# Redis开发与运维

# 初始Redis

## 1.2 Redis特性

Redis全称：Remote Dictionary Server

8个重要特性

1. 速度快：所有数据都是存放在内存中的、是用C语言实现的、使用了单线程架构
2. 基于键值对的数据结构服务器：字符串、哈希、列表、集合、有序集合，在字符串基础上演变出了位图(Bitmaps)和HyperLogLog的数据结构，Redis3.2版本中加入GEO(地理信息定位)的功能。
3. 丰富的功能：键过期功能、发布订阅功能、支持Lua脚本功能、简单的事务功能、提供了流水线(Pipeline)功能
4. 简单稳定：代码5万行，相对其他产品较少，使用单线程模型。模型简单
5. 客户端语言多：Java，PHP，Python、C、C++、Nodejs
6. 持久化：两种持久化方式：RDB和AOF。
7. 主从复制：
8. 高可用和分布式：Redis2.8高可用实现Redis Sentinel。Redis3.0提供了分布式实现Redis Cluster

## 1.3 Redis使用场景

### 1.3.1 可以做什么

1. 缓存
2. 排行榜系统：列表和有序集合数据结构
3. 计数器应用：视频播放数，颠勺网站浏览数
4. 社交网络：赞/踩、粉丝、共同好友/喜好、推送、下拉刷新等
5. 消息对列系统

### 1.3.1 不可以做什么

1. 大数据规模不适合：每天有几亿的用户行为数据
2. 冷数据不适合：不经常使用的数据

# API的理解和使用

## 2.1 预备

### 2.1.1 全局命令

1. 查看所有键

keys \*

1. 键总数：dbsize

dbsize是直接获取Redis内置的键中暑变量，时间复杂度是O(1)，keys遍历所有键，时间复杂度是O(n)，当Redis保存大量键时，线上环境禁止使用

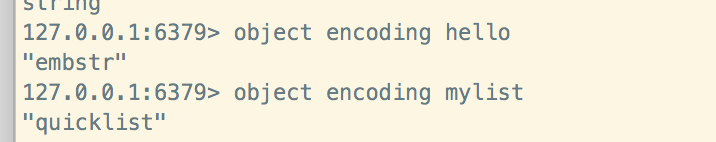
1. 检查键是否存在 exists key
2. 删除键：del key [key....]
3. 键过期：expire key seconds

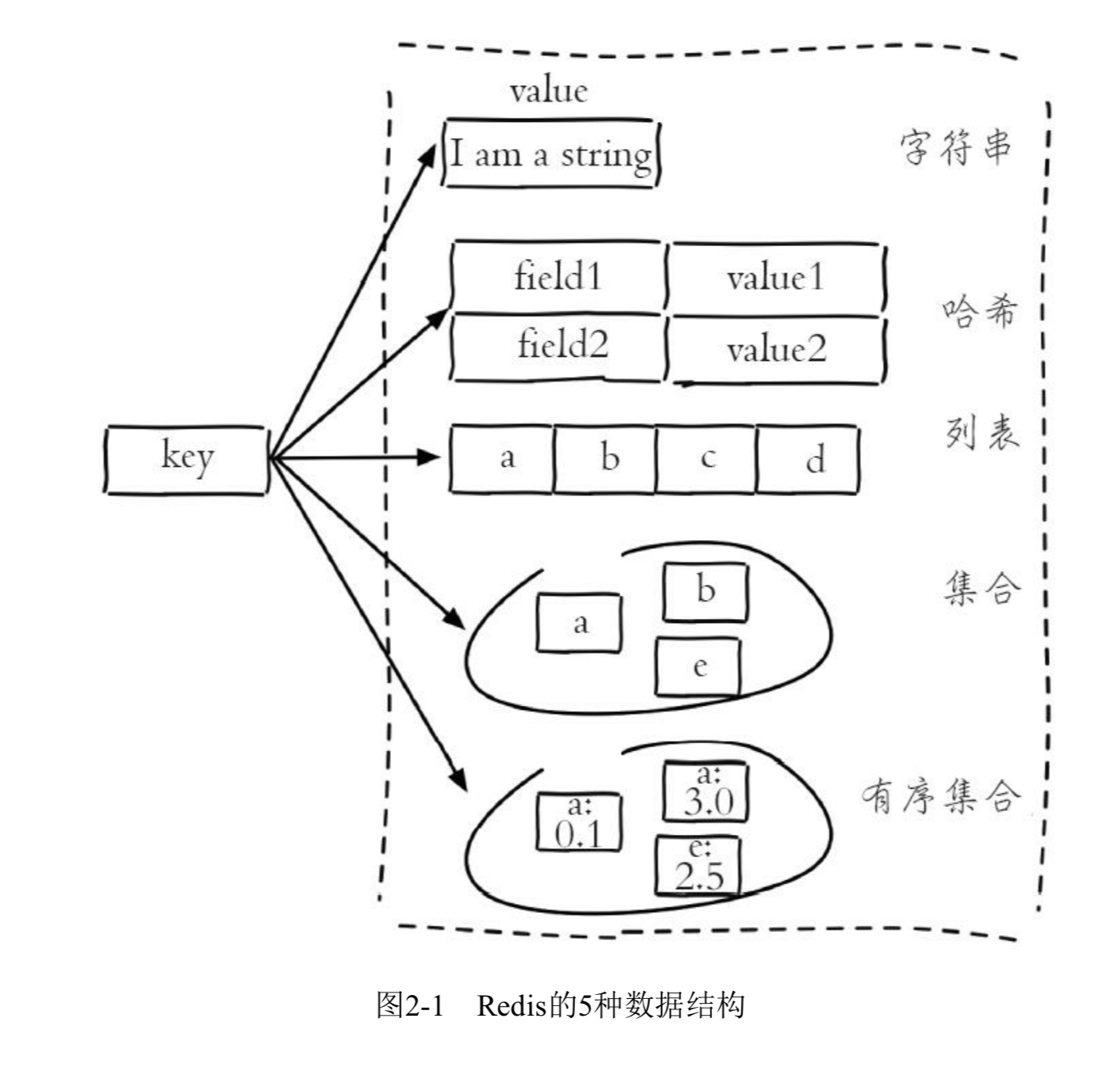
tt命令返回键的剩余过期时间，大于等于0的整数：剩余的过期时间，-1：没设置过期时间，-2：键不存在

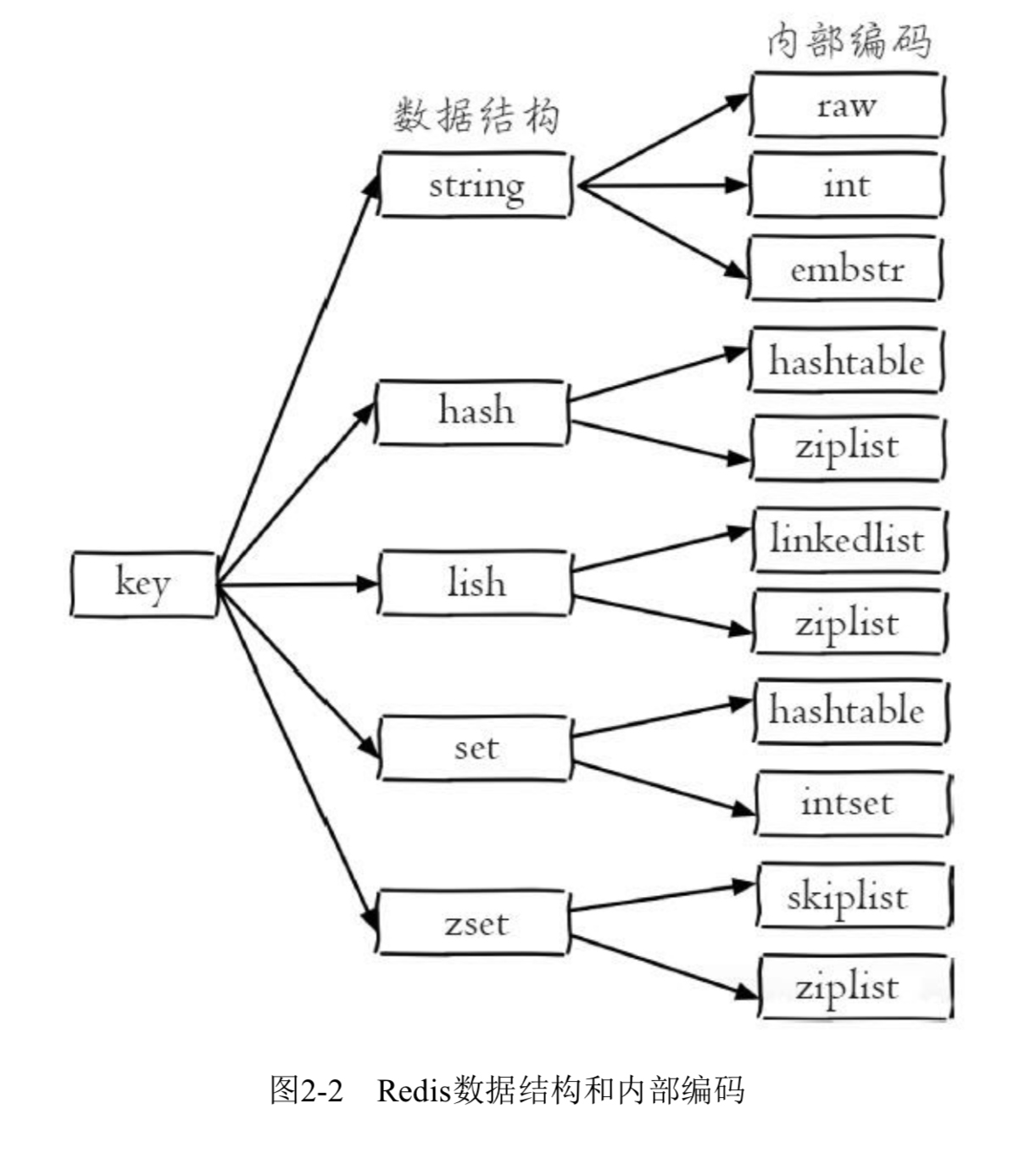
1. 键的数据结构类型：type keys

### 2.1.2 数据结构和内部编码

可以使用object encoding key 查看内部编码







### 2.1.3 单线程架构

Redis使用单线程架构和IO多路复用模型来实现高新能的内存数据库服务。

多个客户端的多条命令到Redis服务，命令会被放到一个对列中，然后单线程执行。

为什么单线程还能这么块？

1. 纯内存访问，Redis将所有数据放在内存中，内存响应时间约韦100纳秒。
2. 非阻塞IO，Redis使用epoll作为IO多路复用技术的实现。
3. 单线程避免了线程的切花和静态产生的消耗

单线程的问题？

对于每个命令的执行时间是由要求的。如果某个命令执行过长，会造成其他命令的阻塞。

## 2.2 字符串

## 2.3 哈希

## 2.4 列表

## 2.5 集合

## 2.6 有序集合

## 2.7 键管理