

## 什么是

Redis 全免，BSD，一个 key-value

Redis 与其他 key-value 有以下三个：

Redis 久，以内存中保中，候以再加使

Redis 不仅仅 key-value，供 list, set, zset, hash 储

Redis 份，master-slave 份

Redis 优

- Redis 110000 /s,写 81000 /s

丰 - Redis 二制例 Strings, Lists, Hashes, Sets Ordered Sets 作

- Redis 作，么功么全不个作  
个作也事，MULTI EXEC 令

丰 - Redis publish/subscribe, , key

Redis 与其他 key-value 储 什么不？

Redis 为且供他们作，个不于其他  
Redis 于，

Redis 内存中但以久到，以不写内存，为  
不于件内存个优，上，  
内存中作，Redis 以做内存事，他  
们凑以加产，为他们不

## 的数据类型？

： Redis 五： string（串）， hash（）， list（列）， set（）， zsetsorted set：（）

们 中 string, hash 你 Redis 中， 加上下 几 HyperLogLog Geo Pub/Sub

你 Redis Module， 像 BloomFilter， RedisSearch， Redis-ML， 亮了

## 使用 有哪些好处？

- 1， 为 内 中， 似于 HashMap， HashMap 优 作 O1)
- 2 丰， string, list, set, Zset, hash
- 3 事， 作， 么全， 么全 不
- 4 丰： 于， key， 会 删

## 相比 有哪些优势？

- 1 Memcached 值 串， redis 作为其 代， 为丰
- 2 Redis Memcached 3 Redis 以 久 其
- 3 关 信公众： Java2B

## 与 的区别都有哪些？

- 1 储 Memecache 全 内 之中， 会， 不 内 Redis 份 上， 保 久
- 2 Memcache Redis
- 3 使 不 们之 以 与 之 信 不 Redis 了 VM 制， 为 函， 会

## 是单进程单线程的？

： Redis， redis 利 列 为串， 了传 串 制

## 一个字符串类型的值能存储最大容量是多少？

： 512M

## 持久化机制是什么？各自的优缺点？

Redis 提供两种持久化机制：RDB 和 AOF 机制。

1. RDB (Redis DataBase) 持久化：Redis 在运行过程中，将内存中的数据集写入到一个临时的文件中，然后持久化到指定的持久化路径上。Redis 使用 fork 的方式，在子进程中完成持久化操作，而主进程可以继续接收客户端的请求。

优点：

1. 持久化文件 dump.rdb，体积小，便于备份。

2. 持久化操作是异步的，对性能影响较小。

3. 持久化操作是异步的，主进程可以继续接收客户端的请求，不会阻塞任何 IO 操作。Redis 使用 fork 的方式，在子进程中完成持久化操作，而主进程可以继续接收客户端的请求。

AOF

：

1. AOF (Append-only file) 持久化：Redis 将客户端的每一个写操作（SET、DEL、INCR 等）都记录到一个日志文件中。Redis 使用 fork 的方式，在子进程中完成持久化操作，而主进程可以继续接收客户端的请求。

2. AOF 持久化文件 append-only file，Redis 将客户端的每一个写操作（SET、DEL、INCR 等）都记录到一个日志文件中。Redis 使用 fork 的方式，在子进程中完成持久化操作，而主进程可以继续接收客户端的请求。

优点：

1. 持久化文件 append-only file，Redis 将客户端的每一个写操作（SET、DEL、INCR 等）都记录到一个日志文件中。Redis 使用 fork 的方式，在子进程中完成持久化操作，而主进程可以继续接收客户端的请求。

2. 持久化操作是异步的，对性能影响较小。

3. AOF 持久化机制 rewrite AOF 文件，rewrite 之前（文件会先被重写），以删除其中一些命令（如 flushall）。

：

1. AOF 持久化文件 RDB 文件，且

2. 持久化操作是异步的，对性能影响较小。

## 常见性能问题和解决方案：

1. Master 节点不写内存，Master 写内存，save 命令 rdbSave 函数，会阻塞主进程，导致性能下降。

2. 持久化操作是异步的，对性能影响较小。

3. 为了主从复制，Master 节点 Slave 节点

4. 避免在主节点上增加从节点

5. 主从复制不，为，：Master <- Slave1

<- Slave2 <- Slave3... 便决，Slave Master Master 了，以刻 Slave1 做 Master，其他不

## 过期键的删除策略？

- 1 删：，创建一个 timer)。临，删作
- 2 删：任不，但从中，，删；，
- 3 删：，删于删，以个，则决

## 的回收策略（淘汰策略）

volatile-lru：从（server.db[i].expires）中使

volatile-ttl：从（server.db[i].expires）中

volatile-random：从（server.db[i].expires）中任

allkeys-lru：从（server.db[i].dict）中使

allkeys-random：从（server.db[i].dict）中任

no-eviction（）：

6 制，volatile allkeys 了从全，lru ttl 以 random 三不，再加上 no-eviction 不

使 则：

1 分，也分，分低，则使 allkeys-lru

2 分，也，则使 allkeys-random

## 为什么 需要把所有数据放到内存中？

：Redis 为了 到 写 到内 中， 写入 以  
redis 具 久 不 内 中， I/O 为严 redis  
内 便 今 ， redis 会 了 使 内 ， 则  
到内 值 不 入 值

## 的同步机制了解么？

：Redis 以使 主从 ， 从从 ， 主 做 bgsave， 修  
作 到内 buffer， rdb 件全 到 制 ， 制 rdb 像  
加 到内 加 ， 再 主 修 作 到 制 了

## 有什么好处，为什么要用 ？

： 以 IO 减为 ， 前 pipeline 令之 关 使  
redis-benchmark 候 以 redis QPS 值 个 pipeline  
令

## 是否使用过 集群，集群的原理是什么？

1. Redis Sentinel 于 ， master 会 slave 为master， 供
2. Redis Cluster 于 ， 个redis 内 不 ， 使 Cluster 分 储

## 集群方案什么 况下会导致整个集群不可用？

： A, B, C 三个 ， 制 况下， B 了， 么 个 会以  
为 5501-11000 个 不

## 支持的 客户端都有哪些？官方推荐用哪个？

： Redisson Jedis lettuce ， 使 Redisson

## 与 对比有什么优缺 ？

： Jedis Redis Java ， 其 API 供了 全 Redis 令 ； Redisson  
了分 Java ， Jedis ， 功 为 ， 不 串 作， 不  
事 分 Redis Redisson 促 使 Redis 关 分 ， 从 使  
力 中 业 上

## 如何设置密码及验证密码？

: config set requirepass 123456 : auth 123456

## 说说 哈希槽的概念？

: Redis 使 hash, 入了 , Redis 16384 个 , 个 key CRC16 16384 决 个 , 个 分 hash

## 集群的主从复制模型是怎样的？

: 为了使 分 分 信 况下 仍 , 以 使 了主从制 , 个 会 N-1 个 制 .

## 集群会有写操作丢失吗？为什么？

: Redis 不 保 , 中 件下 会丢 写 作

## 集群之间是如何复制的？

: 制

## 集群最大节点数是多少？

: 16384 个

## 集群如何选择数据库？

: Redis 前 做 , 0

## 怎么测试 的连通性？

: 使 ping 令

## 怎么理解 事务？

:

- 1) 事 个 作: 事 中 令 会 列 事 中, 不会 其他 令
- 2) 事 个 作: 事 中 令 么全 , 么全 不

## 事务相关的命令有哪几个？

: MULTI EXEC DISCARD WATCH

## 的过期时间和永久有效分别怎么设置？

: EXPIRE PERSIST 令

## 如何做内存优化？

: 使 列 (hashes), 列 (列 储 ) 使 内 ,  
以你 你 到 个 列 你 web 中 个 , 不  
为 个 , , , key, 个 信 储到  
列 . 关 信公众 : Java2B

## 回收进程如何工作的

: 个 了 令, 加了 Redi 内 使 况, 于 maxmemory  
制, 则 个 令 , 以 们不 内 制  
, 不 到 不 到 以下 个 令 内 使 (   
例 交 保 到 个 ), 不 久内 制 会 个内 使

## 都有哪些办法可以降低 的内存使用 况呢

: 你使 32 位 Redis 例, 以 利 Hash,list,sorted set,set  
为 况下 Key-Value 以 凑 到

## 的内存用完了会发生什么？

: 到 上 , Redis 写 令会 信 ( 但 令 以 ) 你  
以 Redis 使 制, Redis 到内 上 会冲刷 内

## 一个 实例最多能存放多少的 ? 他们最多能存放多少元素

: 上 Redis 以 232 keys, 且 中 了 , 个 例 了 2 亿 5  
万 keys 们 些 值 任何 list set sorted set 以 232 个元  
, Redis 储 中 内 值

## 证 里有 数据, 中只存 的数据, 如何保 中的数据都是 数据?

: Redis 内 上 到 候, 会 关 : Redis 供 6  
:

volatile-lru: 从 ( server.db[i].expires) 中

使

volatile-ttl: 从 ( server.db[i].expires) 中

volatile-random: 从 ( server.db[i].expires) 中任

allkeys-lru: 从 ( server.db[i].dict) 中 使

allkeys-random: 从 ( server.db[i].dict) 中任

no-eviction ( ) :

## 最适合的场景?

### 1 会 ( Session Cache)

使 Redis 会 ( session cache) Redis 会 其他 储 ( Memcached) 优 于: Redis 供 久 个不 严 , 信 全 丢 , 分人 会不 兴 , , 他们 会 ? , Redis 些 , 到 么 使 Redis 会 为人 业 Magento 也 供 Redis 件

### 2 全 ( FPC)

会 token 之 , Redis 供 便 FPC 到 , 使 了 Redis 例, 为 久 , 也不会 到 加 下 , 个 , 似 PHP FPC 再 以 Magento 为例, Magento 供 个 件 使 Redis 作为全 , WordPress , Pantheon 个 件 wp-redis, 个 件 你以 加 你

### 3 列

Redis 内 储 优 供 list set 作, 使 Redis 作为 个 列 使 Redis 作为 列使 作, 似于 ( Python) list push/pop 作 你 Google 中 “Redis queues”, 你 上 到 , 些 利 Redis 创 具, 以 列 例 , Celery 个 使 Redis 作为 broker, 你 以从

### 4, /

Redis 内 中 减 作 ( Set) ( Sorted Set) 也使 们 些 作 候 , Redis 供了 两 以, 们 从 中 到 前 10 个 - 们 之为“ user\_scores”, 们 像下 : , 假 你 你 分 做 你 分 , 你 : ZRANGE user\_scores 0 10 WITHSCORES Agora Games 个 例 , Ruby , 使 Redis 储 , 你 以 到

### 5 /

( 但 不 不 ) Redis / 功 / 使 人们 交 中使 , 作为 于 / , Redis / 功 !



假如 里面有 亿个 , 其中有 个 是以某个固定的已知的前缀开头的, 如果将它们全部找出来?

: 使 keys 令 以 出 key 列  
: 个 redis 万 上 业 供 , 使 keys 令 会 什么 ?  
个 候你 redis 关 个 : redis keys 令 会 ,  
上 会停 , 到 令 , 个 候 以使 scan 令, scan 令 以  
出 key 列 , 但 会 , 做 以了, 但  
体 会 keys 令

如果有大量的 需要设置同一时间过期, 一般需要注意什  
么?

: key 于 中, 到 个 , redis 会出  
上加 个 值, 使 分 些

使用过 做异步队列么, 你是怎么用的?

: 使 list 作为 列, rpush 产 , lpop lpop 候,  
sleep 会再

不 以不 sleep ?

list 个 令 blpop, 候, 会 住 到 到 不 产  
? 使 pub/sub 主 , 以 1:N 列

pub/sub 什么 ?

下 况下, 产 会丢 , 使 专业 列 RabbitMQ

redis 何 列?

估 你 你 上 , 么 么 但 你 克  
制, : 使 sortedset, 作为score, 内 作为 key zadd  
产 , zrangebyscore 令 N 之前 到 ,  
你 了 但 他不 刻你 了中 ,

## 使用过 分布式锁么，它是怎么回事？

先 setnx 争 ， 到之 ， 再 expire 加 个 了  
候 会 你 你 不 ， setnx 之 expire 之前 crash  
了， 会 么 ？  
候你 予 ： ， ， 个 不到 了 你  
， 作 刻， 像 下 你主 出 ， ： set 令  
， 个 以 setnx expire 令 ！ 会 ，  
： ， 不

## 如何实现集群中的 共享存储？

Session 上 ， 会到 们 上， 们 以  
传 sessionID, session, Session 不 况下 ( session 到了/  
)， 创 个 Session； 但 ， 们 下， 假 们 两  
A, B, 会 Nginx (别 也 )， ， Nginx  
A 上， A 创 了 session, SessionID ， 其他 ，  
， Nginx B, 于 B 上 sessionId  
session, 以会 创 个 session, 且再 个 sessionID ， ，  
们 以 下， 作 1/2 再 ， 不仅 体 别 ，  
会 上 session ， 加 力  
为了 决 下 session 共享 ， 共 4 决 ：

### 1. session

session Nginx 上， 与

### 1. session 制

session ， 创 修 ， 中 ， 使 上  
session

### 1. session 共享

session, 使 redis, memcached 4.session 久

session 储 中, 像 作 做 session

## 与 的区别?

- 1 Redis 不仅仅 k/v , 供 list, set, zset, hash 储  
memcache ,
- 2 Redis 久 , 以 内 中 保 中, 候 以再 加 使  
( PS: 久 rdb aof)