



链滴

Redis 过期策略和内存淘汰机制

作者: [jianzh5](#)

原文链接: <https://ld246.com/article/1597153204275>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Redis缓存作为提高系统性能最好的方式相信大家对其一定不陌生，各位作为秃头老码农不仅需要掌握edis的基础用法还得更了解Redis的相关原理，比如Redis过期策略和内存淘汰机制。

大家都知道，Redis缓存使用的是内存资源，虽然缓存服务器会配置比较高的内存资源，但如果对于Redis中的缓存数据我们不管不顾，内存资源总有耗尽的时候，这时缓存服务器就无法再对外提供服务。我们要用有限的服务器资源支撑更多的业务服务，就必须要让那些访问频率不高的缓存删除掉，为的缓存腾出内存空间。

Redis主要通过两种方式相互配合来实现键值的清理，即：过期策略和内存淘汰机制。

内存淘汰策略

当Redis节点分配的内存使用到达最大值以后，为了继续提供服务，Redis会启动内存淘汰策略，在Redis 4.0之前主要是以下六种淘汰策略：

- noeviction：不淘汰任何数据，当内存不足时，执行缓存新增操作会报错，这种策略下可以保证数据不丢失，它也是Redis默认的内存淘汰策略。
- allkeys-lru：淘汰整个键值中最久未使用的键值，这也就是我们常说的LRU算法。
- allkeys-random：随机淘汰任意键值。
- volatile-lru：淘汰所有设置了过期时间的键值中最久未使用的键值。
- volatile-random：随机淘汰设置了过期时间的任意键值。
- volatile-ttl：优先淘汰设置了过期时间中更早过期的键值。

通过上面的内存淘汰策略可以看出，以 **allkeys-** 开头的表示从所有key中进行数据淘汰，而以 **volatile-** 开头的会从设置了过期时间的key中进行数据淘汰。

而在Redis 4.0版本中又新增了2种淘汰策略：

- allkeys-lfu，淘汰整个键值中最少使用的键值，这也就是我们常说的LRU算法。
- volatile-lfu，淘汰所有设置了过期时间的键值中最少使用的键值。

LRU (Least Recently Used, 最近最少使用)，根据最近被使用的时间，离当前最远的数据优先被淘汰；

LFU (Least Frequently Used, 最不经常使用)，在一段时间内，缓存数据被使用次数最少的会被淘汰。

大家可以在 redis.conf 配置文件中修改配置项 **maxmemory-policy**，将其修改成需要设置的类型即可。

缓存过期策略

内存淘汰策略是当内存不够用时才会触发的一种机制，是缓存服务层面的操作，而过期策略定义的是缓存数据何时失效。我们在使用Redis的时候经常会给redis的key设置一个过期时间如：**EXPIRE key 30**，过期策略就是指当Redis中缓存的key过期了，Redis如何处理。

对于已经过期的数据，Redis将使用两种策略搭配使用来删除这些过期的键值，分别是**惰性删除**，**定期删除**。

惰性删除

惰性删除 是指 Redis 服务器不主动删除过期的键值，而是在客户端要获取某个键值时，Redis会先去测一下这个key是否已经过期，如果没有过期则返回给客户端，如果已经过期了，那么Redis会删除这key并返回null给客户端。

惰性删除可以解决一些过期了，但没被定期删除随机抽取到的key。但有些过期的key既没有被随机抽，也没有被客户端访问，就会一直保留在数据库，占用内存，长期下去可能会导致内存耗尽。所以Redis提供了内存淘汰机制来解决这个问题。

惰性删除的优点是不会浪费太多的系统资源，只是在每次访问时才检查键值是否过期。缺点是像上面的删除过期键不及时，造成了一定的空间浪费。

定期删除

定期删除 是指 Redis 服务器每隔一段时间会检查一下缓存数据库，看看是否有过期键可以被清除。默认情况下 Redis 定期检查的频率是每秒扫描 10 次，用于定期清除过期键。当然此值还可以通过配置文件进行设置，在 redis.conf 中修改配置 “hz” 即可，默认的值为 “hz 10” 。

因为key太多，如果全盘扫描所有的key会非常耗性能，所以定期删除是随机抽取一些key来删除。这就有可能删除不完，需要惰性删除配合。