

## 目录

1 实验一：高可用集群启动 .....	2
1.1. 实验目的 .....	2
1.2. 实验要求 .....	2
1.3. 实验环境 .....	2
1.4. 实验视图 .....	2
1.5. 实验过程 .....	2
1.5.1. 实验任务一：HA 的启动 .....	2
1.5.1.1. 步骤一：启动 journalnode 守护进程 .....	2
1.5.1.2. 步骤二：初始化 namenode .....	3
1.5.1.3. 步骤三：注册 ZNode .....	3
1.5.1.4. 步骤四：启动 hdfs .....	3
1.5.1.5. 步骤五：启动 yarn .....	4
1.5.1.6. 步骤六：同步 master 数据 .....	4
1.5.1.7. 步骤七：在 slave1 上启动 resourcemanager 和 namenode 进程 .....	5
1.5.1.8. 步骤九：启动 MapReduce 任务历史服务器 .....	5
1.5.1.9. 步骤十：查看端口和进程 .....	5
1.5.2. 实验任务二：HA 的测试 .....	7
1.5.2.1. 步骤一：创建一个测试文件 .....	7
1.5.2.2. 步骤二：在 hdfs 创建文件夹 .....	7
1.5.2.3. 步骤三：将 a.txt 传输到 input 上 .....	7
1.5.2.4. 步骤四：进入到 jar 包测试文件目录下 .....	7
1.5.2.5. 步骤五：测试 mapreduce .....	7
1.5.2.6. 步骤六：查看 hdfs 下的传输结果 .....	9
1.5.2.7. 步骤七：查看文件测试的结果 .....	9
1.5.3. 实验任务三：高可用性验证 .....	9
1.5.3.1. 步骤一：自动切换服务状态 .....	9
1.5.3.2. 步骤二：手动切换服务状态 .....	9

# 1 实验一：高可用集群启动

## 1.1. 实验目的

完成本实验，您应该能够：

- 掌握高可用的手动，自动切换
- 掌握高可用的启动

## 1.2. 实验要求

- 熟悉常用 Linux 操作系统命令
- 熟悉 Hadoop HA 集群规划部署

## 1.3. 实验环境

本实验所需之主要资源环境如表 1-1 所示。

服务器集群	3 个节点，节点间网络互通，各节点最低配置：X 核 CPU、XGB 内存、XG 硬盘
运行环境	CentOS 7.4
用户名/密码	root/password hadoop/password
服务和组件	ZooKeeper，其他服务根据实验需求安装

表 1-1 资源环境

## 1.4. 实验视图

高可用 ZooKeeper 集群部署实验部署流程如图 1-1

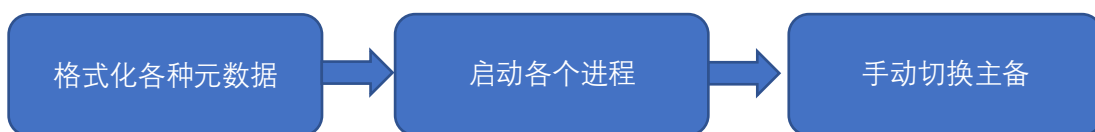


图 1-1 部署流程

## 1.5. 实验过程

### 1.5.1. 实验任务一：HA 的启动

#### 1.5.1.1. 步骤一：启动 journalnode 守护进程

```
[hadoop@master ~]$ hadoop-daemons.sh start journalnode
```

```

master:      starting      journalnode,      logging      to
/usr/local/src/hadoop/logs/hadoop-root-journalnode-master.out
slave1:      starting      journalnode,      logging      to
/usr/local/src/hadoop/logs/hadoop-root-journalnode-slave1.out
slave2:      starting      journalnode,      logging      to
  
```

/usr/local/src/hadoop/logs/hadoop-root-journalnode-slave2.out

### 1.5.1.2. 步骤二: 初始化 namenode

[hadoop@master ~]\$ hdfs namenode -format

```
20/07/21 09:29:00 INFO util.GSet: 0.029999999329447746% max memory 889 MB = 273.1 KB
20/07/21 09:29:00 INFO util.GSet: capacity = 2^15 = 32768 entries
20/07/21 09:29:01 INFO namenode.FSImage: Allocated new BlockPoolId: BP-137083702-192.168.90.205-1595294941233
20/07/21 09:29:01 INFO common.Storage: Storage directory /usr/local/src/hadoop/tmp/hdfs/nn has been successfully formatted.
20/07/21 09:29:01 INFO namenode.NNStorageRetentionManager: Going to retain 1 images with txid >= 0
20/07/21 09:29:01 INFO util.ExitUtil: Exiting with status 0
20/07/21 09:29:01 INFO namenode.NameNode: SHUTDOWN_MSG:
/*****
SHUTDOWN_MSG: Shutting down NameNode at master/192.168.90.205
*****/
```

### 1.5.1.3. 步骤三: 注册 ZNode

[hadoop@master ~]\$ hdfs zkfc -formatZK

```
20/07/01 17:23:15 INFO zookeeper.ZooKeeper: Client
environment:java.library.path=/usr/local/src/hadoop/lib:/usr/local/src/hadoop/lib/native
20/07/01 17:23:15 INFO zookeeper.ZooKeeper: Client environment:java.io.tmpdir=/tmp
20/07/01 17:23:15 INFO zookeeper.ZooKeeper: Client environment:java.compiler=<NA>
20/07/01 17:23:15 INFO zookeeper.ZooKeeper: Client environment:os.name=Linux
20/07/01 17:23:15 INFO zookeeper.ZooKeeper: Client environment:os.arch=amd64
20/07/01 17:23:15 INFO zookeeper.ZooKeeper: Client
environment:os.version=3.10.0-693.el7.x86_64
20/07/01 17:23:15 INFO zookeeper.ZooKeeper: Client environment:user.name=root
20/07/01 17:23:15 INFO zookeeper.ZooKeeper: Client environment:user.home=/root
20/07/01 17:23:15 INFO zookeeper.ZooKeeper: Client
environment:user.dir=/usr/local/src/hadoop/etc/hadoop
20/07/01 17:23:15 INFO zookeeper.ZooKeeper: Initiating client connection,
connectString=master:2181,slave1:2181,slave2:2181 sessionTimeout=5000
watcher=org.apache.hadoop.ha.ActiveStandbyElector$WatcherWithClientRef@27ce24aa
20/07/01 17:23:15 INFO zookeeper.ClientCnxn: Opening socket connection to server
slave2/192.168.1.8:2181. Will not attempt to authenticate using SASL (unknown error)
20/07/01 17:23:15 INFO zookeeper.ClientCnxn: Socket connection established to
slave2/192.168.1.8:2181, initiating session
20/07/01 17:23:15 INFO zookeeper.ClientCnxn: Session establishment complete on server
slave2/192.168.1.8:2181, sessionId = 0x373099bfa8c0000, negotiated timeout = 5000
20/07/01 17:23:15 INFO ha.ActiveStandbyElector: Successfully created /hadoop-ha/ns in ZK.
20/07/01 17:23:15 INFO zookeeper.ZooKeeper: Session: 0x373099bfa8c0000 closed
20/07/01 17:23:15 WARN ha.ActiveStandbyElector: Ignoring stale result from old client with
sessionId 0x373099bfa8c0000
20/07/01 17:23:15 INFO zookeeper.ClientCnxn: EventThread shut down
```

### 1.5.1.4. 步骤四: 启动 hdfs

[hadoop@master ~]\$ start-dfs.sh

Starting namenodes on [master slave1]

```

master:          starting          namenode,          logging          to
/usr/local/src/hadoop/logs/hadoop-root-namenode-master.out
slave1:          starting          namenode,          logging          to
/usr/local/src/hadoop/logs/hadoop-root-namenode-slave1.out
master:          starting          datanode,          logging          to
/usr/local/src/hadoop/logs/hadoop-root-datanode-master.out
slave1:          starting          datanode,          logging          to
/usr/local/src/hadoop/logs/hadoop-root-datanode-slave1.out
slave2:          starting          datanode,          logging          to
/usr/local/src/hadoop/logs/hadoop-root-datanode-slave2.out

```

Starting journal nodes [master slave1 slave2]

master: journalnode running as process 1787. Stop it first.

slave2: journalnode running as process 1613. Stop it first.

slave1: journalnode running as process 1634. Stop it first.

Starting ZK Failover Controllers on NN hosts [master slave1]

slave1: starting zkfc, logging to /usr/local/src/hadoop/logs/hadoop-root-zkfc-slave1.out

master: starting zkfc, logging to /usr/local/src/hadoop/logs/hadoop-root-zkfc-master.out

#### 1.5.1.5. 步骤五: 启动 yarn

**[hadoop@master ~]\$ start-yarn.sh**

starting yarn daemons

```

starting          resourcemanager,          logging          to
/usr/local/src/hadoop/logs/yarn-root-resourcemanager-master.out
master:          starting          nodemanager,          logging          to
/usr/local/src/hadoop/logs/yarn-root-nodemanager-master.out
slave1:          starting          nodemanager,          logging          to
/usr/local/src/hadoop/logs/yarn-root-nodemanager-slave1.out
slave2:          starting          nodemanager,          logging          to
/usr/local/src/hadoop/logs/yarn-root-nodemanager-slave2.out

```

#### 1.5.1.6. 步骤六: 同步 master 数据

复制 namenode 元数据到其它节点(在 master 节点执行)

```

[hadoop@master ~]$ scp -r /usr/local/src/hadoop/tmp/hdfs/nn/*
slave1:/usr/local/src/hadoop/tmp/hdfs/nn/
[hadoop@master ~]$ scp -r /usr/local/src/hadoop/tmp/hdfs/nn/*
slave2:/usr/local/src/hadoop/tmp/hdfs/nn/

```

#### 1.5.1.7. 步骤七：在 slave1 上启动 resourcemanager 和 namenode 进程

### 1.5.1.8. 步骤九：启动 MapReduce 任务历史服务器

### 1.5.1.9. 步骤十：查看端口和进程

```
[hadoop@slave1 ~]$ jps
[hadoop@slave1 ~]$ jps
3681 ResourceManager
3092 NameNode
3366 DFSZKFailoverController
3478 NodeManager
3003 JournalNode
3835 Jps
3213 DataNode
2926 QuorumPeerMain
```

```
[hadoop@slave2 ~]$ jps
```

```
[hadoop@slave2 ~]$ jps
3056 DataNode
3427 Jps
2887 QuorumPeerMain
3223 NodeManager
2957 JournalNode
```

master:50070

The screenshot shows the 'Overview' page for the master NameNode (master:8020) in an active state. The page is titled 'Overview 'master:8020' (active)'. It contains a table with the following information:

Namespace:	mycluster
Namenode ID:	master
Started:	Tue Jul 21 14:25:26 CST 2020
Version:	2.7.1, r15ecc87ccf4a0228f35af08fc56de536e6ce657a
Compiled:	2015-06-29T06:04Z by jenkins from (detached from 15ecc87)
Cluster ID:	CID-7f4c7850-341d-4b28-80f4-f11665009243
Block Pool ID:	BP-137083702-192.168.90.205-1595294941233

Below the table is a 'Summary' section. The browser address bar shows 'master:50070/dfshealth.html#tab-overview'.

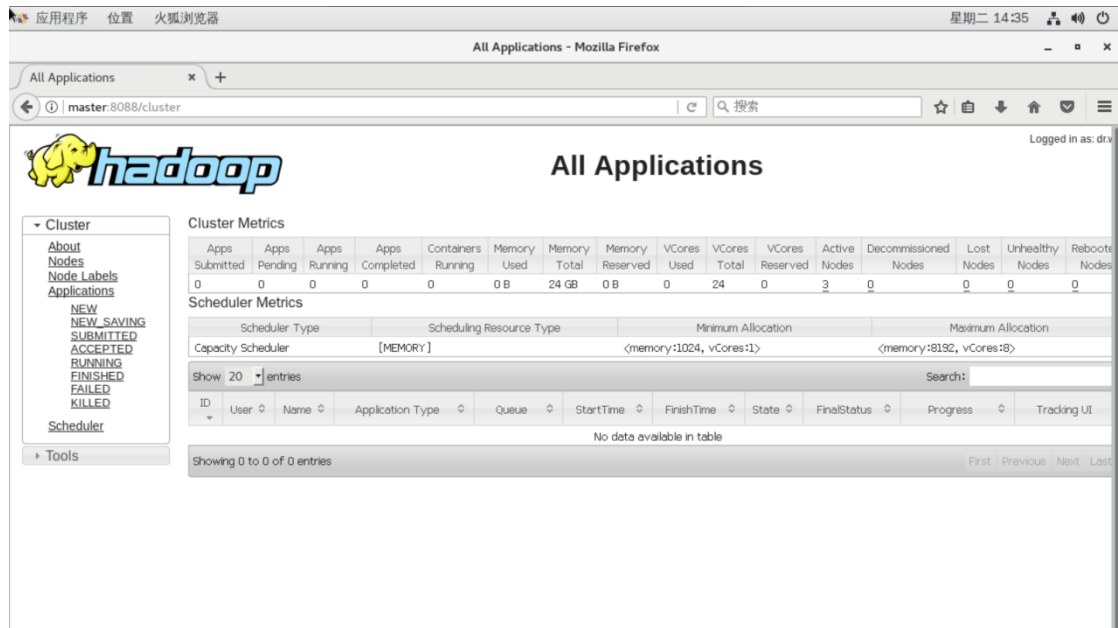
slave1:50070

The screenshot shows the 'Overview' page for the slave NameNode (slave1:8020) in a standby state. The page is titled 'Overview 'slave1:8020' (standby)'. It contains a table with the following information:

Namespace:	mycluster
Namenode ID:	slave1
Started:	Tue Jul 21 14:25:26 CST 2020
Version:	2.7.1, r15ecc87ccf4a0228f35af08fc56de536e6ce657a
Compiled:	2015-06-29T06:04Z by jenkins from (detached from 15ecc87)
Cluster ID:	CID-7f4c7850-341d-4b28-80f4-f11665009243
Block Pool ID:	BP-137083702-192.168.90.205-1595294941233

Below the table is a 'Summary' section. The browser address bar shows 'slave1:50070/dfshealth.html#tab-overview'. On the right side of the browser window, there are tabs for 'ssh', 'vnc', and 'console'.

master:8088



## 1.5.2. 实验任务二：HA 的测试

### 1.5.2.1. 步骤一：创建一个测试文件

```
[hadoop@master ~]$ vi a.txt
```

//内容如下:

Hello World

Hello Hadoop

### 1.5.2.2. 步骤二：在 hdfs 创建文件夹

```
[hadoop@master ~]$ hadoop fs -mkdir /input
```

### 1.5.2.3. 步骤三：将 a.txt 传输到 input 上

```
[hadoop@master ~]$ hadoop fs -put ~/a.txt /input
```

### 1.5.2.4. 步骤四：进入到 jar 包测试文件目录下

```
[hadoop@master ~]$ cd /usr/local/src/hadoop/share/hadoop/mapreduce/
```

### 1.5.2.5. 步骤五：测试 mapreduce

```
[hadoop@master mapreduce]$ hadoop jar hadoop-mapreduce-examples-2.7.1.jar wordcount /input/a.txt /output
```

成功如下:

```
[hadoop@master mapreduce]$ hadoop jar hadoop-mapreduce-examples-2.7.1.jar wordcount /input/a
.txt /output
20/07/21 14:54:45 INFO input.FileInputFormat: Total input paths to process : 1
20/07/21 14:54:45 INFO mapreduce.JobSubmitter: number of splits:1
20/07/21 14:54:45 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1595312762674_
0001
20/07/21 14:54:46 INFO impl.YarnClientImpl: Submitted application application_1595312762674_
0001
20/07/21 14:54:46 INFO mapreduce.Job: The url to track the job: http://master:8088/proxy/app
lication_1595312762674_0001/
20/07/21 14:54:46 INFO mapreduce.Job: Running job: job_1595312762674_0001
20/07/21 14:54:52 INFO mapreduce.Job: Job job_1595312762674_0001 running in uber mode : fals
e
20/07/21 14:54:52 INFO mapreduce.Job: map 0% reduce 0%
20/07/21 14:54:57 INFO mapreduce.Job: map 100% reduce 0%
20/07/21 14:55:02 INFO mapreduce.Job: map 100% reduce 100%
20/07/21 14:55:02 INFO mapreduce.Job: Job job_1595312762674_0001 completed successfully
20/07/21 14:55:02 INFO mapreduce.Job: Counters: 49
    File System Counters
        FILE: Number of bytes read=43
        FILE: Number of bytes written=236323
        FILE: Number of read operations=0
        FILE: Number of large read operations=0
        FILE: Number of write operations=0
        HDFS: Number of bytes read=18
        HDFS: Number of bytes written=25
        HDFS: Number of read operations=6
        HDFS: Number of large read operations=0
        HDFS: Number of write operations=2
    Job Counters
        Launched map tasks=1
        Launched reduce tasks=1
        Data-local map tasks=1
```

```
Total time spent by all maps in occupied slots (ms)=2682
Total time spent by all reduces in occupied slots (ms)=3220
Total time spent by all map tasks (ms)=2682
Total time spent by all reduce tasks (ms)=3220
Total vcore-seconds taken by all map tasks=2682
Total vcore-seconds taken by all reduce tasks=3220
Total megabyte-seconds taken by all map tasks=2746368
Total megabyte-seconds taken by all reduce tasks=3297280
Map-Reduce Framework
    Map input records=2
    Map output records=4
    Map output bytes=41
    Map output materialized bytes=43
    Input split bytes=93
    Combine input records=4
    Combine output records=3
    Reduce input groups=3
    Reduce shuffle bytes=43
    Reduce input records=3
    Reduce output records=3
    Spilled Records=6
    Shuffled Maps =1
    Failed Shuffles=0
    Merged Map outputs=1
    GC time elapsed (ms)=107
    CPU time spent (ms)=1150
    Physical memory (bytes) snapshot=432898048
    Virtual memory (bytes) snapshot=4240924672
    Total committed heap usage (bytes)=322961408
Shuffle Errors
    BAD_ID=0
    CONNECTION=0
    IO_ERROR=0
```



```

WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
File Input Format Counters:
  Bytes Read=25
File Output Format Counters:
  Bytes Written=25
[hadoop@master mapreduce]$

```

#### 1.5.2.6. 步骤六: 查看 hdfs 下的传输结果

```

[hadoop@master mapreduce]$ hadoop fs -lsr /output
[hadoop@master mapreduce]$ hadoop fs -lsr /output
lsr: DEPRECATED: Please use 'ls -R' instead.
-rw-r--r--  2 hadoop supergroup      0 2020-07-21 14:55 /output/_SUCCESS
-rw-r--r--  2 hadoop supergroup    25 2020-07-21 14:55 /output/part-r-00000
[hadoop@master mapreduce]$

```

#### 1.5.2.7. 步骤七: 查看文件测试的结果

```

[hadoop@master mapreduce]$ hadoop fs -cat /output/part-r-00000

```

Hadoop 1

Hello 2

World 1

### 1.5.3. 实验任务三: 高可用性验证

#### 1.5.3.1. 步骤一: 自动切换服务状态

输入代码:

```

[hadoop@master mapreduce]$ cd

```

```

#hdfs haadmin -failover --forcefence --forceactive 主 备

```

```

[hadoop@master ~]$ hdfs haadmin -failover --forcefence --forceactive slave1 master

```

查看状态

```

[hadoop@master ~]$ hdfs haadmin -getServiceState slave1

```

```

[hadoop@master ~]$ hdfs haadmin -getServiceState slave1
standby

```

```

[hadoop@master ~]$ hdfs haadmin -getServiceState master

```

```

[hadoop@master ~]$ hdfs haadmin -getServiceState master
active

```

#### 1.5.3.2. 步骤二: 手动切换服务状态

在 master 停止并启动 namenode

```

[hadoop@master ~]$ hadoop-daemon.sh stop namenode

```

stopping namenode

查看状态

```

[hadoop@master ~]$ hdfs haadmin -getServiceState master

```

```

[hadoop@master ~]$ hdfs haadmin -getServiceState slave1

```

```
[hadoop@master hadoop]$ hdfs haadmin -getServiceState master
20/07/21 15:14:25 INFO ipc.Client: Retrying connect to server: master/192.168.90.205:8020. A
lready tried 0 time(s); retry policy is RetryUpToMaximumCountWithFixedSleep(maxRetries=1, sl
eepTime=1000 MILLISECONDS)
Operation failed: Call From master/192.168.90.205 to master:8020 failed on connection except
ion: java.net.ConnectException: 拒绝连接; For more details see: http://wiki.apache.org/hadoo
p/ConnectionRefused
[hadoop@master hadoop]$ hdfs haadmin -getServiceState slave1
active
```

**[hadoop@master ~]\$ hadoop-daemon.sh start namenode**

```
[hadoop@master hadoop]$ hadoop-daemon.sh start namenode
starting namenode, logging to /usr/local/src/hadoop/logs/hadoop-hadoop-namenode-master.out
```

查看状态

**[hadoop@master ~]\$ hdfs haadmin -getServiceState slave1**

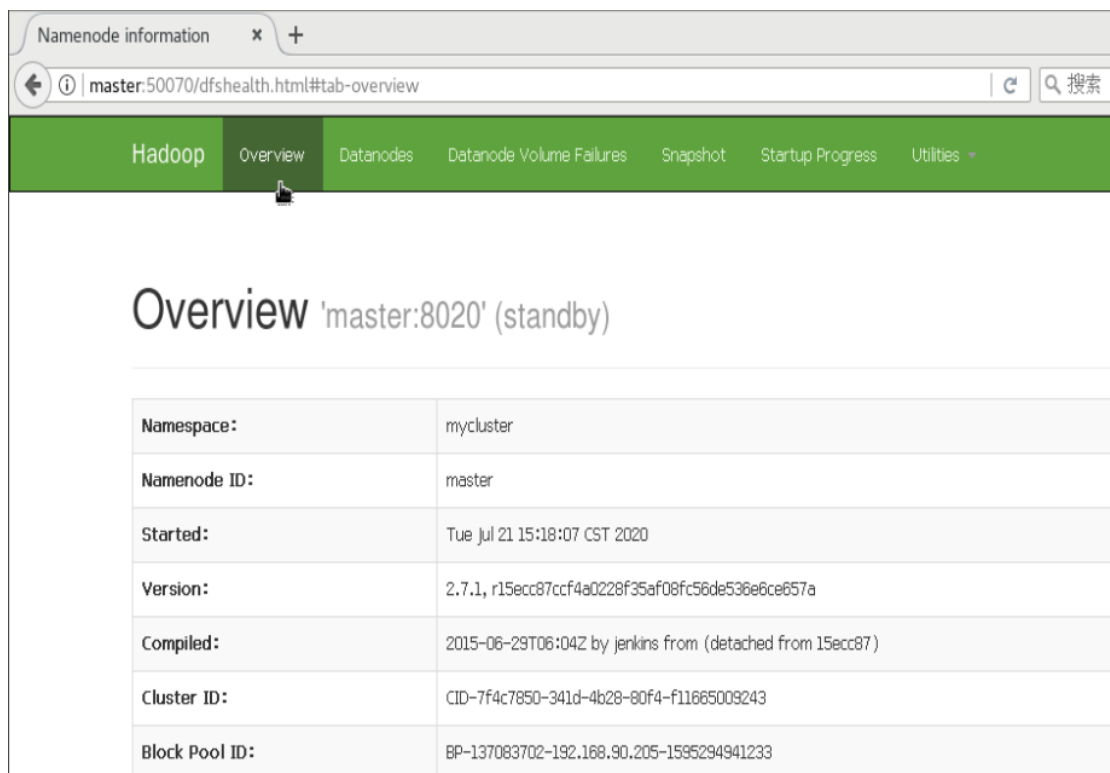
```
[hadoop@master hadoop]$ hdfs haadmin -getServiceState slave1
active
```

**[hadoop@master ~]\$ hdfs haadmin -getServiceState master**

```
[hadoop@master hadoop]$ hdfs haadmin -getServiceState master
standby
```

查看 web 服务端

master:50070



Namenode information

master:50070/dfshealth.html#tab-overview

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

## Overview 'master:8020' (standby)

Namespace:	mycluster
Namenode ID:	master
Started:	Tue Jul 21 15:18:07 CST 2020
Version:	2.7.1, r15ecc87ccf4a0228f35af08fc56de536e6ce657a
Compiled:	2015-06-29T06:04Z by jenkins from (detached from 15ecc87)
Cluster ID:	CID-7f4c7850-341d-4b28-80f4-f11665009243
Block Pool ID:	BP-137083702-192.168.90.205-1595294941233

slave1:50070

Namenode information

slave1:50070/dfshealth.html#tab-overview

搜索

Hadoop

Overview

Datanodes

Datanode Volume Failures

Snapshot

Startup Progress

Utilities

## Overview 'slave1:8020' (active)

Namespace:	mycluster
Namenode ID:	slave1
Started:	Tue Jul 21 14:25:26 CST 2020
Version:	2.7.1, r15ecc87ccf4a0228f35af08fc56de536e6ce657a
Compiled:	2015-06-29T06:04Z by jenkins from (detached from 15ecc87)
Cluster ID:	CID-7f4c7850-341d-4b28-80f4-f11665009243
Block Pool ID:	BP-137083702-192.168.90.205-1595294941233