

### 数据可靠性：

HBase采用WAL，先记录日志再写入数据，理论上不会丢失数据。而Redis采用的是异步复制数据，在failover时可能会丢失数据。

### 应用场景：

HBase适合做大数据的持久存储，而Redis比较适合做缓存。如果数据丢失是不能容忍的，那就用只能用HBase；如果需要一个高性能的环境，而且能够容忍一定的数据丢失，那完全可以考虑使用Redis。

### 两者的结合：

HBase可以用来做数据的固化，也就是数据存储，做这个他非常合适。Redis适合做cache。可以用HBase+Redis实现数据仓库加缓存数据库，速度和扩展性都兼顾

## redis和mongodb的区别

Redis只能使用单线程，性能受限于CPU性能，MongoDB建议集群部署，更多的考虑到集群方案，Redis更偏重于进程顺序写入，虽然支持集群，也仅限于主-从模式。

面试题：

使用Redis有哪些好处？

(1) 速度快，因为数据存在内存中，类似于HashMap，HashMap的优势就是查找和操作的时间复杂度都是O(1)

(2) 支持丰富数据类型，支持string，list，set，sorted set，hash

(3) 支持事务，操作都是原子性，所谓的原子性就是对数据的更改要么全部执行，要么全部不执行

(4) 丰富的特性：可用于缓存，消息，按key设置过期时间，过期后将会自动删除

|  |  |  |  |
| --- | --- | --- | --- |
| **指标** | **MongoDB(v2.4.9)** | **Redis(v2.4.17)** | **比较说明** |
| 实现语言 | C++ | C/C++ | - |
| 协议 | BSON、自定义二进制 | 类Telnet | - |
| 性能 | 依赖内存，TPS较高 | 依赖内存，TPS非常高 | Redis优于MongoDB |
| 可操作性 | 丰富的数据表达、索引；最类似于关系数据库，支持丰富的查询语言 | 数据丰富，较少的IO | MongoDB优于Redis |
| 内存及存储 | 适合大数据量存储，依赖系统虚拟内存管理，采用镜像文件存储；内存占有率比较高，官方建议独立部署在64位系统（32位有最大2.5G文件限制，64位没有改限制） | Redis2.0后增加虚拟内存特性，突破物理内存限制；数据可以设置时效性，类似于memcache | 不同的应用角度看，各有优势 |
| 可用性 | 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 | 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；不支持自动sharding,需要依赖程序设定一致hash机制 | MongoDB优于Redis；单点问题上，MongoDB应用简单，相对用户透明，Redis比较复杂，需要客户端主动解决。（MongoDB 一般会使用replica sets和sharding功能结合，replica sets侧重高可用性及高可靠性，而sharding侧重于性能、易扩展） |
| 可靠性 | 从1.8版本后，采用binlog方式（MySQL同样采用该方式）支持持久化，增加可靠性 | 依赖快照进行持久化；AOF增强可靠性；增强可靠性的同时，影响访问性能 | MongoDB优于Redis |
| 一致性 | 不支持事务，靠客户端自身保证 | 支持事务，比较弱，仅能保证事务中的操作按顺序执行 | Redis优于MongoDB |
| 数据分析 | 内置数据分析功能（mapreduce） | 不支持 | MongoDB优于Redis |
| 应用场景 | 海量数据的访问效率提升 | 较小数据量的性能及运算 | MongoDB优于Redis |

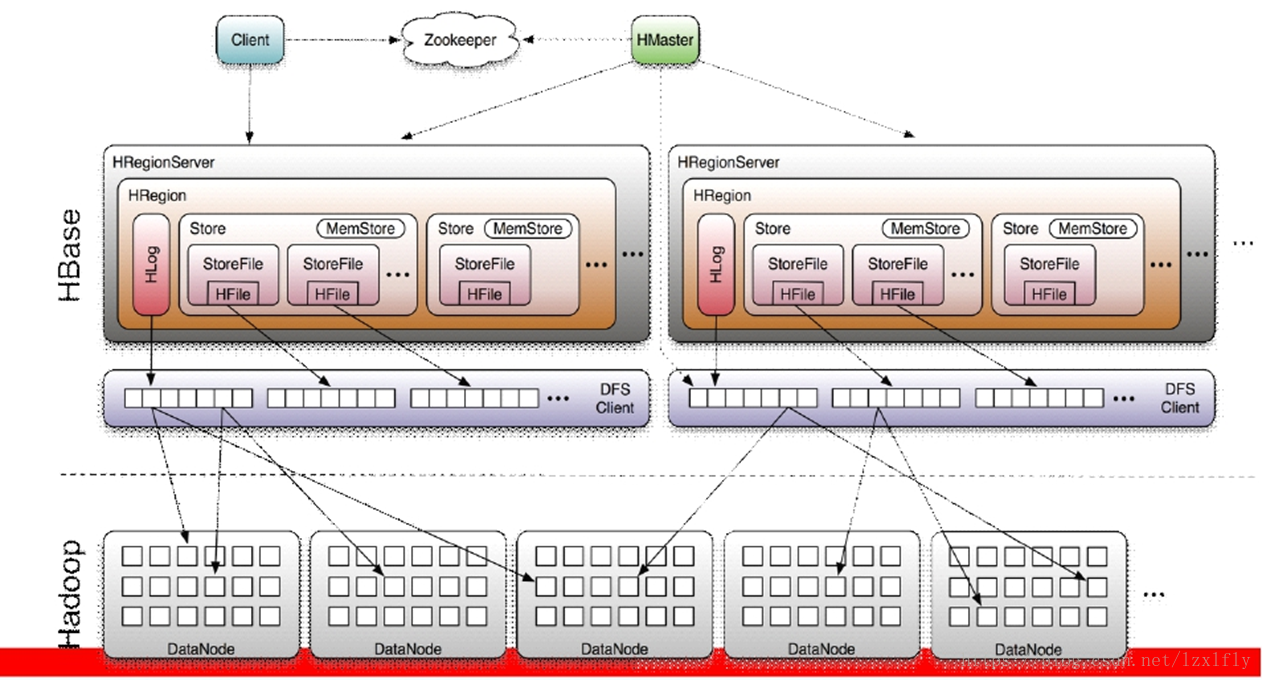
## HBase和MongoDB的区别

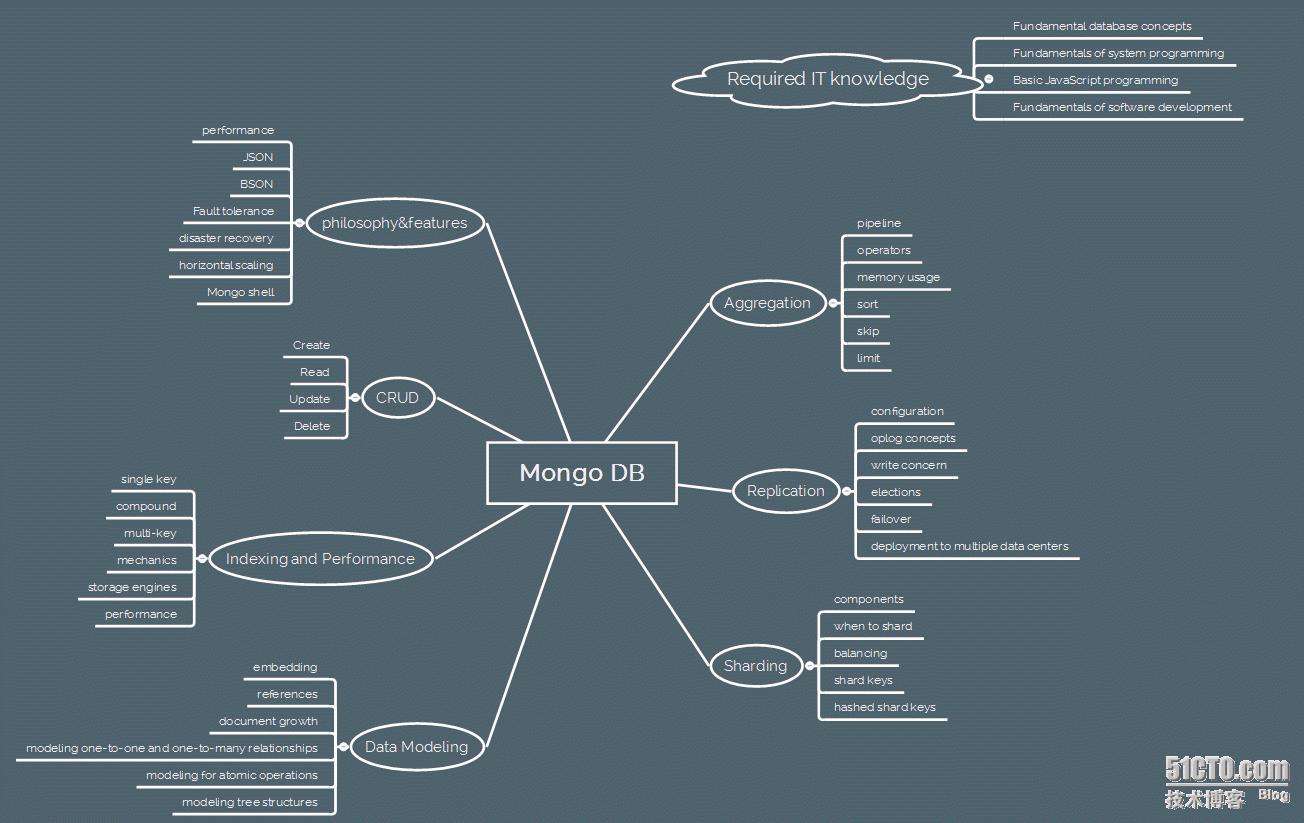
### 数据模型区别

HBase：表、行、列族、列、时间戳

MongoDB：Database<--Collection<--Document<--Field BSON是一种类json的一种二进制存储格式

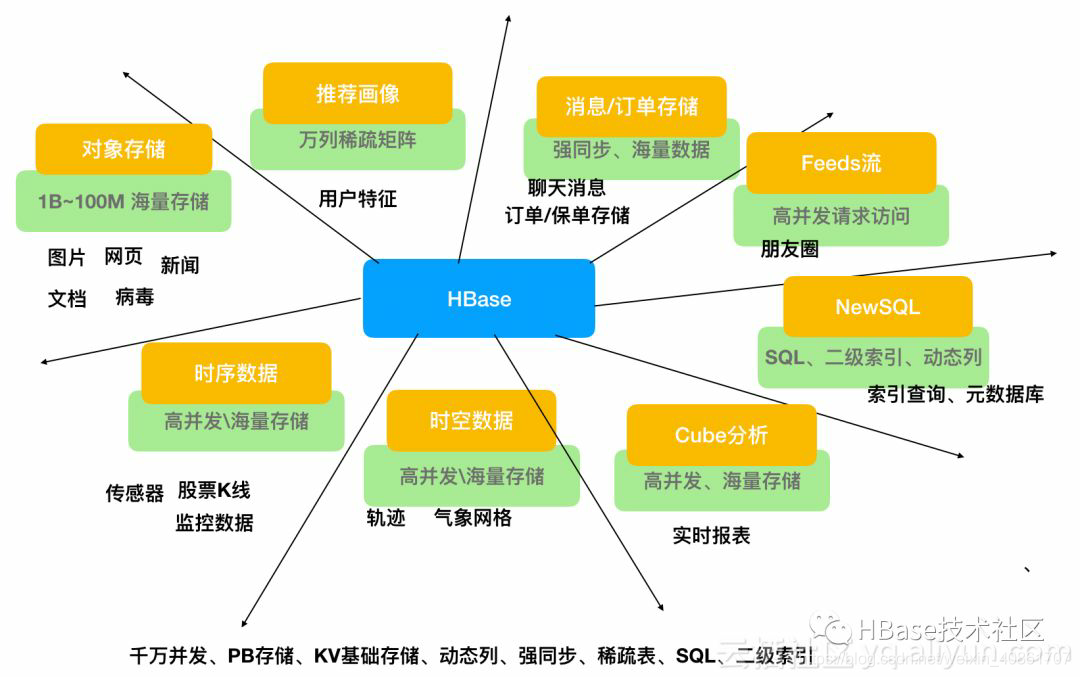
### 系统架构区别



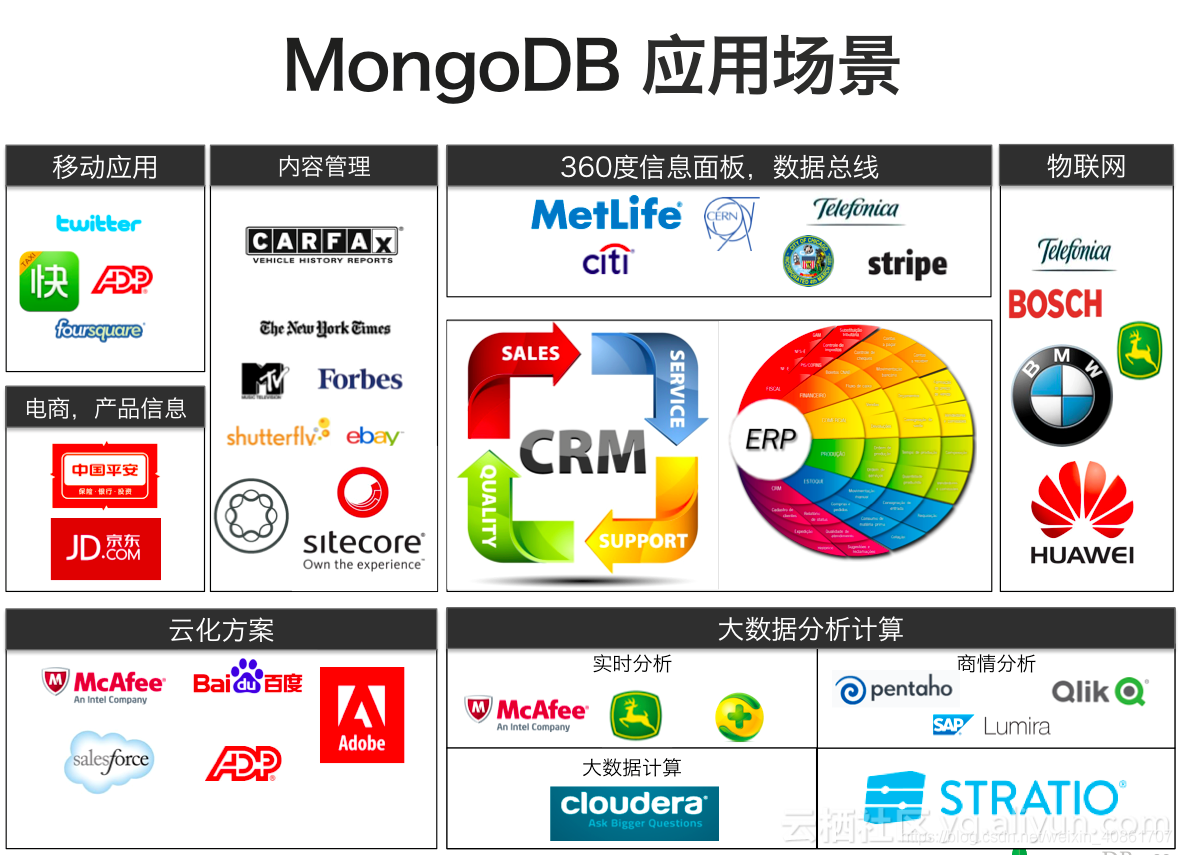


从查询的灵活度和各种条件的查询效率而言,MongoDB略胜一筹

### 应用场景区别



1. 对象存储：我们知道不少的头条类、新闻类的的新闻、网页、图片存储在HBase之中，一些病毒公司的病毒库也是存储在HBase之中
2. 时序数据：HBase之上有OpenTSDB模块，可以满足时序类场景的需求
3. 推荐画像：特别是用户的画像，是一个比较大的稀疏矩阵，蚂蚁的风控就是构建在HBase之上
4. 时空数据：主要是轨迹、气象网格之类，滴滴打车的轨迹数据主要存在HBase之中，另外在技术所有大一点的数据量的车联网企业，数据都是存在HBase之中
5. CubeDB OLAP：Kylin一个cube分析工具，底层的数据就是存储在HBase之中，不少客户自己基于离线计算构建cube存储在hbase之中，满足在线报表查询的需求
6. 消息/订单：在电信领域、银行领域，不少的订单查询底层的存储，另外不少通信、消息同步的应用构建在HBase之上
7. Feeds流：典型的应用就是xx朋友圈类似的应用
8. NewSQL：之上有Phoenix的插件，可以满足二级索引、SQL的需求，对接传统数据需要SQL非事务的需求



总结:

1. 从目前阿里云 MongoDB 云数据库上的用户看，MongoDB 的应用已经渗透到各个领域，比如游戏、物流、电商、内容管理、社交、物联网、视频直播等，以下是几个实际的应用案例。
2. 游戏场景，使用 MongoDB 存储游戏用户信息，用户的装备、积分等直接以内嵌文档的形式存储，方便查询、更新
3. 物流场景，使用 MongoDB 存储订单信息，订单状态在运送过程中会不断更新，以 MongoDB 内嵌数组的形式来存储，一次查询就能将订单所有的变更读取出来。
4. 社交场景，使用 MongoDB 存储存储用户信息，以及用户发表的朋友圈信息，通过地理位置索引实现附近的人、地点等功能
5. 物联网场景，使用 MongoDB 存储所有接入的智能设备信息，以及设备汇报的日志信息，并对这些信息进行多维度的分析
6. 视频直播，使用 MongoDB 存储用户信息、礼物信息等