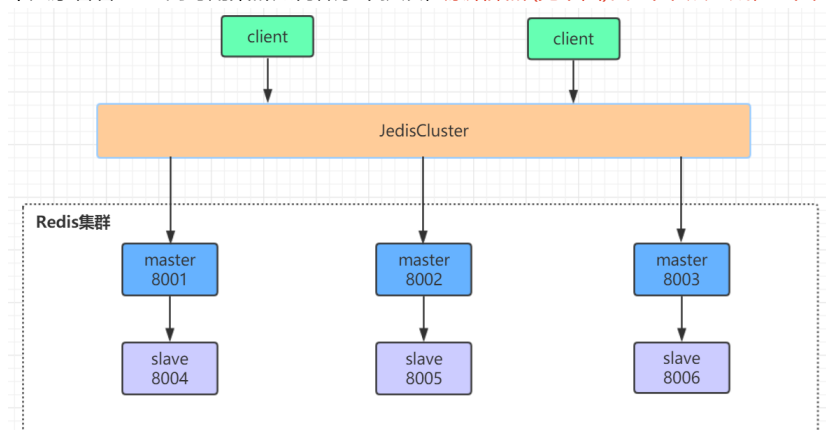Redis3.0以后的版本虽然有了集群功能，提供了比之前版本的哨兵模式更高的性能与可用性，但是集群的水平扩展却比较麻烦，今天就来带大家看看redis高可用集群如何做水平扩展，原始集群(见下图)由6个节点组成，6个节点分布在三台机器上，采用三主三从的模式

1、启动集群

启动整个集群

```
1  /usr/local/redis-5.0.3/src/redis-server /usr/local/redis-cluster/8001/redis.conf
2  /usr/local/redis-5.0.3/src/redis-server /usr/local/redis-cluster/8002/redis.conf
3  /usr/local/redis-5.0.3/src/redis-server /usr/local/redis-cluster/8003/redis.conf
4  /usr/local/redis-5.0.3/src/redis-server /usr/local/redis-cluster/8004/redis.conf
5  /usr/local/redis-5.0.3/src/redis-server /usr/local/redis-cluster/8005/redis.conf
6  /usr/local/redis-5.0.3/src/redis-server /usr/local/redis-cluster/8006/redis.conf
```

客户端连接8001端口的redis实例

```
1  /usr/local/redis-5.0.3/src/redis-cli -a zhuge -c -h 192.168.0.61 -p 8001
```
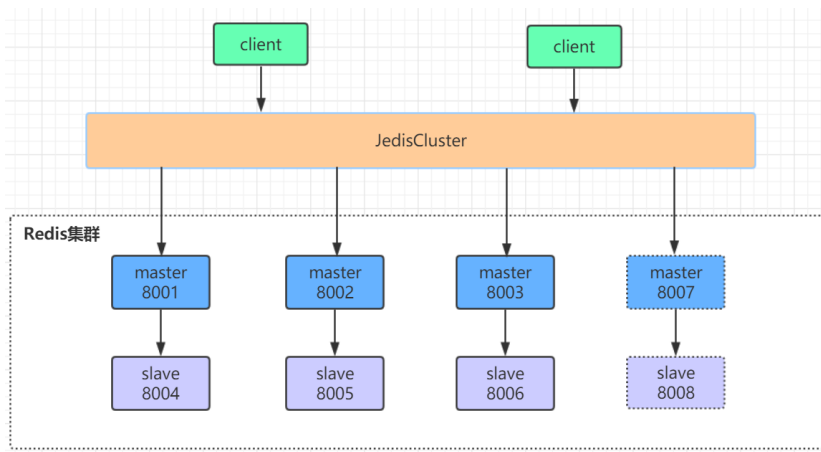
查看集群状态

```
1  192.168.0.61:8001> cluster  nodes
```


从上图可以看出，整个集群运行正常，三个master节点和三个slave节点，8001端口的实例节点存储0-5460这些hash槽，8002端口的实例节点存储5461-10922这些hash槽，8003端口的实例节点存储10923-16383这些hash槽，这三个master节点存储的所有hash槽组成redis集群的存储槽位，slave点是每个主节点的备份从节点，不显示存储槽位

2、集群操作

我们在原始集群基础上再增加一主(8007)一从(8008)，增加节点后的集群参见下图，新增节点用虚线框表示

- **增加redis实例**

在/usr/local/redis-cluster下创建8007和8008文件夹，并拷贝8001文件夹下的redis.conf文件到8007和8008这两个文件夹下

```
1  mkdir 8007 8008
2  cd 8001
3  cp redis.conf /usr/local/redis-cluster/8007/
4  cp redis.conf /usr/local/redis-cluster/8008/
5
6  # 修改8007文件夹下的redis.conf配置文件
7  vim /usr/local/redis-cluster/8007/redis.conf
8  # 修改如下内容:
9  port:8007
10 dir /usr/local/redis-cluster/8007/
11 cluster-config-file nodes-8007.conf
12
13 # 修改8008文件夹下的redis.conf配置文件
14 vim /usr/local/redis-cluster/8008/redis.conf
15 修改内容如下:
16 port:8008
17 dir /usr/local/redis-cluster/8008/
18 cluster-config-file nodes-8008.conf
19
20 # 启动8007和8008俩个服务并查看服务状态
21 /usr/local/redis-5.0.3/src/redis-server /usr/local/redis-cluster/8007/redis.conf
22 /usr/local/redis-5.0.3/src/redis-server /usr/local/redis-cluster/8008/redis.conf
23 ps -el | grep redis
```

- **查看redis集群的命令帮助**

```
1  cd /usr/local/redis-5.0.3
2  src/redis-cli --cluster help
```

```
[root@centos-new redis-5.0.2]# src/redis-cli --cluster help
Cluster Manager Commands:
  create           host1:port1 ... hostN:portN
                   --cluster-replicas <arg>
  check            host:port
  info             host:port
  fix              host:port
  reshard          host:port
                   --cluster-from <arg>
                   --cluster-to <arg>
                   --cluster-slots <arg>
                   --cluster-yes
                   --cluster-timeout <arg>
                   --cluster-pipeline <arg>
  rebalance        host:port
                   --cluster-weight <node1=w1...nodeN=wN>
                   --cluster-use-empty-masters
                   --cluster-timeout <arg>
                   --cluster-simulate
                   --cluster-pipeline <arg>
                   --cluster-threshold <arg>
  add-node         new_host:new_port existing_host:existing_port
                   --cluster-slave
                   --cluster-master-id <arg>
  del-node         host:port node_id
  call             host:port command arg arg .. arg
```

1.create：创建一个集群环境host1:port1 ... hostN:portN

2.call：可以执行redis命令

3.add-node：将一个节点添加到集群里，第一个参数为新节点的ip:port，第二个参数为集群中任意一个已经存在的节点的ip:port

4.del-node：移除一个节点

5.reshard：重新分片

6.check：检查集群状态

- **配置8007为集群主节点**

使用add-node命令新增一个主节点8007(master)，前面的ip:port为新增节点，后面的ip:port为已知存在节点，看到日志最后有"[OK]
New node added correctly"提示代表新节点加入成功

```
1  /usr/local/redis-5.0.3/src/redis-cli -a zhuge --cluster add-node 192.168.0.61:8007 192.168.0.61:8001
```

查看集群状态

```
1  /usr/local/redis-5.0.3/src/redis-cli -a zhuge -c -h 192.168.0.61 -p 8001
2  192.168.0.61:8001> cluster nodes
```

```
192.168.0.61:8001> cluster nodes
57bee15ef8d796eaa36f3f1a178ddaf574f378a6 192.168.0.63:8003@18003 master - 0 1571378417587 3 connected 10923-16383
3e689f247a438bb0b286dfb790d0dc8a2a193d84 192.168.0.61:8004@18004 slave 57bee15ef8d796eaa36f3f1a178ddaf574f378a6 0 1571378418523 4 connected
dfca1388f124dec92f394a7cc85cf98cfa02f86f 192.168.0.61:8001@18001 myself,master - 0 1571378418000 1 connected 0-5460
2728a594a0498e98e4b83a537e19f9a0a3790f38 192.168.0.61:8007@18007 master - 0 1571378418918 0 connected
bff9cb673443a9c894233eb60bcba95359c1efa9 192.168.0.62:8002@18002 master - 0 1571378418534 8 connected 5461-10922
ce25e853a3cee4db9966367c25a93a06abe5e70e 192.168.0.62:8005@18005 slave dfca1388f124dec92f394a7cc85cf98cfa02f86f 0 1571378418000 5 connected
c7504f90c52ec14cecf10124a103f996c9b6fbbb 192.168.0.63:8006@18006 slave bff9cb673443a9c894233eb60bcba95359c1efa9 0 1571378417078 8 connected
```

注意：当添加节点成功以后，新增的节点不会有任何数据，因为它还没有分配任何的slot(hash槽)，我们需要为新节点手工分配hash槽

使用redis-cli命令为8007分配hash槽，找到集群中的任意一个主节点，对其进行重新分片工作。

```
1  /usr/local/redis-5.0.3/src/redis-cli -a zhuge --cluster reshard 192.168.0.61:8001
```

输出如下：

… …

How many slots do you want to move (from 1 to 16384)? 600

(ps:需要多少个槽移动到新的节点上，自己设置，比如600个hash槽)

What is the receiving node ID? 2728a594a0498e98e4b83a537e19f9a0a3790f38

(ps:把这600个hash槽移动到哪个节点上去，需要指定节点id)

Please enter all the source node IDs.

  Type 'all' to use all the nodes as source nodes for the hash slots.

  Type 'done' once you entered all the source nodes IDs.

Source node 1:all

(ps:输入all为从所有主节点(8001,8002,8003)中分别抽取相应的槽数指定到新节点中，抽取的总槽数为600个)

… …

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

(ps:输入yes确认开始执行分片任务)

… …

查看下最新的集群状态

```
1  /usr/local/redis-5.0.3/src/redis-cli -a zhuge -c -h 192.168.0.61 -p 8001
2  192.168.0.61:8001> cluster nodes
```


如上图所示，现在我们的8007已经有hash槽了，也就是说可以在8007上进行读写数据啦！到此为止我们的8007已经加入到集群中，并且是主节点(Master)

- **配置8008为8007的从节点**

添加从节点8008到集群中去并查看集群状态

```
1  /usr/local/redis-5.0.3/src/redis-cli -a zhuge --cluster add-node 192.168.0.61:8008 192.168.0.61:8001
```


如图所示，还是一个master节点，没有被分配任何的hash槽。

我们需要执行replicate命令来指定当前节点(从节点)的主节点id为哪个,首先需要连接新加的8008节点的客户端，然后使用集群命令进行操作，把当前的8008(slave)节点指定到一个主节点下(这里使用之前创建的8007主节点)

```
1  /usr/local/redis-5.0.3/src/redis-cli -a zhuge -c -h 192.168.0.61 -p 8008
2  192.168.0.61:8008> cluster replicate 2728a594a0498e98e4b83a537e19f9a0a3790f38  #后面这串id为8007的节点id
```

查看集群状态，8008节点已成功添加为8007节点的从节点

- **删除8008从节点**

用del-node删除从节点8008，指定删除节点ip和端口，以及节点id(红色为8008节点id)

```
1  /usr/local/redis-5.0.3/src/redis-cli -a zhuge --cluster del-node 192.168.0.61:8008 a1cfe35722d151cf70585cee212755653
   93c0956
```

再次查看集群状态，如下图所示，8008这个slave节点已经移除，并且该节点的redis服务也已被停止

```
192.168.0.61:8007> cluster nodes
ce25e853a3cee4db9966367c25a93a06abe5e70e 192.168.0.62:8005@18005 slave dfca1388f124dec92f394a7cc85cf98cfa02f86f 0 1571380561747 1 connected
57bee15ef8d796eaa36f3f1a178ddaf574f378a6 192.168.0.63:8003@18003 master - 0 1571380561000 3 connected 11122-16383
2728a594a0498e98e4b83a537e19f9a0a3790f38 192.168.0.61:8007@18007 myself,master - 0 1571380561000 9 connected 0-198 5461-5661 10923-11121
c7504f90c52ec14cecf10124a103f996c9b6fbbb 192.168.0.63:8006@18006 slave bff9cb673443a9c894233eb60bcba95359c1efa9 0 1571380562000 8 connected
bff9cb673443a9c894233eb60bcba95359c1efa9 192.168.0.62:8002@18002 master - 0 1571380561000 8 connected 5662-10922
3e689f247a438bb0b286dfb790d0dc8a2a193d84 192.168.0.61:8004@18004 slave 57bee15ef8d796eaa36f3f1a178ddaf574f378a6 0 1571380560520 3 connected
dfca1388f124dec92f394a7cc85cf98cfa02f86f 192.168.0.61:8001@18001 master - 0 1571380562000 1 connected 199-5460
```

- **删除8007主节点**

最后，我们尝试删除之前加入的主节点8007，这个步骤相对比较麻烦一些，因为主节点的里面是有分配了hash槽的，所以我们这里必须先把8007里的hash槽放入到其他的可用主节点中去，然后再进行移除节点操作，不然会出现数据丢失问题(目前只能把master的数据迁移到一个节点上，暂时做不了平均分配功能)，执行命令如下：

```
1  /usr/local/redis-5.0.3/src/redis-cli -a zhuge --cluster reshard 192.168.0.61:8007
```

输出如下：

 … …

How many slots do you want to move (from 1 to 16384)? 600

What is the receiving node ID? dfca1388f124dec92f394a7cc85cf98cfa02f86f

(ps:这里是需要把数据移动到哪？8001的主节点id)

Please enter all the source node IDs.

  Type 'all' to use all the nodes as source nodes for the hash slots.

  Type 'done' once you entered all the source nodes IDs.

Source node 1:2728a594a0498e98e4b83a537e19f9a0a3790f38

(ps:这里是需要数据源，也就是我们的8007节点id)

Source node 2:done

(ps:这里直接输入done 开始生成迁移计划)

 … …

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

(ps:这里输入yes开始迁移)

至此，我们已经成功的把8007主节点的数据迁移到8001上去了，我们可以看一下现在的集群状态如下图，你会发现8007下面已经没有任何hash槽了，证明迁移成功！

```
192.168.0.61:8007> cluster nodes
ce25e853a3cee4db9966367c25a93a06abe5e70e 192.168.0.62:8005@18005 slave dfca1388f124dec92f394a7cc85cf98cfa02f86f 0 1571380844545 10 co
57bee15ef8d796eaa36f3f1a178ddaf574f378a6 192.168.0.63:8003@18003 master - 0 1571380845578 3 connected 11122-16383
2728a594a0498e98e4b83a537e19f9a0a3790f38 192.168.0.61:8007@18007 myself,master - 0 1571380844000 9 connected
c7504f90c52ec14cecf10124a103f996c9b6fbbb 192.168.0.63:8006@18006 slave bff9cb673443a9c894233eb60bcba95359c1efa9 0 1571380846090 8 com
bff9cb673443a9c894233eb60bcba95359c1efa9 192.168.0.62:8002@18002 master - 0 1571380845000 8 connected 5662-10922
3e689f247a438bb0b286dfb790d0dc8a2a193d84 192.168.0.61:8004@18004 slave 57bee15ef8d796eaa36f3f1a178ddaf574f378a6 0 1571380846579 3 con
dfca1388f124dec92f394a7cc85cf98cfa02f86f 192.168.0.61:8001@18001 master - 0 1571380845666 10 connected 0-5661 10923-11121
192.168.0.61:8007>
```

最后我们直接使用del-node命令删除8007主节点即可

```
1  /usr/local/redis-5.0.3/src/redis-cli -a zhuge --cluster del-node 192.168.0.61:8007 2728a594a0498e98e4b83a537e19f9a0a3790f38
```

查看集群状态，一切还原为最初始状态啦！大功告成！

```
192.168.0.61:8001> cluster nodes
57bee15ef8d796eaa36f3f1a178ddaf574f378a6 192.168.0.63:8003@18003 master - 0 1571380924000 3 connected 11122-16383
3e689f247a438bb0b286dfb790d0dc8a2a193d84 192.168.0.61:8004@18004 slave 57bee15ef8d796eaa36f3f1a178ddaf574f378a6 0 1571380925755 4 c
dfca1388f124dec92f394a7cc85cf98cfa02f86f 192.168.0.61:8001@18001 myself,master - 0 1571380924000 10 connected 0-5661 10923-11121
bff9cb673443a9c894233eb60bcba95359c1efa9 192.168.0.62:8002@18002 master - 0 1571380925557 8 connected 5662-10922
ce25e853a3cee4db9966367c25a93a06abe5e70e 192.168.0.62:8005@18005 slave dfca1388f124dec92f394a7cc85cf98cfa02f86f 0 1571380925000 10 c
c7504f90c52ec14cecf10124a103f996c9b6fbbb 192.168.0.63:8006@18006 slave bff9cb673443a9c894233eb60bcba95359c1efa9 0 1571380924747 8 c
```

```
1  文档：03-VIP-Redis高可用集群之水平扩展
2  链接：http://note.youdao.com/noteshare?id=174eaa7104ce961f8012d521d8155bd9&sub=5CB163DE61434F4E92C5B62C22EC095E
```