阿里云 Redis 开发规范深入解读，别只会 set、get！

芋道源码 2020-02-15

以下文章来源于阿飞的博客 ，作者阿飞的博客

阿飞的博客

主要用来分享Java服务端相关的所有技术，坚持原创为主，已累计100+篇原创，包括但...

点击上方“芋道源码”，选择“设为星标”

做积极的人，而不是积极废人！

源码精品专栏

- 原创 | Java 2019 超神之路，很肝~
- 中文详细注释的开源项目
- RPC 框架 Dubbo 源码解析
- 网络应用框架 Netty 源码解析
- 消息中间件 RocketMQ 源码解析
- 数据库中间件 Sharding-JDBC 和 MyCAT 源码解析
- 作业调度中间件 Elastic-Job 源码解析
- 分布式事务中间件 TCC-Transaction 源码解析
- Eureka 和 Hystrix 源码解析
- Java 并发源码

来源：阿飞的博客

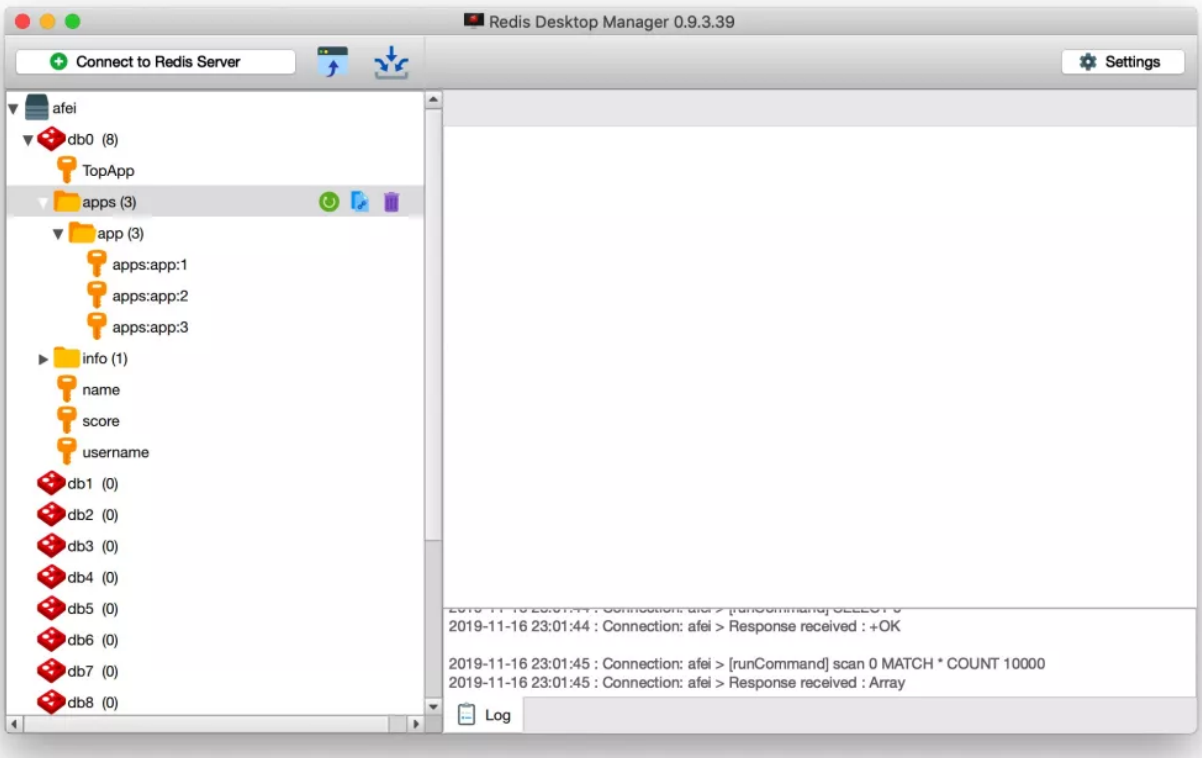
jianshu.com/p/932b69c8aa2f

- Key命名设计：可读性、可管理性、简介性
- Value设计：拒绝bigkey
- 控制Key的生命周期：设定过期时间
- 时间复杂度为O(n)的命令需要注意N的数量
- 禁用命令：KEYS、FLUSHDB、FLUSHALL等
- 推荐使用批量操作提升操作效率
- monitor命令控制使用时间
- 写在最后

Key命名设计：可读性、可管理性、简介性

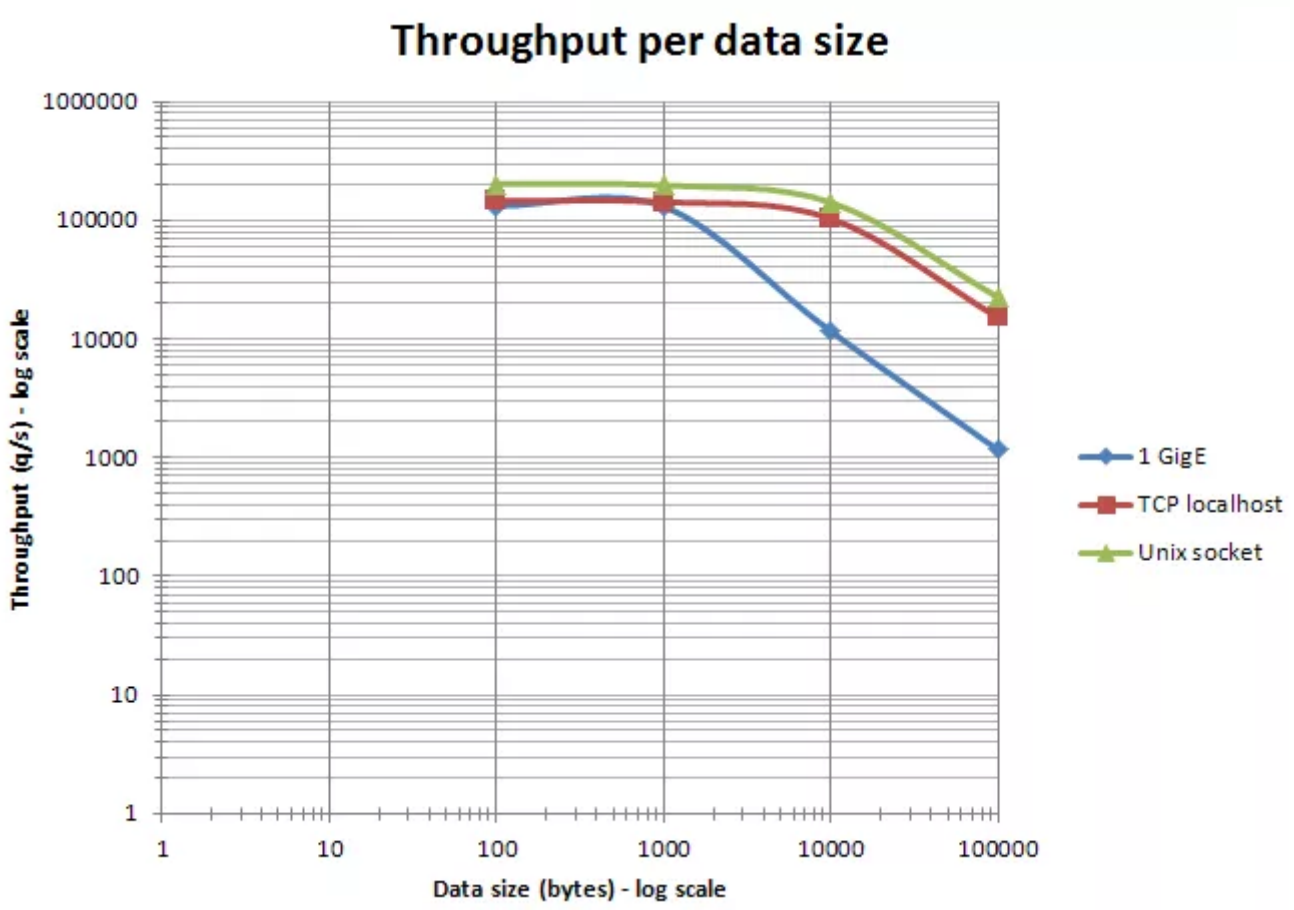
规范建议使用冒号即:进行分割拼接，因为很多Redis客户端是根据冒号分类的。比如有几个Key：apps:app:1、apps:app:2和apps:app:3。Redis Desktop Manager能自动归类到apps目

录下。如下图所示：



Value设计：拒绝bigkey

规范建议String类型的Value控制在10KB范围以内。这是因为Redis随着Value不断增长，在超过10KB后，有一个非常奇妙的性能拐点，如下图所示（图片来自Redis官网：http://redis.cn/topics/benchmarks.html：



假设内网带宽是千兆网卡，即1000MB。假设你的Redis中有一个大Key的Value长度是10KB，并且这个Key的QPS是10W，那么这一个Key就会把带宽打满：10KB*100000=1000MB。

控制Key的生命周期：设定过期时间

尽可能对每一个Key都设置过期时间，这个是非常有益处的。否则，你想想一下，半年以后，一年以后，你的Redis集群中有上百G甚至更多的数据，谁都不知道这些数据哪些是有价值的，哪些已经成为垃圾。如果你的每个Key都设置了过期时间，那么就不会出现这个问题了。集群在运行过程中，或自动淘汰那些已经不再使用的垃圾缓存数据。

时间复杂度为O(n)的命令需要注意N的数量

这个建议的意思是，以List类型为例，LINDEX、LREM等命令的时间复杂度就是O(n)。也就是说，随着List中元素数量越来越多，这些命令的性能越来越差。而Redis又是单线程的，如果出现一个慢命令，会导致在这个命令之后执行的命令耗时也会增长，这是使用Redis的大忌。

事实上这也是JDK8为什么要对HashMap进行链条冲突优化：当entry数量不少于64时，如果冲突链表长度达到8，就会将其转成红黑树。因为链表长度越长，性能会越来越差。

禁用命令：KEYS、FLUSHDB、FLUSHALL等

这些命令应该在搭建Redis环境的时候就要禁用掉（在config配置文件中通过rename-command禁用）。FLUSHDB和FLUSHALL这两个命令会清空数据，后果可想而知。

至于KEYS命令，还记得那个由于使用这个命令导致几百万损失的案例嘛？而且，这个命令的不当使用导致的损失，会随着你的业务并大越大价值越大而导致损失越大：

推荐使用批量操作提升操作效率

批量命令主要分为两类，原生命令和非原生命令：

- 原生命令包括：例如mget、mset、hmget、hmset、LPUSH key value集合等。
- 非原生命令包括：Pipeline。

合理使用这些命令对操作性能提升是**极其巨大** 的，尤其在单机Redis或者Sentinel模式下。因为这两种架构不涉及跨Slot，Redis集群性能也有提升，但是使用会受到一些限制，例如不支持跨Slot的操作。

当然批量虽好，但不要贪多。俗话说的好，贪多嚼不烂。一般不要超过1000，具体限制还与操作数据大小有关。

monitor命令控制使用时间

monitor命令一般是用来观察redis服务端都在执行哪些命令并实时输出。例如在其他redis-cli中执行两个set命令，在monitor中监控结果如下：

```
afeiMacBook-Pro:redis-3.2.11 afei$ src/redis-cli monitor
OK
1573915193.053188 [0127.0.0.1:55357] "COMMAND"
```

```
1573915197.087383 [0127.0.0.1:55357] "set""name""afei"
1573915217.938838 [0127.0.0.1:55357] "set""公 众 号""阿飞的博客"
```

之所以规范建议控制monitor命令的使用时间，是因为随着monitor命令执行时间越来越长，会导致越来越多的数据积压在输出缓冲区，从而导致输出缓冲区占用内存越来越大。而且，这种影响会由于Redis并发越高，而更加放大。关于这个问题，美团有一个很经典的案例，感兴趣的同学可以搜索关键词："**美团在REDIS上踩过的一些坑-3.REDIS内存占用飙升**"。

写在最后

总而言之，任何一门技术都有利有弊，**技术的世界里没有银弹** 。所以，我们对使用到生产环境的任何一个技术，都要非常熟悉：知道它所擅长的和它的弱点，这样才能结合自己的项目特点，设计出更合理的架构，编写出最合理的代码，不给生产环境造成影响，不给公司带来损失 -- 千万不要成为那个执行KEYS命令导致给公司造成金钱损失的人！

欢迎加入我的知识星球，一起探讨架构，交流源码。加入方式，**长按下方二维码噢**：

已在知识星球更新源码解析如下：

《精尽面试题（附答案）》	《精尽学习指南（附视频）》
01. Dubbo 面试题	00. 精尽学习指南 —— 路线
02. Netty 面试题	01. Dubbo 学习指南
03. Spring 面试题	02. Netty 学习指南
04. Spring MVC 面试题	03. Spring 学习指南
05. Spring Boot 面试题	04. Spring MVC 学习指南
06. Spring Cloud 面试题	05. Spring Boot 学习指南
07. MyBatis 面试题	06. Spring Cloud 学习指南
08. 消息队列面试题	07. MyBatis 学习指南
09. RocketMQ 面试题	08. RocketMQ 学习指南
10. RabbitMQ 面试题	09. RabbitMQ 学习指南
11. Kafka 面试题	10. Kafka 学习指南
12. 缓存面试题	11. Redis 学习指南
13. Redis 面试题	12. MySQL 学习指南
14. MySQL 面试题	13. MongoDB 学习指南
15.【分库分表】面试题	14. Elasticsearch 学习指南
16.【分布式事务】面试题	15. 设计模式学习指南
17. Elasticsearch 面试题	16. Java【基础】学习指南
18. MongoDB 面试题	17. Java【并发】学习指南
19. 设计模式面试题	18. Java【虚拟机】学习指南
20. Java【基础】面试题	19. Linux 学习指南
21. Java【集合】面试题	20. 数据结构与算法学习指南
22. Java【并发】面试题	21. 计算机网络学习指南
23. Java【虚拟机】面试题	22. Maven 学习指南
24. Linux 面试题	23. Jenkins 学习指南
25. Git 面试题	24. Git 学习指南
26. 计算机网络面试题	25. Intellij IDEA 学习指南
27. Maven 面试题	26. Docker 学习指南
28. Jenkins 面试题	27. Kubernetes 学习指南
29. Zookeeper 面试题	28. Zookeeper 学习指南
30. Nginx 面试题	29. Nginx 学习指南

《Dubbo 源码解析》

01. 调试环境搭建
02. 项目结构一览
03. 配置 Configuration
04. 核心流程一览
05. 拓展机制 SPI
06. 线程池 ThreadPool
07. 服务暴露 Export
08. 服务引用 Refer
09. 注册中心 Registry
10. 动态编译 Compile
11. 动态代理 Proxy
12. 服务调用 Invoke
13. 调用特性
14. 过滤器 Filter
15. NIO 服务器
16. P2P 服务器
17. HTTP 服务器
18. 序列化 Serialization
19. 集群容错 Cluster
20. 优雅停机 Shutdown
21. 日志适配 Logging
22. 状态检查 Status
23. 监控中心 Monitor
24. 管理中心 Admin
25. 运维命令 QOS
26. 链路追踪 Tracing
27. Spring Boot 集成
28. Spring Cloud 集成

... 一共 **73+** 篇

《Spring Cloud 源码解析》

01. 网关 Spring Cloud Gateway 25 篇
02. 注册中心 Eureka 23 篇
03. 熔断器 Hystrix 9 篇
04. 配置中心 Apollo 32 篇
05. 链路追踪 SkyWalking 38 篇
06. 调度中心 Elastic Job 24 篇

《Netty 源码解析》

01. 调试环境搭建
02. NIO 基础
03. Netty 简介
04. 启动 Bootstrap
05. 事件轮询 EventLoop
06. 通道管道 ChannelPipeline
07. 通道 Channel
08. 字节缓冲区 ByteBuf
09. 通道处理器 ChannelHandler
10. 编解码 Codec
11. 工具类 Util

... 一共 **61+** 篇

《MyBatis 源码解析》

01. 调试环境搭建
02. 项目结构一览
03. MyBatis 初始化
04. SQL 初始化
05. SQL 执行
06. 插件体系
07. Spring 集成

... 一共 **34+** 篇

最近更新《芋道 SpringBoot 2.X 入门》系列，已经 20 余篇，覆盖了 MyBatis、Redis、MongoDB、ES、分库分表、读写分离、SpringMVC、Webflux、权限、WebSocket、Dubbo、RabbitMQ、RocketMQ、Kafka、性能测试等等内容。

提供近 3W 行代码的 SpringBoot 示例，以及超 4W 行代码的电商微服务项目。

获取方式：点“**在看**”，关注公众号并回复 **666** 领取，更多内容陆续奉上。

如果你喜欢这篇文章，喜欢，转发。
生活很美好，明天见(。·ω·。)ﾉ♡

阅读原文