**集群管理脚本**

\*\* 通过脚本实现的: **分布式存储高可用集群 \*\***

rudy语言编写的

可部署在任意redis服务器上

分布式存储: 数据存储在不同服务器上,而非存在一台服务器上

原理: 算法 获取存储数据的变量名,??????????

编号范围 0-16383 (16384)

**要求: 必须清空数据库  *---> 若忘记清空,则删除dump.rdb,再重启***

**主库最少3个**

须知: 16351端口(6351+10000): //集群通信端口

从服务器自动从主库备份数据

从库自动生成主库

创建集群

–必须清空数据库

–主库最少3个

#############################################################################

**部署集群脚本**

可部署在任意redis服务器上

本次:192.168.4.57

运行环境

• rudy(环境)

• gem(安装命令)

• redis-3.2.1.gem : 和redis通信的程序

]$ scp /linux-soft/03/redis/redis-3.2.1.gem [root@192.168.4.57:/root/](mailto:root@192.168.4.57:/root/)

1. 安装ruby & redis-3.2.1.gem

]# yum -y install ruby

]# gem install redis-3.2.1.gem // .gem 需gem命令

1. 移动管理脚本

源码包的src目录下

• redis-trib.rb (**.**rb: ruby的简写)

]# ls \***.**rb // 源码包的src目录下搜索以 **.**rb 结尾的文件

]# madir /root/bin

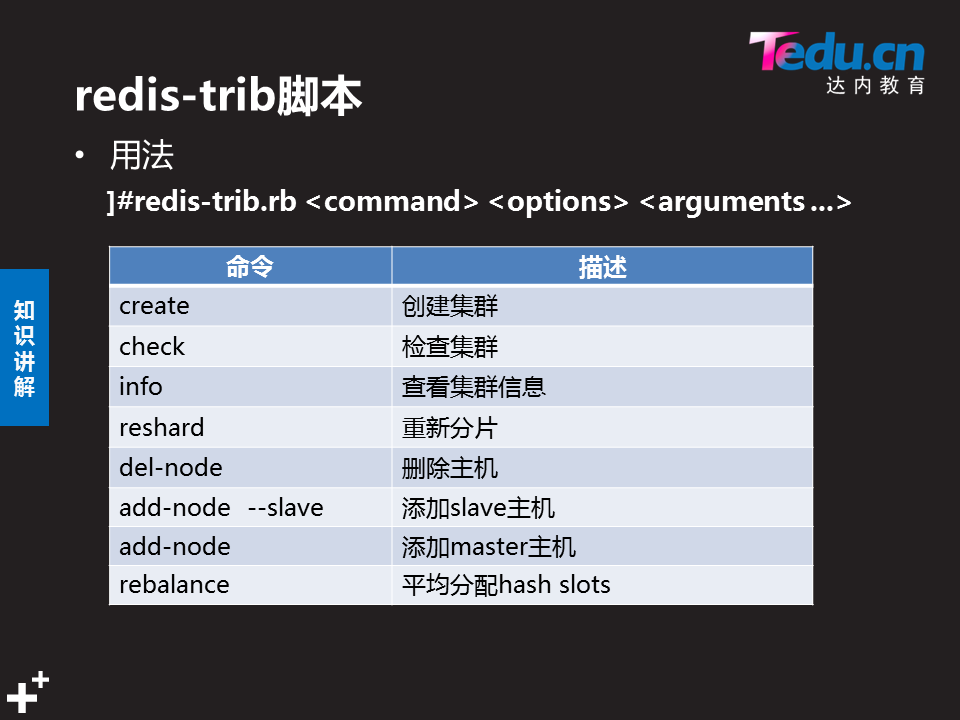
]# cp /root/redis-4.0.8/src/redis-trib.rb /roo/bin/

拷贝到root/bin/下, 就可以命令行敲脚本了(没有bin,需自己创建)

###########################################################################

**redis-trib.rb命令用法:**

**(help)**



########################################################################

**创建集群**

cluster info

1.redis开启集群功能

• 修改配置文件

– /etc/redis/6379.conf

815 cluster-enabled yes #开启集群功能

823 cluster-config-file nodes-6379.conf #集群信息文件(数据库目录下)

829 cluster-node-timeout 5000 #超时时间,认定为坏的(改为5000)

• 重启

– 多出16351端口(6351+10000): 集群通信端口

从服务器自动从主库备份数据

2.脚本创建集群

–必须清空数据库

–主库最少3个

]# redis-trib.rb create --replicas 1 \ #replicas 数字 : 每个主有一个从

**192.168.4.51:6351 \**

**192.168.4.52:6352 \**

**..........**

(随机分配master-slave)

[OK] All 16384 slots covered. //集群创建完成

1. 查看集群信息

• 在管理主机上查看

–redis-trib.rb info 服ip:端口 //查看集群信息(任意一个主服ip都代表集群)

–redis-trib.rb check 服ip:端口 //检查集群(详细,可查看各id)

[root@mgm57 ~]# redis-trib.rb info 192.168.4.51:6351

192.168.4.55:6355 (2d343a9d...) -> 3 keys | 4096 slots | 1 slaves.

192.168.4.53:6353 (9e44139c...) -> 3 keys | 4096 slots | 1 slaves.

192.168.4.52:6352 (324e05df...) -> 2 keys | 4096 slots | 1 slaves.

[root@mgm57 ~]# redis-trib.rb check 192.168.4.51:6351

>>> Performing Cluster Check (using node 192.168.4.51:6351)

S: --------> id <---------- Ip:端口

M: --------> id <---------- ip:端口

[OK] 13 keys in 4 masters.  **<------ //主服务器个数4台**

[OK] All 16384 slots covered.

• 在每一台,redis服务器本机查看

– > cluster info //查看集群信息

– > cluster nodes //查看集群节点信息

1. 访问集群

]# redis-cli **-c** -h任意一个机器的ip -p 端口号

-c: 集群模式 若不加-c,访问的为单个数据库,没有集群功能

Hash 槽

Id值

############################################################################

**管理集群**

从库自动生成主库

默认新添加的主机被选为master

若想为从 加 --slave

添加删除master都需要分配hash槽

添加命令

–redis-trib.rb add-node [--slave] 新IP 主IP:端口

重新分hash

移除命令

–redis-trib.rb add-node IP:端口 id //删除谁+删除的id

部署新的redis服务器

–装包，初始化，启用集群功能，重启服务

添加master服务器

–添加master主机

–分配hash槽(slots)

第一步

]# 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

第二步

]# redis-trib.rb reshard 192.168.4.53:6353

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

**//拿出4096个hash 槽给主机192.168.4.58**

What is the receiving node **ID? c5e0da48f335c46a2ec199faa99b830f537dd8a0**

**//主机192.168.4.58的id值**

Source node #1:**all //从当前所有master服务器获取hash槽**

Do you want to proceed with the proposed reshard plan (yes/no)?yes //同意以上配置

...

添加slave角色主机到集群里

]# redis-trib.rb add-node --slave 新IP 主IP:端口 //执行添加命令

>>> Configure node as replica of 192.168.4.58:6358. //提示添加完成

[OK] New node added correctly.

移除slave

]#redis-trib.rb del-node ip:端口 f6649ea99b2f01faca26...(id号)

移除master

先删除master占用的槽

]# redis-trib.rb reshard IP:端口

How many ...... move (from 1 to 16384)?4096 //移除4096个数槽

What is .... node ID? bc5c4e082a5a3391b634cf433a6486c867cfc44b

//要移动给谁的id即目标主机（这里可以随机写一个master的ID）

Source node #1: c5e0da48f335c46a2ec199faa99b830f537dd8a0

//从谁那移动即源主机（这里写4.58的ID）

Source node #2:done //设置完毕

再删master

]#删除命令