目录

[1. 实验一：HBase组件安装与配置 2](#_Toc14561)

[1.1. 实验目的 2](#_Toc31586)

[1.2. 实验要求 2](#_Toc15085)

[1.3. 实验环境 2](#_Toc20689)

[1.4. 实验过程 2](#_Toc5259)

[1.4.1. 实验任务一：HBase安装与配置 2](#_Toc28767)

[1.4.2. 实验任务二：HBase常用Shell命令 7](#_Toc585)

# 实验一：HBase组件安装与配置

## 实验目的

完成本实验，您应该能够：

* 掌握HBase安装与配置
* 掌握HBase常用Shell命令

## 实验要求

* 了解HBase原理
* 熟悉HBase常用Shell命令

## 实验环境

本实验所需之主要资源环境如表1-1所示。

表1-1 资源环境

|  |  |
| --- | --- |
| **服务器集群** | 3个以上节点，节点间网络互通，各节点最低配置：双核CPU、8GB内存、100G硬盘 |
| **运行环境** | CentOS 7.4 |
| **大数据平台** | H3C教学与实践管理系统 |
| **服务和组件** | 完成前面章节实验，其他服务及组件根据实验需求安装 |

## 实验过程

### 实验任务一：HBase安装与配置

#### 步骤一：解压缩HBase安装包

[root@master ~]# tar -zxvf /opt/software/hbase-1.2.1-bin.tar.gz -C /usr/local/src/

#### 步骤二：重命名HBase安装文件夹

[root@master ~]# cd /usr/local/src/

[root@master src]# mv hbase-1.2.1 hbase

#### 步骤三：在所有节点添加环境变量

[root@master src]# vi /etc/profile

# set hbase environment

export HBASE\_HOME=/usr/local/src/hbase

export PATH=$HBASE\_HOME/bin:$PATH

[root@master src]# scp /etc/profile root@slave1:/etc/

**root@slave1's password:**

**profile 100% 2366 2.3MB/s 00:00**

[root@master src]# scp /etc/profile root@slave2:/etc/

**root@slave2's password:**

**profile 100% 2366 903.9KB/s 00:00**

#### 步骤四：在master节点进入配置文件目录

[root@master ~]# cd /usr/local/src/hbase/conf/

#### 步骤五：在master节点配置hbase-env.sh文件

#在文件中修改

[root @master conf]# vi hbase-env.sh

#Java安装位置

export JAVA\_HOME=/usr/local/src/java

#值为true使用HBase自带的ZooKeeper,值为false使用在Hadoop上装的ZooKeeper

export HBASE\_MANAGES\_ZK=false

#HBase类路径

export HBASE\_CLASSPATH=/usr/local/src/hadoop/etc/hadoop/

#### 步骤六：在master节点配置hbase-site.xml

[root @master conf]# vi hbase-site.xml

<property>

<name>hbase.rootdir</name>

<value>hdfs://master:9000/hbase</value> # 使用9000端口

<description>The directory shared by region servers.</description>

</property>

<property>

<name>hbase.master.info.port</name>

<value>60010</value> # 使用master节点60010端口

</property>

<property>

<name>hbase.zookeeper.property.clientPort</name>

<value>2181</value> # 使用master节点2181端口

<description>Property from ZooKeeper's config zoo.cfg. The port at which the clients will connect.

</description>

</property>

<property>

<name>zookeeper.session.timeout</name>

<value>120000</value> # ZooKeeper超时时间

</property>

<property>

<name>hbase.zookeeper.quorum</name>

<value>master,slave1,slave2</value> # ZooKeeper管理节点

</property>

<property>

<name>hbase.tmp.dir</name>

<value>/usr/local/src/hbase/tmp</value> # HBase临时文件路径

</property>

<property>

<name>hbase.cluster.distributed</name>

<value>true</value> # 使用分布式HBase

</property>

hbase.rootdir：该项配置了数据写入的目录，默认hbase.rootdir是指向/tmp/hbase-${user.name}，也就说你会在重启后丢失数据(重启的时候操作系统会清理/tmp目录)。

hbase.zookeeper.property.clientPort：指定zk的连接端口

zookeeper.session.timeout：RegionServer与ZooKeeper间的连接超时时间。当超时时间到后，ReigonServer会被ZooKeeper从RS集群清单中移除，HMaster收到移除通知后，会对这台server负责的regions重新balance，让其他存活的RegionServer接管.

hbase.zookeeper.quorum：默认值是localhost，列出zookeepr ensemble中的servers

hbase.master.info.port：浏览器的访问端口

#### 步骤七：在master节点修改regionservers文件

#删除localhost，每一行写一个slave节点主机机器名

[root @master conf]$ vi regionservers

slave1

slave2

#### 步骤八：在master节点创建hbase.tmp.dir目录

[root @master usr]# mkdir /usr/local/src/hbase/tmp

#### 步骤九：将master上的hbase安装文件同步到slave1 slave2

[root @master ~]# scp -r /usr/local/src/hbase/ root@slave1:/usr/local/src/

[root @master ~]# scp -r /usr/local/src/hbase/ root@slave2:/usr/local/src/

#### 步骤十：在所有节点修改hbase目录权限

master节点修改权限

[root @master ~]# chown -R hadoop:hadoop /usr/local/src/hbase/

slave1节点修改权限

[root @slave1 ~]# chown -R hadoop:hadoop /usr/local/src/hbase/

slave2节点修改权限

[root @slave2 ~]# chown -R hadoop:hadoop /usr/local/src/hbase/

#### 步骤十一：在所有节点切换到hadoop用户并使环境变量生效

master节点

[root @master ~]# su - hadoop

[root @master ~]# source /etc/profile

slave1节点

[root @slave1 ~]# su - hadoop

[root @slave1 ~]# source /etc/profile

slave2节点

[root @slave2~]# su - hadoop

[root @slave2~]# source /etc/profile

#### 步骤十二：启动HBase

先启动Hadoop，然后启动ZooKeeper，最后启动HBase。

首先在master节点启动Hadoop。

[hadoop@master ~]$ start-all.sh

[hadoop@master ~]$ jps

master节点

[hadoop@master ~]$ jps

**10288 ResourceManager**

**9939 NameNode**

**10547 Jps**

**10136 SecondaryNameNode**

slave1节点

[hadoop@slave1 ~]$ jps

**4465 NodeManager**

**4356 DataNode**

**4584 Jps**

slave2节点

[hadoop@slave2 ~]$ jps

**3714 DataNode**

**3942 Jps**

**3823 NodeManager**

#### 步骤十三：在所有节点启动ZooKeeper

master节点

[hadoop@master ~]$ zkServer.sh start

[hadoop@master ~]$ jps

**10288 ResourceManager**

**9939 NameNode**

**10599 Jps**

**10136 SecondaryNameNode**

**10571 QuorumPeerMain**

slave1节点

[hadoop@slave1 ~]$ zkServer.sh start

[hadoop@slave1 ~]$ jps

**1473 QuorumPeerMain**

**1302 NodeManager**

**1226 DataNode**

**1499 Jps**

slave2节点

[hadoop@slave2 ~]$ zkServer.sh start

[hadoop@slave2 ~]$ jps

**1296 NodeManager**

**1493 Jps**

**1222 DataNode**

**1469 QuorumPeerMain**

#### 步骤十四：在master节点启动HBase

注意：各节点之间时间必须同步，否则HBase启动不了。

在每个节点执行date命令，查看每个节点的时间是否同步，不同步的话，在各节点执行date命令，date -s "2016-04-15 12:00:00"。

master节点

[hadoop@master ~]$ start-hbase.sh

[hadoop@master ~]$ jps

**1669 ResourceManager**

**2327 Jps**

**1322 NameNode**

**2107 HMaster**

**1948 QuorumPeerMain**

**1517 SecondaryNameNode**

slave1节点

[hadoop@slave1 ~]$ jps

**1473 QuorumPeerMain**

**1557 HRegionServer**

**1702 Jps**

**1302 NodeManager**

**1226 DataNode**

slave2节点

[hadoop@slave2 ~]$ jps

**1296 NodeManager**

**1222 DataNode**

**1545 HRegionServer**

**1725 Jps**

**1469 QuorumPeerMain**

#### 步骤十五：在浏览器输入master:60010出现如下图7-2所示的界面。

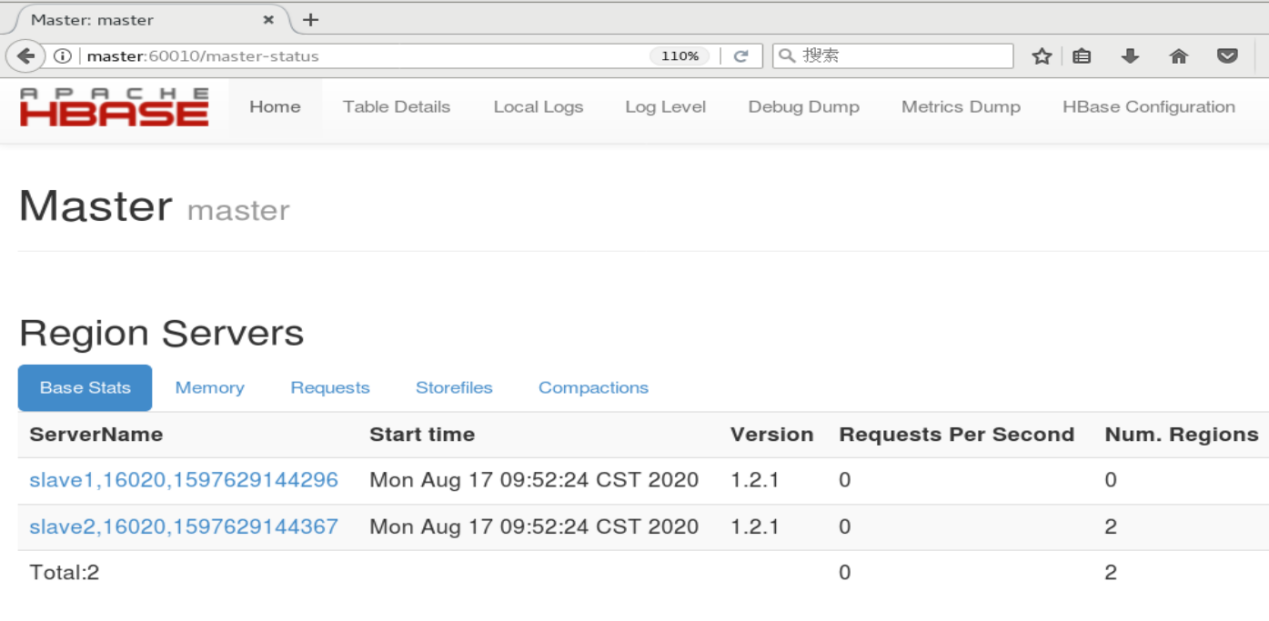


图7-2 HBase浏览器界面

注意：若slave显示不全，具体问题查看日志解决

### 实验任务二：HBase常用Shell命令

启动hdfs、zookeeper、hbase服务

#### 步骤一：进入HBase命令行

[hadoop@master ~]$ hbase shell

#### 步骤二：建立表scores，两个列簇：grade和course

hbase(main):001:0> create 'scores','grade','course'

**0 row(s) in 1.4480 seconds**

**=> Hbase::Table - scores**

#### 步骤三：查看数据库状态

hbase (main) :001 :0> status

**1 active master, 0 backup masters, 2 servers, 0 dead, 1.0000 average load**

#### 步骤四：查看数据库版本

hbase (main) :002:0> version

**1.2.1,r8d8a7107dc4ccbf36a92f64675dc60392f85c015,Wed Mar 30 11:19:21 CDT 20**

**16**

#### 步骤五：查看表

hbase(main):008:0> list

**TABLE**

**scores**

**1 row(s) in 0.0100 seconds**

**=>["scores"]**

#### 步骤六：插入记录1：zhangsan,grade: 143cloud

hbase(main):003:0> put 'scores','zhangsan','grade:','146cloud'

**0 row(s) in 0.2250 seconds**

#### 步骤七：插入记录2：zhangsan,course:math,86

hbase(main):004:0> put 'scores','zhangsan','course:math','86'

**0 row(s) in 0.0190 seconds**

#### 步骤八：插入记录3：zhangsan,course:cloud,92

hbase(main):005:0> put 'scores','zhangsan','course:cloud','92'

**0 row(s) in 0.0170 seconds**

#### 步骤九：插入记录4：lisi,grade:133soft

hbase(main):006:0> put 'scores','lisi','grade:','133soft'

**0 row(s) in 0.0070 seconds**

#### 步骤十：插入记录5：lisi,grade:math,87

hbase(main):007:0> put 'scores','lisi','course:math','87'

**0 row(s) in 0.0060 seconds**

#### 步骤十一：插入记录6：lisi,grade:cloud,96

hbase(main):008:0> put 'scores','lisi','course:cloud','96'

**0 row(s) in 0.0070 seconds**

#### 步骤十二：读取zhangsan的记录

hbase(main):009:0> get 'scores','zhangsan'

**COLUMN CELL**

**course:c Loud timestamp= 1460479208148, value=92**

**course : math timestamp=1460479163325，value=86**

**grade : timestamp=1460479064086，value= 146cloud**

**3 row(s) in 0.0800 seconds**

#### 步骤十三：读取zhangsan的班级

hbase(main):012:0> get 'scores','zhangsan','grade'

**COLUMN CELL**

**grade : timestamp= 1460479064086，value=146cloud**

**1 row( s) in 0.0150 seconds**

#### 步骤十四：查看整个表记录

hbase(main):013:0> scan 'scores'

**ROW COLUMN+CELL**

**lisi column=course:cloud, timestamp=1598492107507, value=96**

**lisi column=course:math, timestamp=1598492101697, value=87**

**lisi column=grade:, timestamp=1598492094568, value=133soft**

**zhangsan column=course:cloud, timestamp=1598492087700, value=92**

**zhangsan column=course:math, timestamp=1598492077530, value=86**

**zhangsan column=grade:, timestamp=1598492061508, value=146cloud**

**2 row(s) in 0.0570 seconds**

#### 步骤十五：按例查看表记录

hbase(main):014:0> scan 'scores',{COLUMNS=>'course'}

**ROW COLUMN+CELL**

**zhangsan column=course:cloud, timestamp= 1460479208148 ,value=92**

**zhangsan column=course:math, timestamp=1460479163325,value=86**

**lisi column=course:cloud, timestamp=1460479342925,value=96**

**lisi column=course:math, times tamp=1460479312963, value=87**

**2 row(s) in 0. 0230 seconds**

#### 步骤十六：删除指定记录

hbase(main):015:0> delete 'scores','lisi','grade'

**0 row(s) in 0.0390 seconds**

#### 步骤十七：删除后，执行scan命令

hbase(main):016:0> scan 'scores'

**ROW COLUMN+CELL**

**zhangsan column=course: cloud, timestamp=1460479208148,value=92**

**zhangsan column=course :math, timestamp= 1460479163325，value=86**

**zhangsan co lumn=grade: ,timestamp= 1460479064086，value=146cloud**

**lisi column=course :cloud, timestamp= 1460479342925, value=96**

**lisi column=course : math, timestamp= 1460479312963， value=87**

**row( s) in 0. 0350 seconds**

#### 步骤十八：增加新的列簇

hbase(main):017:0> alter 'scores',NAME=>'age'

**Updating all regions with the new schema.**

**0/ 1 regions updated.**

**1/ 1 regions updated.**

**Done.**

**0 row(s) in 3.0060 seconds**

#### 步骤十九：查看表结构

hbase(main):018:0> describe 'scores'

**Table scores is ENABLED**

**scores**

**COLUMN FAMIL IES DESCRIPTION**

**{NAME => age'，BL O0MFILTER =>ROW'，VERSIONS => '1' ，IN\_ MEMORY => 'false', KEEP DELETED\_ CELLS =>FAL SE'，DATA BLOCK ENCODING =>**

**NONETTL => ' FOREVER', COMPRESSION=>NONE'，MIN VERSIONS => '0' ，**

**BLOCKCACHE => ' true BLOCKSIZE =>65536'，REPLICATION\_ SCOPE => '0'}**

**{NAME =>course，BLOOMFILTER => ' ROW', VERSIONS =>IN MEMORY =>**

**false', KEEPDELETED CELLS =>FALSE', DATA BLOCK ENCODING =>**

**1 NONE'，TTL =>FOREVER'，COMPRESSION =>NONE'，MIN VERSIONS => '0'**

**BLOCKCACHE => 'true' ，BLOCKSIZE =>65536'，REPLICATION SCOPE =>0'}{NAME => grade BLOOMFILTER => 'ROW' ，VERSIONS =>IN\_ MEMORY => ' false', KEEPDELETED CELLS =>FALSE'，DATA BLOCK ENCODING =>1 NONE 'TTL =>FOREVER'，COMPRESSION =>NONE'，MIN VERSIONS => '0'**

**BLOCKCACHE => 'true' ，BLOCKSIZE => '65536 ，REPLICATION SCOPE => '0'}**

**3 row(s) in 0.0400 seconds**

#### 步骤二十：删除列簇

hbase(main):020:0> alter 'scores',NAME=>'age',METHOD=>'delete'

**Updating all regions with the new schema…**

**1/ 1 regions updated.**

**Done，**

**0 row(s)in 2.1600seconds**

#### 步骤二十一：删除表

hbase(main):021:0> disable 'scores'

**0 row(s)in 2.2930seconds**

**hbase(main):022:0> drop 'scores'**

**0 row(s)in 1.2530seconds**

**hbase( main) :023:0> list**

**TABLE**

**0 row(s)in 0.0150 seconds**

**==>[]**

#### 步骤二十二：退出

hbase(main):024:0> quit

[hadoopdmaste r~]$

#### 步骤二十三：关闭HBase

在master节点关闭HBase。

[hadoop@master ~]$ stop-hbase.sh

在所有节点关闭ZooKeeper。

[hadoop@master ~]$ zkServer.sh stop

[hadoop@slave1 ~]$ zkServer.sh stop

[hadoop@slave2 ~]$ zkServer.sh stop

在master节点关闭Hadoop。

[hadoop@master ~]$ stop-all.sh