# Redis

Keys \* 查询所有key

Select 5选择5库

Exists key是否存在

Ttl key 还剩多少秒过期

Expire key seconds 设置过期时间

## 配置

## 常用数据类型

### String

### Hash

### List

### Set

### Sorted Set

## 持久化

### Rdb

指定时间间隔把内存数据快照写入磁盘，snapshot快照，恢复时候直接将快照读入内存。会通过一个子线程

通过一次子线程来持久化写入临时文件，之后替换掉上次持久化的文件。

保存在dump.rdb文件

服务器启动时候自动加载该文件恢复数据

#### 如何触发rdb

默认配置文件快照

命令save（阻塞）和bgsave（后台异步）

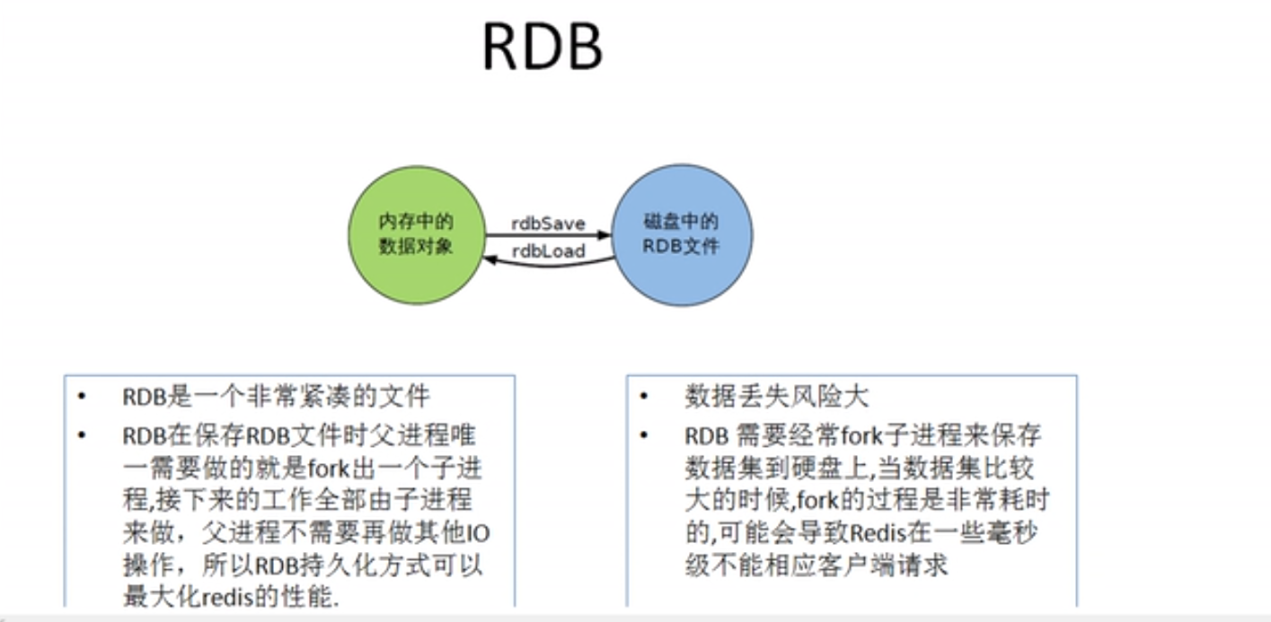
#### 优缺点

优点：

1. 适合大规模数据恢复
2. 对数据完整性要求不高

缺点：

1. 可能会丢失最后一次修改的记录
2. 内存要抗住fork时候的数据



### Aof（Append only file）

以日志形式记录写操作，只需追加文件

文件名appendonly.aof

当该文件出现问题时候，可以使用redis-check-aof文件来恢复

当同时存在aof和rdb文件时候，启动优先采用aof文件

#### 重写

超过文件大小阀值，进行内容压缩会保留恢复数据的最小指令集，可以使用命令bgrewriteaof

原理：会fork一个新进程来重新写入，不采用原来旧的aof文件，替代旧的

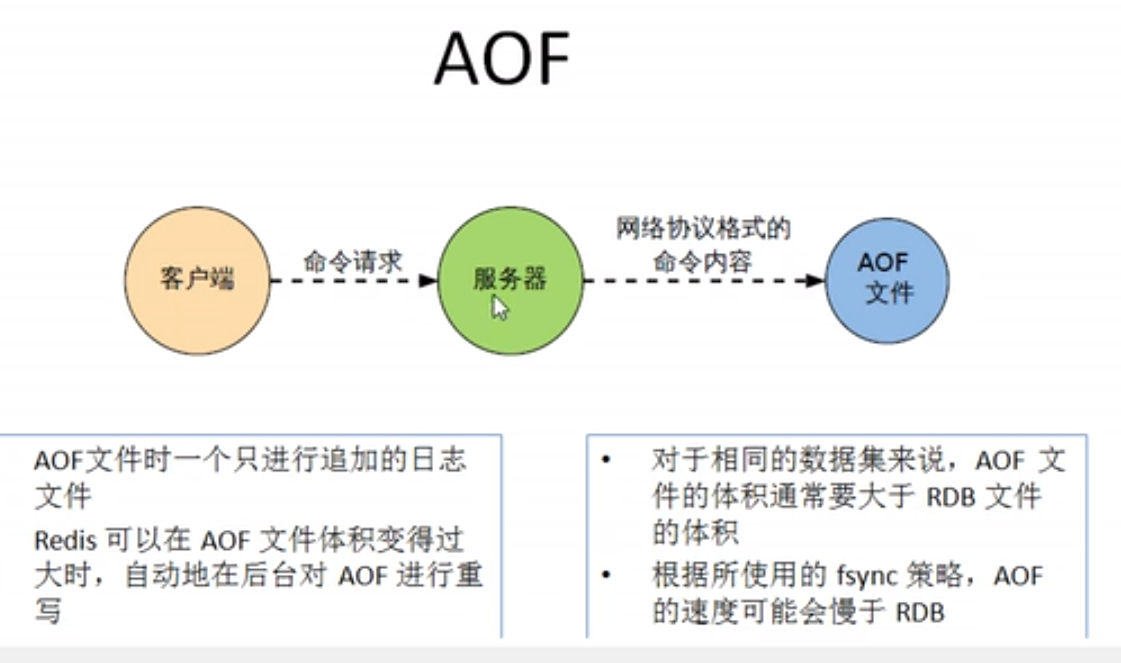
auto-aof-rewrite-percentage 100 默认重写触发要大于原来的一倍

auto-aof-rewrite-min-size 64mb 默认重写触发要大于64M

#### 优缺点

优点：完整性高

缺点：文件大，数据恢复速率慢



## 主从复制

主库写，从库只能读

Info replication来显示当前服务器的主从状态

Slaveof host port从属于某个服务器

### 一主二从

1、主机断了，从机不会上位，依靠心跳检测等待主机归位。

2、但是从机与主机断开后，需要重新连接，否则不会再是从机，除非在配置文件里配置

### 薪火相传

1、去中心化

2、上一个从机是下一个从机的master，减轻master压力

### 反客为主

主机挂了，从机自己当主机

Slaveof no one

### 复制原理

Slave第一次连接master全量复制数据，后面就是增量复制数据

## 面试

Redis数据结构使用场景

String 封锁ip地址，记录ip地址记录

Hash 存储用户信息（id，name，age） put（userId，id） put（userName，name）

List 实现消息的排行，可以将任务存在list中，实现FIFO或者LIFO，模拟消息队列

Set 可以自动去重，好友不允许重复，求共同好友，只需要求交集

Zset 以某一个条件为权重来排名