

推荐环境搭建教程

1、环境说明

提供两种环境，单机环境（1台centos，50G）以及分布式环境(3台centos，150G)。两个环境根据电脑配置选择

- 选择单机版环境：
 - 电脑物理内存：12G以下的，磁盘剩余空间小于200G
- 选择分布式环境：
 - 电脑物理内存：12G以上的，磁盘剩余空间大于等于200G

注：建议使用单机版环境，在项目中代码编写差别不大。单机版和分布式版本主要区别在于

- 1、环境的使用配置，分布式会多一些前期环境使用操作
 - 2、分布式环境遇到的hadoop、hbase、spark等组件问题会多一些
- python开发环境：
 - 默认黑马头条的开发环境包，已经安装在reco_sys的虚拟环境中
 - 单机版：就是在mastr中：**source activate reco_sys**
 - 分布式版：三台都必须安装相关虚拟环境和库(默认三台都安装了)
 - 开发项目就在master上面编程操作即可

注：分布式开发区别，就在于三台都需要安装相关虚拟环境，否则开发会出现一些未知库的错误

2、数据说明

注：头条数据以及模型数据文件都已经上传到hadoop平台，大家可以直接使用

- 单机版：HIVE没有建立相关表关联hadoop，需要各位同学在做项目的时候自己根据课件创建表关联
- 分布式版本都已经关联好。

3、前期配置

前提：VMware虚拟机设置NAT模式，VM虚拟网卡固定好IP不变

基础部分已经设置过，此部分可以忽略

4、单机版本使用教程

- 一台centos hadoop-master: 192.168.19.137

下面操作，每次关机之后开机都需要操作！！！！，建议每天结束VM待机centos

打开虚拟机后，登录账号密码:

```
用户: root
密码: itcast
```

- 1、先进行防火墙关闭，防止hadoop manage启动失败，注：操作只在master操作即可

```
[root@hadoop-master ~]# systemctl stop firewalld.service
```

- 2、后面开启HIVE需要mysql对应连接的元数据(默认HIVE关联本地mysql)，需要启动mysql,centos中mysql使用docker安装的，后续模型部署中也会用到docker，注：操作只在master操作即可

```
[root@hadoop-master ~]# docker start mysql
```

```
# 然后需要进入mysql,操作相关命令
```

```
docker exec -it mysql bash
```

```
mysql -uroot -p
```

```
密码: password
```

- 3、启动hadoop、hbase、spark以及hive，已经配置好一键启动的脚本在/root/scripts/目录中
 - 都在scripts目录下有一键启动hadoop,hbase,spark的脚本，也有一键关闭hadoop,hbase,spark的脚本,hive如果需要关闭直接kill即可
 - 注：操作只在master操作即可

```
[root@hadoop-master scripts]# pwd
```

```
/root/scripts
```

```
[root@hadoop-master scripts]# ls
```

```
all.sh my.cnf start.sh stop.sh
```

```
[root@hadoop-master scripts]#
```

```
# 开启hadoop, hbase, spark
```

```
[root@hadoop-master ~]#. start.sh
```

```
# 开启hive元数据服务
```

```
[root@hadoop-master ~]# hive --service metastore &
```

启动之后确定有以下内容或者浏览器查看状态

```
5249 DataNode
11541 HMaster
6422 NodeManager
12262 Master
5703 SecondaryNameNode
4840 NameNode
5865 ResourceManager
12634 Worker
12685 Jps
12436 RunJar
```

- 4、关闭hadoop安全模式，注：操作只在master操作即可

```
[root@hadoop-master ~]# hdfs dfsadmin -safemode leave
```

然后再浏览器中查看三个组件的状态，**hadoop:50070**端口，**hbase:16010**端口，**spark:8088**端口

- 5、hbase开发时候使用happybase，需要开启thriftserver,可以提前打开保持一致开启状态

```
[root@hadoop-master ~] hbase-daemon.sh start thrift
```

注：操作只在master操作即可

5、分布式环境使用教程

分布式环境相对于单机版前期要操作命令较多，总共有三台

- hadoop-master: 192.168.19.137
- hadoop-slave1: 192.168.19.138
- hadoop-slave2: 192.168.19.139

每次开关机之后都要重复操作下面这些步骤

- 1、首先还是一样开机三台centos之后，首先要做的操作，注：操作需要三台centos上面操作
 - 三台时间同步，后期HBase分布式需要三台时间差不超过30s
 - 三台防火墙都需要进行关闭

```
# 时间有误差问题同步一下：
```

```
[root@hadoop-master ~]# ntpdate 0.cn.pool.ntp.org
```

```
[root@hadoop-slave1 ~]# ntpdate 0.cn.pool.ntp.org
```

```
[root@hadoop-slave2 ~]# ntpdate 0.cn.pool.ntp.org
```

```
# 三台防火墙进行关闭
```

```
[root@hadoop-master ~]# systemctl stop firewalld.service
```

```
[root@hadoop-slave1 ~]# systemctl stop firewalld.service
```

```
[root@hadoop-slave2 ~]# systemctl stop firewalld.service
```

- 2、与单机版本一样，开启mysql,后面开启HIVE需要mysql对应连接的元数据(默认HIVE关联本地mysql)，需要启动mysql,centos中mysql使用docker安装的，后续模型部署中也会用到docker，注：操作只在master操作即可

```
[root@hadoop-master ~]# docker start mysql
```

```
# 然后需要进入mysql,操作相关命令
```

```
docker exec -it mysql bash
```

```
mysql -uroot -p
```

```
密码: password
```

- 3、启动hadoop、hbase、spark以及hive，已经配置好一键启动的脚本在/root/scripts/目录中
 - 都在scripts目录下有一键启动hadoop,hbase,spark的脚本，也有一键关闭hadoop,hbase,spark的脚本，hive如果需要关闭直接kill即可
 - start.sh与stop.sh，一键启动一键关闭，注：操作只在master操作即可

```
[root@hadoop-master scripts]# pwd
```

```
/root/scripts
```

```
[root@hadoop-master scripts]# ls
```

```
my.cnf  start.sh  stop.sh
```

```
[root@hadoop-master scripts]#
```

```
# 开启hadoop, hbase, spark
```

```
[root@hadoop-master ~]# . start.sh
```

```
# 开启hive元数据服务
```

```
[root@hadoop-master ~]# hive --service metastore &
```

在浏览器中查看分布式的状态，确定与下图一致

- 4、hbase开发时候使用happybase，需要开启thriftserver,可以提前打开保持一致开启状态，注：操作只在master操作即可

```
[root@hadoop-master ~] hbase-daemon.sh start thrift
```

如果开启之后确定每台机器启动结果如下：

- hadoop-master

已经开启hadoop、hbase、spark、hive

```
20160 Jps
18786 Master
4131 RunJar # hive
17395 ResourceManager
19219 Worker
16757 NameNode
17206 SecondaryNameNode
18683 HRegionServer
8637 ThriftServer # happybase使用
18253 HMaster
18159 HQuorumPeer
```

- hadoop-slave1、hadoop-slave2

开启hadoop、hbase、spark

```
3857 NodeManager
4290 Worker
4680 Jps
3740 DataNode
3980 HQuorumPeer
4093 HRegionServers
```

hadoop

← → ↻ 不安全 | 192.168.19.137:50070/explorer.html#/user/hive/warehouse/profile.db/user_action ☆ ↗

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Browse Directory

/user/hive/warehouse/profile.db/user_action Go! 📁 📄 🗑️

Show 25 entries Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	root	supergroup	0 B	Apr 11 12:17	0	0 B	2019-03-05	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	root	supergroup	0 B	Apr 11 12:17	0	0 B	2019-03-06	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	root	supergroup	0 B	Apr 11 12:18	0	0 B	2019-03-07	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	root	supergroup	0 B	Apr 11 12:18	0	0 B	2019-03-08	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	root	supergroup	0 B	Apr 11 12:18	0	0 B	2019-03-09	<input type="checkbox"/>
<input type="checkbox"/>	drwxrwxrwx	root	supergroup	0 B	Apr 13 19:34	0	0 B	2019-03-10	<input type="checkbox"/>

hbase

Master hadoop-master

Region Servers

Base Stats Memory Requests Storefiles Compactions

ServerName	Start time	Last contact	Version	Requests Per Second
hadoop-master,16020,1557848619567	Tue May 14 23:43:39 CST 2019	0 s	2.0.3	0
hadoop-slave1,16020,1557973272368	Thu May 16 10:21:12 CST 2019	88 s	2.0.3	0
hadoop-slave2,16020,1557848611142	Tue May 14 23:43:31 CST 2019	90 s	2.0.3	0
Total:3				0

spark



RUNNING Applications

- Cluster
- About
- Nodes
- Node Labels
- Applications
- NEW
- NEW SAVING
- SUBMITTED
- ACCEPTED
- RUNNING
- FINISHED
- FAILED
- KILLED
- Scheduler
- Tools

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Re
228	0	0	228	0	0 B	16 GB	0 B

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes	Unhealthy Nodes
2	0	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocation
Capacity Scheduler	[MEMORY]	<memory:1024, vCores:1>	<memory:8192, vCores:4>

Show 20 entries

ID	User	Name	Application Type	Queue	Application Priority	StartTime	FinishTime	State	FinalStatus	Running Containers	Allocated CPU VCores	Allocated Memory MB	Reserved CPU VCores
No data available in table													

Showing 0 to 0 of 0 entries