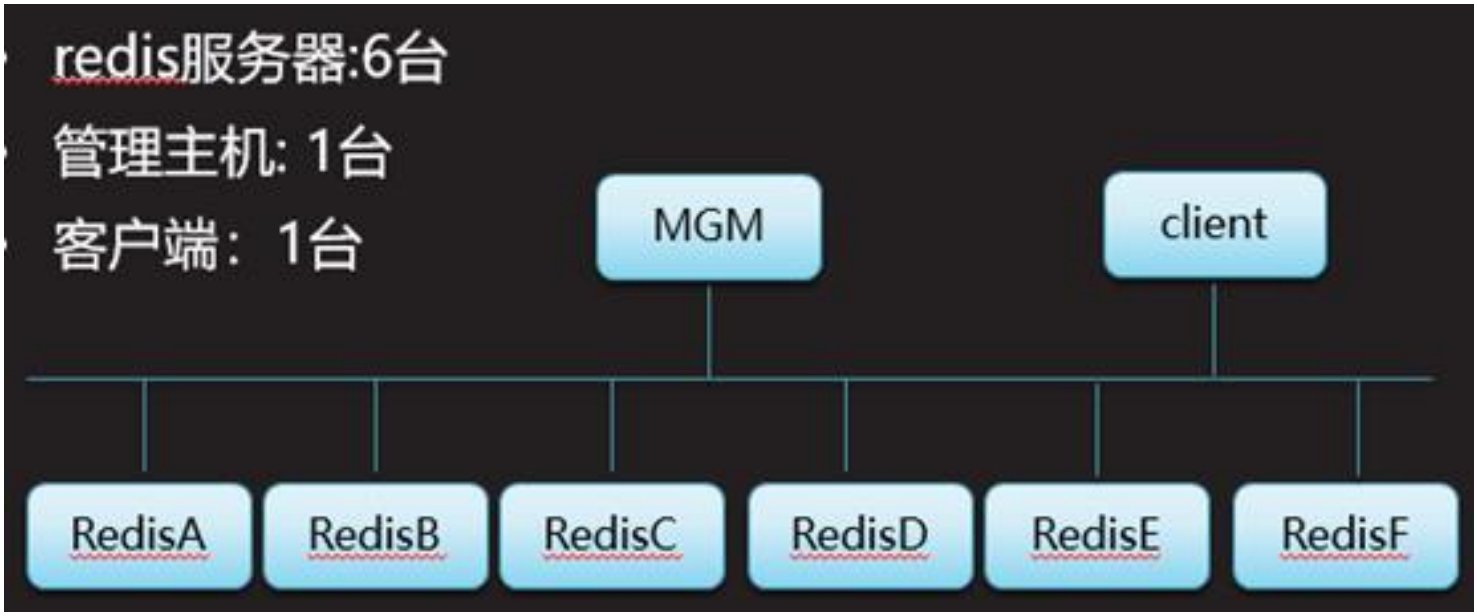
11_nosql02 创建集群+管理集群[高可用集群]

一 创建集群-集群环境

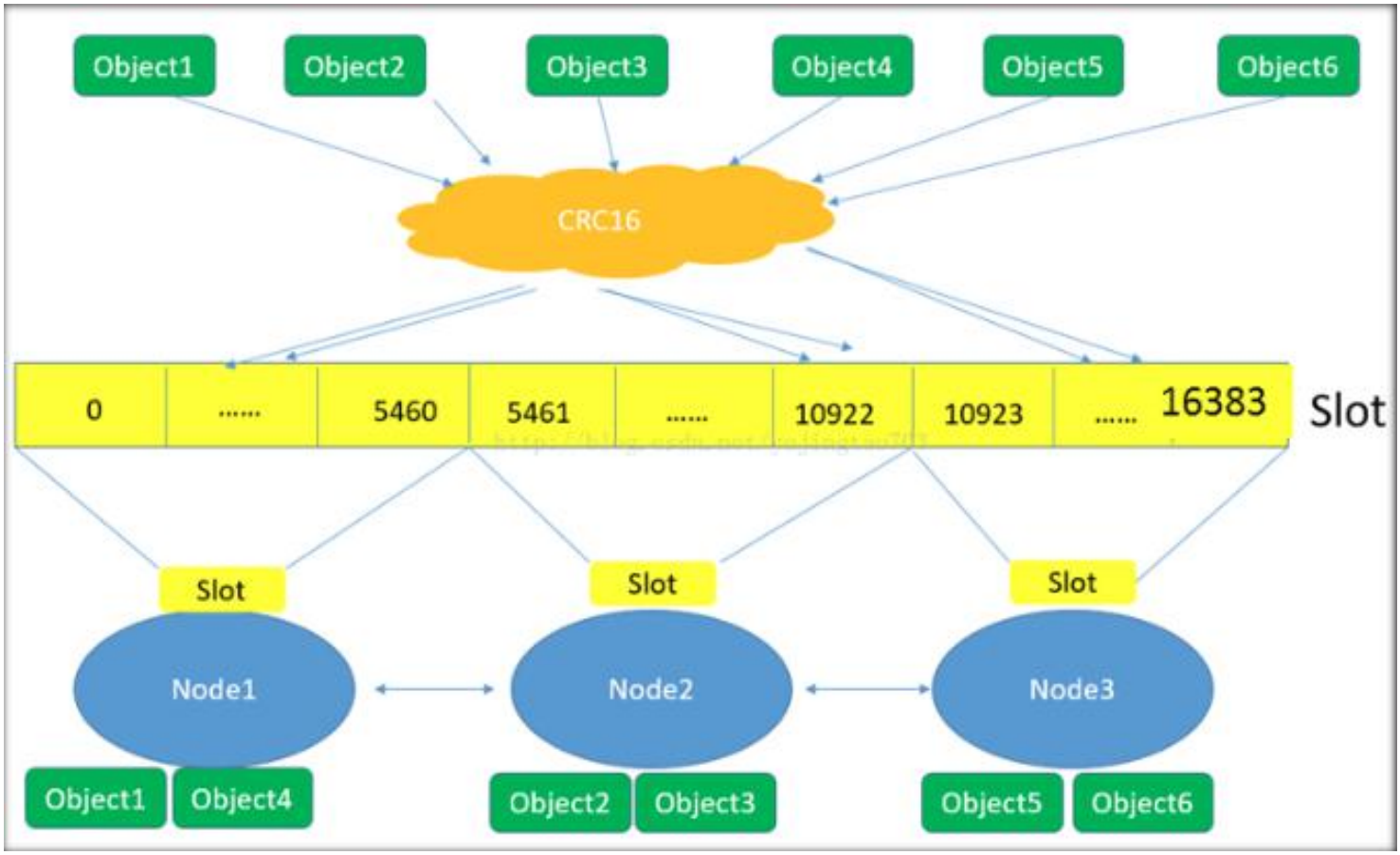
1.1 拓扑结构



1.2 IP 规划

主机名	IP 地址	端口号
client50	192.168.4.50	无
redisA	192.168.4.51	6351
redisB	192.168.4.52	6352
redisC	192.168.4.53	6353
redisD	192.168.4.54	6354
redisE	192.168.4.55	6355
redisF	192.168.4.56	6356
Mgm57	192.168.4.57	无

1.3 工作原理



二 创建集群

2.1 部署管理主机

管理主机可部署在任意一台 redis 服务器上,但最好单独部署

2.1.1 部署 ruby 脚本运行环境

```
redis57 ~]#yum -y install ruby rubygems   #用于安装 gem 软件包的软件
redis57 ~]# which gem
/usr/bin/gem
room9pc01 ~]$ scp /linux-soft/03/redis/redis-3.2.1.gem
root@192.168.4.57:/root    #传输 gem 安装包
redis57 ~]# gem install redis-3.2.1.gem    #安装 gem
```

2.1.2 创建管理集群脚本

```
redis57 ~]#mkdir  /root/bin     #创建命令检索目录
room9pc01 ~]$ scp /linux-soft/03/redis/redis-4.0.8.tar.gz
root@192.168.4.57:/root
redis57 ~]#tar -zxvf redis-4.0.8.tar.gz
redis57 ~]#cd redis-4.0.8/src/
redis57 ~]#cp redis-trib.rb /root/bin/ #创建管理集群 ruby 脚本
redis57 ~]#chmod +x /root/bin/redis-trib.rb
redis57 ~]#redis-trib.rb help  #查看命令帮助
```

2.2 redis-trib.rb 脚本

用法]# redis-trib.rb <command> <options> <arguments...>

常用命令	描述
create	创建集群
check	检查集群
info	查看集群信息
reshard	重新分片
del-node	删除主机
add-node --slave	添加 **slave** 主机
add-node	添加 **master** 主机
rebalance	平均分配 **hash slots**

2.3 创建集群

2.3.1 启动服务器 **192.168.4.51\52\53\54\55\56** 的集群功能

redis51 ~]# /etc/init.d/redis_6379 stop #停止 redis 服务

redis51 ~]# vim **/etc/redis/6379.conf** #修改配置文件

70 bind 192.168.4.51 #修改 ip

93 port 6351 #修改端口（可选配置）

815 cluster-enabled yes #解除注释,启用集群功能

823 cluster-config-file nodes-6379.conf #存储集群信息的配置文件

829 cluster-node-timeout 5000 #集群节点通信超时时间

redis51 ~]# **rm -rf /var/lib/redis/6379/* #清空数据**

redis51 ~]# **vim +43 /etc/init.d/redis_6379**

$CLIEXEC -h 192.168.4.51 -p 6351 shutdown #修改端口

redis51 ~]# **/etc/init.d/redis_6379 start**

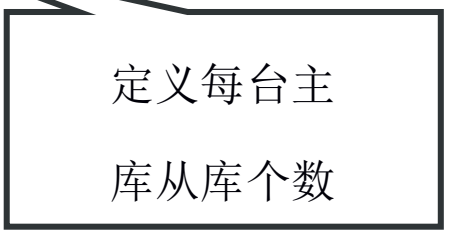
redis51 ~]# **netstat -utnlp | grep redis-server #多个 16351 端口**

tcp 0 0 192.168.4.51:6351 0.0.0.0:* LISTEN 21201/redis-server

tcp 0 0 192.168.4.51:16351 0.0.0.0:* LISTEN 21201/redis-server

2.3.2 57 上创建集群

```
redis57 ~]# redis-trib.rb create --replicas 1 \
> 192.168.4.51:6351 192.168.4.52:6352 \
> 192.168.4.53:6353 192.168.4.54:6354 \
> 192.168.4.55:6355 192.168.4.56:6356
Can I set the above configuration? (type 'yes' to accept): yes
    #提示信息出现时输入 yes
 [OK] All 16384 slots covered.    #表示创建成功
```

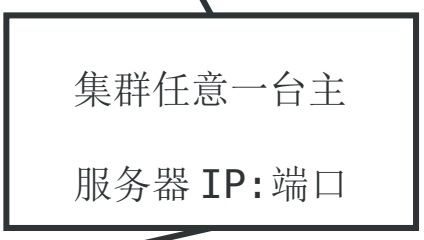


排错: 按 2.3.1 步骤进行:停服务,清空数据,检查配置文件,起服务

2.4 查看集群信息

2.4.1 在管理主机查看集群信息

```
redis57 ~]# redis-trib.rb info 192.168.4.51:6351 #查看集群信息
192.168.4.51:6351 (e1070da9...) -> 0 keys | 5461 slots | 1 slaves.
192.168.4.52:6352 (e6e67595...) -> 0 keys | 5462 slots | 1 slaves.
192.168.4.53:6353 (efb31698...) -> 0 keys | 5461 slots | 1 slaves.
[OK] 0 keys in 3 masters.
0.00 keys per slot on average.
```



2.4.2 在管理主机检查集群

```
redis57 ~]# redis-trib.rb check 192.168.4.53:6353
```

2.4.3 在每台 **redis** 服务器本机,查看集群信息

```
redis51 ~]# redis-cli  -h 192.168.4.51 -p 6351  #登录 redis
192.168.4.51:6351> cluster info  #查看集群信息
cluster_state:ok
cluster_slots_assigned:16384
cluster_slots_ok:16384
cluster_slots_pfail:0
cluster_slots_fail:0
cluster_known_nodes:6
cluster_size:3
cluster_current_epoch:6
cluster_my_epoch:1
cluster_stats_messages_ping_sent:1970
cluster_stats_messages_pong_sent:1697
cluster_stats_messages_sent:3667
cluster_stats_messages_ping_received:1692
cluster_stats_messages_pong_received:1970
cluster_stats_messages_meet_received:5
cluster_stats_messages_received:3667
192.168.4.51:6351> cluster  nodes   #查看集群节点信息
9648...873f 192.168.4.54:6354@16354 slave
```

efb3...6958 0 1568603165414 4 connected

e107...b3f0 **192.168.4.51:6351@16351** myself,master -

0 1568603163000 1 connected 0-5460

1c1c...8a0b **192.168.4.55:6355@16355** slave

e107...b3f0 0 1568603165000 5 connected

e6e6...9c2c **192.168.4.52:6352@16352** master -

0 1568603165916 2 connected 5461-10922

e3b2...76c3 **192.168.4.56:6356@16356** slave

e6e6...9c2c 0 1568603164912 6 connected

efb3...6958 **192.168.4.53:6353@16353** master

- 0 1568603164511 3 connected 10923-16383

2.5 访问集群

redis-cli -c -h ip 地址 **-p** 端口号　　选项**-c** 表示集群模式

在客户端连接集群中的任意一台服务器存取数据

redis50 ~]# redis-cli -c -h 192.168.4.51 -p 6351 #连接服务器 51

192.168.4.51:6351> set x 100 #存储

-> Redirected to slot [16287] located at 192.168.4.53:6353

   #提示存储在 53 主机

OK

192.168.4.53:6353> keys *

```
1) "x"
192.168.4.53:6353>
192.168.4.53:6353> set y 200
OK
192.168.4.53:6353> keys *
1) "y"
2) "x"
192.168.4.53:6353> set z 300 #存储
-> Redirected to slot [8157] located at 192.168.4.52:6352 #提示
存储在 52 主机
OK
192.168.4.52:6352> keys *  #在 52 主机查看数据 只有变量 z
1) "z"
192.168.4.52:6352> get x
-> Redirected to slot [16287] located at 192.168.4.53:6353
#连接 53 主机获取数据
"100"
192.168.4.53:6353> keys *
1) "y"
2) "x"
```

```
192.168.4.53:6353> get z
-> Redirected to slot [8157] located at 192.168.4.52:6352
"300"
192.168.4.52:6352> set i 400
-> Redirected to slot [15759] located at 192.168.4.53:6353
OK
192.168.4.53:6353> set j 500
-> Redirected to slot [3564] located at 192.168.4.51:6351
OK
```

三 管理集群-测试集群功能

3.1 故障切换测试

停止 master 主机的 redis 服务

master 宕机后,对应的 slave 自动被选举为 master

原 master 启动后,会自动配置为当前 master 的 slave

3.2 检测集群

在管理主机查看信息

```
redis-trib.rb check   192.168.4.52:6352
redis-trib.rb info    192.168.4.52:6352
```

四 管理集群-添加服务器

4.1 添加 master 服务器

部署一台新 redis 服务器 58,运行服务并启用集群配置

4.1.1 添加 master 主机到集群

添加 master 主机,添加时不指定主机角色,默认新主机被选为 master

redis57 ~]# **redis-trib.rb add-node** 192.168.4.58:6358 192.168.4.53:6353 #执行添加命令

[OK] New node added correctly. #提示添加完成

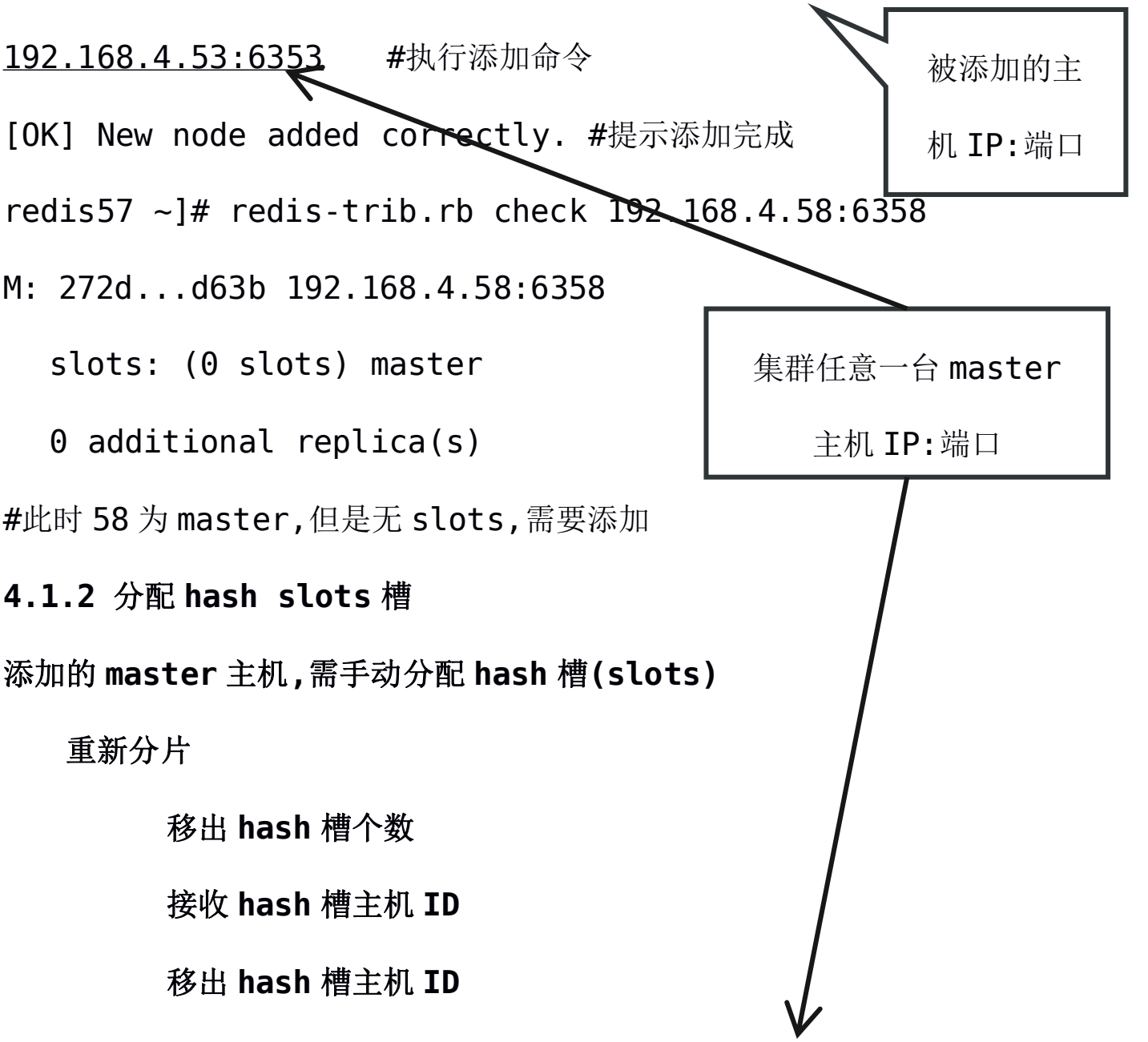
redis57 ~]# redis-trib.rb check 192.168.4.58:6358

M: 272d...d63b 192.168.4.58:6358

slots: (0 slots) master

0 additional replica(s)

#此时 58 为 master,但是无 slots,需要添加



4.1.2 分配 hash slots 槽

添加的 master 主机,需手动分配 hash 槽(slots)

重新分片

移出 **hash** 槽个数

接收 **hash** 槽主机 **ID**

移出 **hash** 槽主机 **ID**

redis57 ~]# **redis-trib.rb reshard** 192.168.4.58:6358

How many slots do you want to move (from 1 to 16384)? 4096

What is the receiving node ID?

272d37013046c6a72bfd8082483d700ef1f2d63b　　#58 的 ID

Source node #1:all　　#从所有 master 主机中移出 hash 槽

Do you want to proceed with the proposed reshard plan (yes/no)?

yes　　#是否继续执行提议的分片计划,输入 yes

4.1.3 查看集群信息

redis57 ~]# redis-trib.rb check 192.168.4.58:6358

M: 272d...d63b 192.168.4.58:6358

　slots:0-1364,5461-6826,10923-12287 (4096 slots) master

#此时 58 为 master,已经有了手动重新分片后的 4096 个 hash slots 槽

4.1.4 访问集群存取数据

client ~]# redis-cli -c -h 192.168.4.58 -p 6358

#50 上登录 58 的 redis

192.168.4.58:6358> keys *　#列出已有数据

192.168.4.58:6358> set v10 101　　#写入数据

192.168.4.58:6358> set v20 102

不指定主服务器的 ID,默认把新节点添加为从节点最少的主服务器

4.2 添加 slave 服务器

4.2.1 部署一台新 redis 服务器 59,运行服务并启用集群配置

redis-trib.rb add-node --slave [--master-id id 值]

从服务器 ip:端口　集群中任意一台主服务器 ip 地址:端口

4.2.2 将 slave 主机 59 添加到集群中

```
redis57 ~]# redis-trib.rb add-node --slave 192.168.4.59:6359
192.168.4.51:6351    #未指定从服务器 59 的主服务器
[OK] New node added correctly.    #提示添加成功
```

4.2.3 检查集群状态

```
redis57 ~]# redis-trib.rb check 192.168.4.51:6351
S: 18a2...b439 192.168.4.59:6359
   slots: (0 slots) slave
   replicates 272d....d63b    #272d....d63b 为主服务器 58 的 ID
```

4.2.4 登录 59 的 redis 读写数据

```
client ~]# redis-cli -c -h 192.168.4.59 -p 6359
192.168.4.59:6359> keys *  #和 58 上的数据一致
1) "shuaige"
2) "sex"
3) "y"
4) "c"
192.168.4.59:6359> set v100 100
-> Redirected to slot [9407] located at 192.168.4.52:6352
OK
```

五 管理集群-移除服务器

移除服务器要先移除 slave 服务器,再移除 master 服务器

5.1 移除 slave 服务器

5.1.1 从服务器没有 hash 槽,直接移除;移除时指定从服务器的 ID 值;集群会自动停止移除的从服务器的 redis 服务

redis-trib.rb del-node 集群任意一台主主机 IP:端口 要移除的从主机 ID

```
redis57 ~]# redis-trib.rb del-node 192.168.4.51:6351
18a2....b439    #移除从服务器 59
```

5.1.2 查看集群信息,显示主服务器 58 没有从服务器了

```
redis57 ~]# redis-trib.rb info 192.168.4.51:6351
192.168.4.58:6358 (272d3701...) -> 4 keys | 8192 slots | 0 slaves.
    #显示 58 的没有从服务器了
```

5.1.3 59 上查看 redis 服务,被集群自动停止

```
redish ~]# ss -antulp | grep redis-server  #无查询结果,被停止了
```

5.2 移除 master 服务器

5.2.1 释放占用的 hash 槽

5.2.1.1 未释放 hash 槽时移除 58,提示移除 hash 槽,重新分片

```
redis57 ~]# redis-trib.rb del-node 192.168.4.51:6351 272d...d63b
[ERR] Node 192.168.4.58:6358 is not empty! Reshard data away and
try again.  #提示移除 hash 槽,重新分片
```

5.2.1.2 重新分片

redis-trib.rb reshard 集群任意一台主机 IP:端口

指定移出 slots 的个数

指定接收 slots 的主服务器 ID

指定移出 slots 的主服务器 ID

```
redis57 ~]# redis-trib.rb reshard 192.168.4.51:6351
How many slots do you want to move (from 1 to 16384)? 8192
What is the receiving node ID? e107...b3f0  #51 接收 slots
Source node #1:272d...d63b    #从 58 移出
Source node #2:done    #输入 done
Do you want to proceed with the proposed reshard plan (yes/no)?
yes    #是否继续执行提议的分片计划,输入 yes
```

5.2.2 移除 master 服务器

redis-trib.rb del-node 集群任意一台主服务器 IP:端口 移除的主机 ID 值

5.2.2.1 移除主服务器 58

```
redis57 ~]# redis-trib.rb del-node 192.168.4.51:6351 272d...d63b
    #移除主服务器 58
>>> SHUTDOWN the node.    #提示移除成功
```

5.2.2.2 58 上查看 redis 服务状态,被集群自动停止

```
redis58 ~]# ss -antulp | grep redis-server  #无查询结果,被停止了
```

5.2.2.3 57 上查看集群信息,已无 58 主服务器

```
redis57 ~]# redis-trib.rb info 192.168.4.51:6351
192.168.4.51:6351 (e1070da9...) -> 5 keys | 10923 slots | 1 slaves.
192.168.4.52:6352 (e6e67595...) -> 2 keys | 2731 slots | 1 slaves.
192.168.4.53:6353 (efb31698...) -> 1 keys | 2730 slots | 1 slaves.
```

5.3 把移除的服务器再添加到集群里

5.3.1 移除的服务器内不能有数据,删除数据库目录内的数据

```
rm -rf /var/lib/redis/6379/*
```

5.3.2 启动 **redis** 服务**(**移除时 **redis** 服务被集群自动停止了**)**

```
redis58 ~]# /etc/init.d/redis_6379 start
```

5.3.3 移除的服务器登录本机 **redis,**执行 **cluster reset**

```
redis58 ~]# redis-cli -h 192.168.4.58 -p 6358
192.168.4.58:6358> cluster reset
OK
```

5.3.4 集群添加被移除的服务器**:** 按 四 的步骤进行

5.4 让一台主服务器有 **2** 个从服务器

5.4.1 清空 **58 59** 数据库目录内数据

```
rm -rf /var/lib/redis/6379/*
```

5.4.2 58 59 启动 **redis** 服务

```
/etc/init.d/redis_6379 start
```

5.4.3 58 59 登录本机 **redis,**执行 **cluster reset**

```
redis-cli -h 192.168.4.59 -p 6359
192.168.4.59:6359> cluster reset
OK
```

5.4.4 集群添加从服务器 **58 59**

redis-trib.rb add-node --slave [--master-id id 值**]**

从服务器 **ip:**端口　集群中任意一台服务器 **ip** 地址**:**端口

5.4.4.1 添加 **58** 为 **51** 的从服务器

```
redis57 ~]# redis-trib.rb add-node --slave --master-id
e107...b3f0 192.168.4.58:6358 192.168.4.51:6351
```

5.4.4.2 添加 **59** 为 **52** 的从服务器

```
redis57 ~]# redis-trib.rb add-node --slave --master-id
e6e6...9c2c 192.168.4.59:6359 192.168.4.51:6351
```

5.4.4.3 查看集群信息**,51 52** 各有两个从服务器

```
redis57 ~]# redis-trib.rb info 192.168.4.51:6351
192.168.4.51:6351 (e1070da9...) -> 5 keys | 10923 slots | 2 slaves.
192.168.4.52:6352 (e6e67595...) -> 2 keys | 2731 slots | 2 slaves.
192.168.4.53:6353 (efb31698...) -> 1 keys | 2730 slots | 1 slaves.
```

5.4.4.4 停止 **51** 的 **redis** 服务

```
redis51 ~]# /etc/init.d/redis_6379 stop
```

5.4.4.5 57 上连接 **51,**提示无法连接

```
redis57 ~]# redis-cli -c -h 192.168.4.51 -p 6351
Could not connect to Redis at 192.168.4.51:6351: Connection
refused
```

5.4.4.6 57 上检查集群

```
redis57 ~]# redis-trib.rb check 192.168.4.52:6352
```

#51 消失,55 变成了主服务器,58 为 55 的从服务器

5.4.4.7 50 连接 **52** 写入数据

```
client ~]# redis-cli -c -h 192.168.4.52 -p 6352
192.168.4.52:6352> set xy 99
-> Redirected to slot [11854] located at 192.168.4.55:6355
OK
```

5.4.4.8 启动 **51** 的 **redis** 服务

```
redis51 ~]# /etc/init.d/redis_6379 start
```

5.4.4.9 57 检查集群状态,提示 **51** 为 **55** 的从服务器

```
redis57 ~]# redis-trib.rb check 192.168.4.52:6352
S: e107...b3f0 192.168.4.51:6351
   slots: (0 slots) slave
   replicates 1c1c...8a0b
```

5.4.4.10 50 连接 **51** 的 **redis** 服务,查看 **51** 上的数据,与 **55** 上的数据一致

```
client ~]# redis-cli -c -h 192.168.4.51 -p 6351
```

192.168.4.51:6351> keys *

5.5 查看本机复制状态

info replication

5.6 当一组主从服务器全部 down 后,整个集群 down

六 集群服务器还原

6.1 停止 redis 服务 /etc/init.d/reids_6379 stop

6.2 删除数据库目录下数据 /var/lib/redis/6379/*

6.3 配置文件注释集群语句 /etc/redis/6379.conf

6.4 启动 redis 服务,查看端口,此时应无 10000+端口

6.5 登录 redis,查看集群信息,cluster info