

hbase主从同步

一、replication 测试

打开复制开关



```
####Master-Slave模式 BEGIN ####
```

```
##HBase Master集群上执行
```

```
remove_peer '1'
```

```
add_peer '1', CLUSTER_KEY => "bigdata-6.baofoo.cn, bigdata-7.baofoo.cn, bigdata-8.baofoo.cn:2181:/hbase" #从集群的HBase在Zookeeper中的信息
```

所有对于replication的shell操作，只对列族为REPLICATION_SCOPE=>'1'才有效。

```
alter 'ns1:t1', {NAME => 'f1', REPLICATION_SCOPE => '1'}
```

测试：

```
put 'ns1:t1', 'c1', 'f1:ccc', '333'
```

```
put 'ns1:t1', 'c1', 'f1:ddd', '444'
```

```
put 'ns1:t1', 'c1', 'f1:eee', '555'
```

```

hbase(main):027:0>
hbase(main):028:0* scan 'ns1:t1'
ROW                                COLUMN+CELL
c1                                  column=f1:ccc, timestamp=1571725923735, value=333
c1                                  column=f1:ddd, timestamp=1571726035796, value=444
c1                                  column=f1:eee, timestamp=1571726420618, value=555
1 row(s)
Took 0.0110 seconds
hbase(main):029:0>

```

```
create 'ns1:t2', {NAME => 'f1', VERSIONS => 5}
```

二、大批量load Hfile数据测试

```

hadoop fs -mkdir /tmp/TestTable/
hadoop fs -cp
/hbase3/data/default/TestTable/0faa3eb85e44bd8a9d6c5ce1c43d22de/info0
/tmp/TestTable/
hadoop fs -cp
/hbase3/data/default/TestTable/e3717b09ac095cda1878b6c909fd8bc6/info0
/tmp/TestTable/
hadoop fs -chown -R hbase:hbase /tmp/TestTable/

sudo -u hbase hbase org.apache.hadoop.hbase.mapreduce.LoadIncrementalHFiles -
Dhbase.mapreduce.bulkload.max.hfiles.perRegion.perFamily=2048 /tmp/TestTable/
TestTable1

```

查看备份集群 TestTable1表 数据为空

结论：备份集群是没有同步数据

三、跨集群迁移数据-Snapshot 实现

<https://www.jianshu.com/p/22ee73b62927>

1. 创建快照(怎么判断是否完成)

```
snapshot 'TestTable', 'TestTable-20191022'
```

2. 导出快照到备份集群

```
sudo -u hbase hbase org.apache.hadoop.hbase.snapshot.ExportSnapshot --snapshot  
TestTable-20191022 --copy-from hdfs://ns1/hbase3 --copy-to hdfs://ns1/hbase9 --  
overwrite
```

3. 用快照恢复数据，它需要先禁用表，再进行恢复

```
disable 'TestTable'  
drop 'TestTable'  
restore_snapshot 'TestTable-20191022'
```

注意： 测试下来需要先删除 在恢复。

验证

```
count 'TestTable', INTERVAL => 100000
```