

info 命令展示了 redis 当前的一些实时状态

比较重要的一些参数：

Clients

connected_clients: 已连接客户端的数量

blocked_clients: 正在等待阻塞命令 (BLPOP、BRPOP、BRPOPLPUSH) 的客户端的数量

Memory

used_memory_rss_human: 以人类可读的格式, 从操作系统的角度, 返回 Redis 已分配的内存总量 (俗称常驻集大小)。这个值和 top、ps 等命令的输出一致。

Stats

keyspace_hits: 命中次数

keyspace_misses: 没命中次数

详细解释如下：

Server:

redis_version	Redis 服务器版本
redis_git_sha1	Git SHA1
redis_git_dirty	Git dirty flag
redis_build_id	Git dirty flag
redis_mode	运行模式, 单机或者集群
os	服务器的宿主操作系统
arch_bits	架构 (32 或 64 位)
multiplexing_api	Redis 所使用的事件处理机制
atomicvar_api	原子处理api
gcc_version	编译 Redis 时所使用的 GCC 版本
process_id	服务器进程的 PID
run_id	Redis 服务器的随机标识符 (用于 Sentinel 和集群)

tcp_port	TCP/IP 监听端口
uptime_in_seconds	自 Redis 服务器启动以来, 经过的秒数
uptime_in_days	自 Redis 服务器启动以来, 经过的天数
hz	edis内部调度 (进行关闭timeout的客户端, 删除过期key等等) 频率, 程序规定 serverCron每秒运行10次。
lru_clock	自增的时钟, 用于LRU管理,该时钟100ms(hz=10,因此每1000ms/10=100ms执行一次定时任务)更新一次。
executable	执行文件
config_file	配置文件路径

Clients:

connected_clients	已连接客户端的数量 (不包括通过从属服务器连接的客户端)
client_longest_output_list	当前连接的客户端当中, 最长的输出列表
client_biggest_input_buf	当前连接的客户端当中, 最大输入缓存
blocked_clients	正在等待阻塞命令 (BLPOP、BRPOP、BRPOPLPUSH) 的客户端的数量

Memory

used_memory	由 Redis 分配器分配的内存总量, 以字节 (byte) 为单位
used_memory_human	以人类可读的格式返回 Redis 分配的内存总量
used_memory_rss	从操作系统的角度, 返回 Redis 已分配的内存总量 (俗称常驻集大小)。这个值和 top、ps等命令的输出一致。
used_memory_rss_human	以人类可读的格式, 从操作系统的角度, 返回 Redis 已分配的内存总量 (俗称常驻集大小)。这个值和 top、ps等命令的输出一致。
used_memory_peak	redis的内存消耗峰值(以字节为单位)
used_memory_peak_human	以人类可读的格式返回redis的内存消耗峰值
used_memory_peak_perc	$(used_memory / used_memory_peak) * 100\%$
used_memory_overhead	Redis为了维护数据集的内部机制所需的内存开销, 包括所有客户端输出缓冲区、查询缓冲区、AOF重写缓冲区和主从复制的backlog。
used_memory_startup	Redis服务器启动时消耗的内存
used_memory_dataset	$used_memory - used_memory_overhead$

used_memory_dataset_perc	$100 * (\text{used_memory_dataset} / (\text{used_memory} - \text{used_memory_startup}))$
total_system_memory	整个系统内存
total_system_memory_human	以人类可读的格式, 显示整个系统内存
used_memory_lua	Lua脚本存储占用的内存
used_memory_lua_human	以人类可读的格式, 显示Lua脚本存储占用的内存
maxmemory	Redis实例的最大内存配置
maxmemory_human	以人类可读的格式, 显示Redis实例的最大内存配置
maxmemory_policy	当达到maxmemory时的淘汰策略
mem_fragmentation_ratio	$\text{used_memory_rss} / \text{used_memory}$
mem_allocator	内存分配器
active_defrag_running	表示没有活动的defrag任务正在运行, 1表示有活动的defrag任务正在运行 (defrag:表示内存碎片整理)
lazyfree_pending_objects	0表示不存在延迟释放 (也有资料翻译为惰性删除) 的挂起对象

Persistence

loading	服务器是否正在载入持久化文件
rdb_changes_since_last_save	离最近一次成功生成rdb文件, 写入命令的个数, 即有多少个写入命令没有持久化
rdb_bgsave_in_progress	服务器是否正在创建rdb文件
rdb_last_save_time	离最近一次成功创建rdb文件的时间戳。当前时间戳 - rdb_last_save_time = 多少秒未成功生成rdb文件
rdb_last_bgsave_status	最近一次rdb持久化是否成功
rdb_last_bgsave_time_sec	最近一次成功生成rdb文件耗时秒数
rdb_current_bgsave_time_sec	如果服务器正在创建rdb文件, 那么这个域记录的就是当前的创建操作已经耗费的秒数
rdb_last_cow_size	RDB过程中父进程与子进程相比执行了多少修改(包括读缓冲区, 写缓冲区, 数据修改等)。
aof_enabled	是否开启了aof

aof_rewrite_in_progress	标识aof的rewrite操作是否在进行中
aof_rewrite_scheduled	rewrite任务计划, 当客户端发送bgrewriteaof指令, 如果当前rewrite子进程正在执行, 那么将客户端请求的bgrewriteaof变为计划任务, 待aof子进程结束后执行rewrite
aof_last_rewrite_time_sec	最近一次aof rewrite耗费的时长
aof_current_rewrite_time_sec	如果rewrite操作正在进行, 则记录所使用的时间, 单位秒
aof_last_bgrewrite_status	上次bgrewriteaof操作的状态
aof_last_write_status	上次aof写入状态
aof_last_cow_size	AOF过程中父进程与子进程相比执行了多少修改(包括读缓冲区, 写缓冲区, 数据修改等)。

Stats

total_connections_received	新创建连接个数, 如果新创建连接过多, 过度地创建和销毁连接对性能有影响, 说明短连接严重或连接池使用有问题, 需调研代码的连接设置
total_commands_processed	redis处理的命令数
instantaneous_ops_per_sec	redis当前的qps, redis内部较实时的每秒执行的命令数
total_net_input_bytes	redis网络入口流量字节数
total_net_output_bytes	redis网络出口流量字节数
instantaneous_input_kbps	redis网络入口kps
instantaneous_output_kbps	redis网络出口kps
rejected_connections	拒绝的连接个数, redis连接个数达到maxclients限制, 拒绝新连接的个数
sync_full	主从完全同步成功次数
sync_partial_ok	主从部分同步成功次数
sync_partial_err	主从部分同步失败次数

expired_keys	运行以来过期的key的数量
expired_stale_perc	过期的比率
expired_time_cap_reached_count	过期计数
evicted_keys	运行以来剔除(超过了maxmemory后)的key的数量
keyspace_hits	命中次数
keyspace_misses	没命中次数
pubsub_channels	当前使用中的频道数量
pubsub_patterns	当前使用的模式的数量
latest_fork_usec	最近一次fork操作阻塞redis进程的耗时数, 单位微秒
migrate_cached_sockets	是否已经缓存了到该地址的连接
slave_expires_tracked_keys	从实例到期key数量
active_defrag_hits	主动碎片整理命中次数
active_defrag_misses	主动碎片整理未命中次数
active_defrag_key_hits	主动碎片整理key命中次数
active_defrag_key_misses	主动碎片整理key未命中次数

Replication

role	实例的角色, 是master or slave
connected_slaves	连接的slave实例个数
master_replid	主实例启动随机字符串
master_replid2	主实例启动随机字符串2
master_repl_offset	主从同步偏移量,此值如果和上面的offset相同说明主从一致没延迟, 与master_replid可被用来标识主实例复制流中的位置。
second_repl_offset	主从同步偏移量2,此值如果和上面的offset相同说明主从一致没延迟
repl_backlog_active	复制积压缓冲区是否开启
repl_backlog_size	复制积压缓冲大小
repl_backlog_first_byte_offset	复制缓冲区里偏移量的大小
repl_backlog_histlen	此值等于 master_repl_offset - repl_backlog_first_byte_offset,该值不会超过repl_backlog_size的大小

CPU

used_cpu_sys	将所有redis主进程在核心态所占用的CPU时求和累计起来
used_cpu_user	将所有redis主进程在用户态所占用的CPU时求和累计起来
used_cpu_sys_children	将后台进程在核心态所占用的CPU时求和累计起来
used_cpu_user_children	将后台进程在用户态所占用的CPU时求和累计起来

Cluster

cluster_enabled	实例是否启用集群模式
-----------------	------------

Keyspace

db0	db0的key的数量,以及带有生存期的key的数,平均存活时间
-----	---------------------------------