# Hadoop集群安装配置教程\_Hadoop2.6.0\_Ubuntu/CentOS

 给力星 2014年8月10日 (updated: 2016年3月31日) 136158

[](http://dblab.xmu.edu.cn/post/bigdataroadmap/)

本教程讲述如何配置 Hadoop 集群，默认读者已经掌握了 Hadoop 的单机伪分布式配置，否则请先查看[Hadoop安装教程\_单机/伪分布式配置](http://dblab.xmu.edu.cn/blog/install-hadoop/" \t "_blank) 或 [CentOS安装Hadoop\_单机/伪分布式配置](http://dblab.xmu.edu.cn/blog/install-hadoop-in-centos/)。

本教程由[厦门大学数据库实验室](http://dblab.xmu.edu.cn/" \t "_blank)出品，转载请注明。本教程适合于原生 Hadoop 2，包括 Hadoop 2.6.0, Hadoop 2.7.1 等版本，主要参考了[官方安装教程](http://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/ClusterSetup.html" \t "_blank)，步骤详细，辅以适当说明，**保证按照步骤来，都能顺利安装并运行 Hadoop**。另外有[Hadoop安装配置简略版](http://dblab.xmu.edu.cn/blog/install-hadoop-simplify/" \t "_blank)方便有基础的读者快速完成安装。

**为了方便新手入门，我们准备了两篇不同系统的 Hadoop 伪分布式配置教程。但其他 Hadoop 教程我们将不再区分，可同时适用于 Ubuntu 和 CentOS/RedHat 系统。例如本教程以 Ubuntu 系统为主要演示环境，但对 Ubuntu/CentOS 的不同配置之处、CentOS 6.x 与 CentOS 7 的操作区别等都会尽量给出注明。**

## 环境

本教程使用 **Ubuntu 14.04 64位** 作为系统环境，基于原生 Hadoop 2，在 **Hadoop 2.6.0 (stable)** 版本下验证通过，可适合任何 Hadoop 2.x.y 版本，例如 Hadoop 2.7.1，Hadoop 2.4.1 等。

本教程简单的使用两个节点作为集群环境: 一个作为 Master 节点，局域网 IP 为 192.168.1.121；另一个作为 Slave 节点，局域网 IP 为 192.168.1.122。

## 准备工作

Hadoop 集群的安装配置大致为如下流程:

1. 选定一台机器作为 Master
2. 在 Master 节点上配置 hadoop 用户、安装 SSH server、安装 Java 环境
3. 在 Master 节点上安装 Hadoop，并完成配置
4. 在其他 Slave 节点上配置 hadoop 用户、安装 SSH server、安装 Java 环境
5. 将 Master 节点上的 /usr/local/hadoop 目录复制到其他 Slave 节点上
6. 在 Master 节点上开启 Hadoop

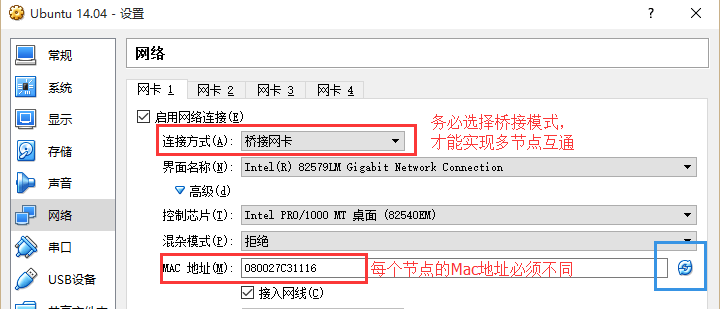
配置 hadoop 用户、安装 SSH server、安装 Java 环境、安装 Hadoop 等过程已经在[Hadoop安装教程\_单机/伪分布式配置](http://dblab.xmu.edu.cn/blog/install-hadoop/" \t "_blank) 或 [CentOS安装Hadoop\_单机/伪分布式配置](http://dblab.xmu.edu.cn/blog/install-hadoop-in-centos/)中有详细介绍，请前往查看，不再重复叙述。

**继续下一步配置前，请先完成上述流程的前 4 个步骤**。

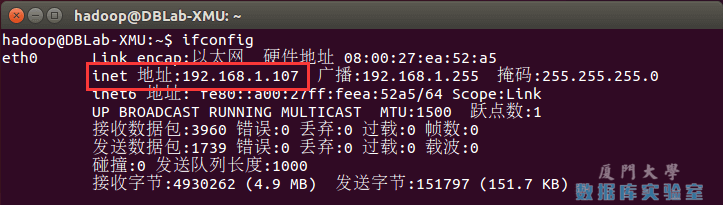
## 网络配置

假设集群所用的节点都位于同一个局域网。

如果使用的是虚拟机安装的系统，那么需要更改网络连接方式为桥接（Bridge）模式，才能实现多个节点互连，例如在 VirturalBox 中的设置如下图。此外，如果节点的系统是在虚拟机中直接复制的，要确保各个节点的 Mac 地址不同（可以点右边的按钮随机生成 MAC 地址，否则 IP 会冲突）：

VirturalBox中节点的网络设置

Linux 中查看节点 IP 地址的命令为 ifconfig，即下图所示的 inet 地址（**注意虚拟机安装的 CentoS 不会自动联网，需要点右上角连上网络才能看到 IP 地址**）：

Linux查看IP命令

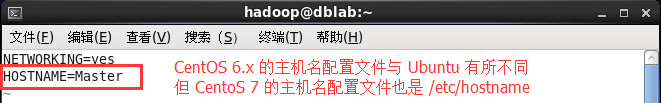
首先在 Master 节点上完成准备工作，并关闭 Hadoop (/usr/local/hadoop/sbin/stop-dfs.sh)，再进行后续集群配置。

为了便于区分，可以修改各个节点的主机名（在终端标题、命令行中可以看到主机名，以便区分）。在 Ubuntu/CentOS 7 中，我们在 Master 节点上执行如下命令修改主机名（即改为 Master，注意是区分大小写的）：

1. sudo vim /etc/hostname

Shell 命令

如果是用 CentOS 6.x 系统，则是修改 /etc/sysconfig/network 文件，改为 HOSTNAME=Master，如下图所示：

CentOS中hostname设置

然后执行如下命令修改自己所用节点的IP映射：

1. sudo vim /etc/hosts

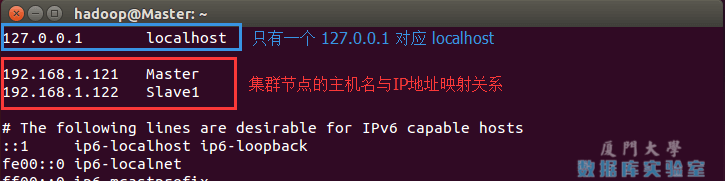
Shell 命令

例如本教程使用两个节点的名称与对应的 IP 关系如下：

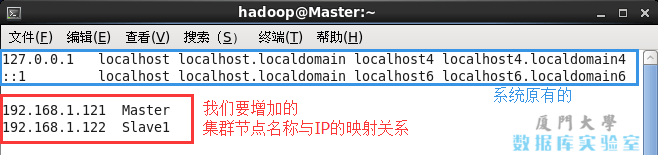
192.168.1.121 Master

192.168.1.122 Slave1

我们在 /etc/hosts 中将该映射关系填写上去即可，如下图所示（一般该文件中只有一个 127.0.0.1，其对应名为 localhost，如果有多余的应删除，特别是不能有 “127.0.0.1 Master” 这样的记录）：

Hadoop中的hosts设置

CentOS 中的 /etc/hosts 配置则如下图所示：

CentOS中的hosts设置

**修改完成后需要重启一下，重启后在终端中才会看到机器名的变化**。接下来的教程中请注意区分 Master 节点与 Slave 节点的操作。

**需要在所有节点上完成网络配置**

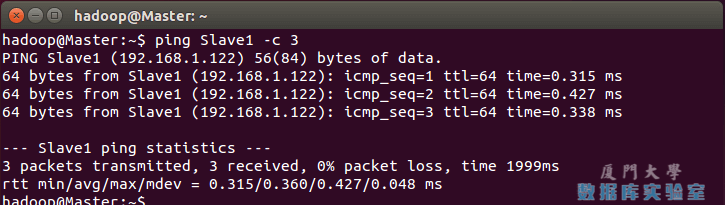
。如上面讲的是 Master 节点的配置，而在其他的 Slave 节点上，也要对 /etc/hostname（修改为 Slave1、Slave2 等） 和 /etc/hosts（跟 Master 的配置一样）这两个文件进行修改！

配置好后需要在各个节点上执行如下命令，测试是否相互 ping 得通，如果 ping 不通，后面就无法顺利配置成功：

1. ping Master -c 3 *# 只ping 3次，否则要按 Ctrl+c 中断*
2. ping Slave1 -c 3

Shell 命令

例如我在 Master 节点上 ping Slave1，ping 通的话会显示 time，显示的结果如下图所示：

检查是否ping得通

**继续下一步配置前，请先完成所有节点的网络配置，修改过主机名的话需重启才能生效**。

## SSH无密码登陆节点

这个操作是要让 Master 节点可以无密码 SSH 登陆到各个 Slave 节点上。

首先生成 Master 节点的公匙，在 Master 节点的终端中执行（因为改过主机名，所以还需要删掉原有的再重新生成一次）：

1. cd ~/.ssh *# 如果没有该目录，先执行一次ssh localhost*
2. rm ./id\_rsa\* *# 删除之前生成的公匙（如果有）*
3. ssh-keygen -t rsa *# 一直按回车就可以*

Shell 命令

让 Master 节点需能无密码 SSH 本机，在 Master 节点上执行：

1. cat ./id\_rsa.pub >> ./authorized\_keys

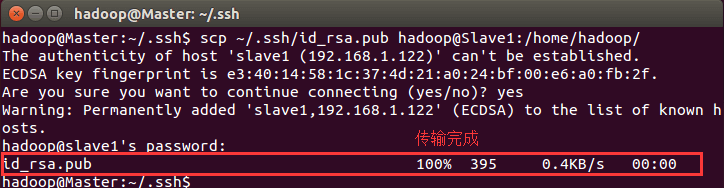
Shell 命令

完成后可执行 ssh Master 验证一下（可能需要输入 yes，成功后执行 exit 返回原来的终端）。接着在 Master 节点将上公匙传输到 Slave1 节点：

1. scp ~/.ssh/id\_rsa.pub hadoop@Slave1:/home/hadoop/

Shell 命令

scp 是 secure copy 的简写，用于在 Linux 下进行远程拷贝文件，类似于 cp 命令，不过 cp 只能在本机中拷贝。执行 scp 时会要求输入 Slave1 上 hadoop 用户的密码(hadoop)，输入完成后会提示传输完毕，如下图所示：

通过scp向远程主机拷贝文件

接着在 Slave1 节点上，将 ssh 公匙加入授权：

1. mkdir ~/.ssh *# 如果不存在该文件夹需先创建，若已存在则忽略*
2. cat ~/id\_rsa.pub >> ~/.ssh/authorized\_keys
3. rm ~/id\_rsa.pub *# 用完就可以删掉了*

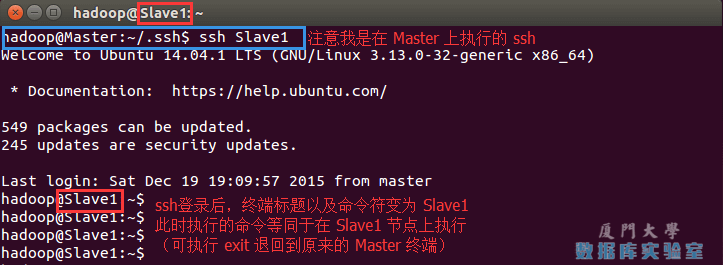
Shell 命令

如果有其他 Slave 节点，也要执行将 Master 公匙传输到 Slave 节点、在 Slave 节点上加入授权这两步。

这样，在 Master 节点上就可以无密码 SSH 到各个 Slave 节点了，可在 Master 节点上执行如下命令进行检验，如下图所示：

1. ssh Slave1

Shell 命令

在Master节点中ssh到Slave节点

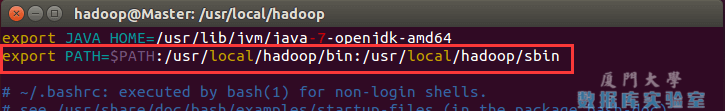
## 配置PATH变量

（CentOS 单机配置 Hadoop 的教程中有配置这一项了，这一步可以跳过）

在单机伪分布式配置教程的最后，说到可以将 Hadoop 安装目录加入 PATH 变量中，这样就可以在任意目录中直接使用 hadoo、hdfs 等命令了，如果还没有配置的，需要在 Master 节点上进行配置。首先执行 vim ~/.bashrc，加入一行：

export PATH=$PATH:/usr/local/hadoop/bin:/usr/local/hadoop/sbin

如下图所示：

配置PATH变量

保存后执行 source ~/.bashrc 使配置生效。

## 配置集群/分布式环境

集群/分布式模式需要修改 /usr/local/hadoop/etc/hadoop 中的5个配置文件，更多设置项可点击查看官方说明，这里仅设置了正常启动所必须的设置项： slaves、[core-site.xml](http://hadoop.apache.org/docs/r2.6.0/hadoop-project-dist/hadoop-common/core-default.xml" \t "_blank)、[hdfs-site.xml](http://hadoop.apache.org/docs/r2.6.0/hadoop-project-dist/hadoop-hdfs/hdfs-default.xml" \t "_blank)、[mapred-site.xml](http://hadoop.apache.org/docs/r2.6.0/hadoop-mapreduce-client/hadoop-mapreduce-client-core/mapred-default.xml" \t "_blank)、[yarn-site.xml](http://hadoop.apache.org/docs/r2.6.0/hadoop-yarn/hadoop-yarn-common/yarn-default.xml" \t "_blank) 。

1, 文件 **slaves**，将作为 DataNode 的主机名写入该文件，每行一个，默认为 localhost，所以在伪分布式配置时，节点即作为 NameNode 也作为 DataNode。分布式配置可以保留 localhost，也可以删掉，让 Master 节点仅作为 NameNode 使用。

本教程让 Master 节点仅作为 NameNode 使用，因此将文件中原来的 localhost 删除，只添加一行内容：Slave1。

2, 文件 **core-site.xml** 改为下面的配置：

1. <configuration>
2. <property>
3. <name>fs.defaultFS</name>
4. <value>hdfs://Master:9000</value>
5. </property>
6. <property>
7. <name>hadoop.tmp.dir</name>
8. <value>file:/usr/local/hadoop/tmp</value>
9. <description>Abase for other temporary directories.</description>
10. </property>
11. </configuration>

XML

3, 文件 **hdfs-site.xml**，dfs.replication 一般设为 3，但我们只有一个 Slave 节点，所以 dfs.replication 的值还是设为 1：

1. <configuration>
2. <property>
3. <name>dfs.namenode.secondary.http-address</name>
4. <value>Master:50090</value>
5. </property>
6. <property>
7. <name>dfs.replication</name>
8. <value>1</value>
9. </property>
10. <property>
11. <name>dfs.namenode.name.dir</name>
12. <value>file:/usr/local/hadoop/tmp/dfs/name</value>
13. </property>
14. <property>
15. <name>dfs.datanode.data.dir</name>
16. <value>file:/usr/local/hadoop/tmp/dfs/data</value>
17. </property>
18. </configuration>

XML

4, 文件 **mapred-site.xml** （可能需要先重命名，默认文件名为 mapred-site.xml.template），然后配置修改如下：

1. <configuration>
2. <property>
3. <name>mapreduce.framework.name</name>
4. <value>yarn</value>
5. </property>
6. <property>
7. <name>mapreduce.jobhistory.address</name>
8. <value>Master:10020</value>
9. </property>
10. <property>
11. <name>mapreduce.jobhistory.webapp.address</name>
12. <value>Master:19888</value>
13. </property>
14. </configuration>

XML

5, 文件 **yarn-site.xml**：

1. <configuration>
2. <property>
3. <name>yarn.resourcemanager.hostname</name>
4. <value>Master</value>
5. </property>
6. <property>
7. <name>yarn.nodemanager.aux-services</name>
8. <value>mapreduce\_shuffle</value>
9. </property>
10. </configuration>

XML

配置好后，将 Master 上的 /usr/local/Hadoop 文件夹复制到各个节点上。因为之前有跑过伪分布式模式，建议在切换到集群模式前先删除之前的临时文件。在 Master 节点上执行：

1. cd /usr/local
2. sudo rm -r ./hadoop/tmp *# 删除 Hadoop 临时文件*
3. sudo rm -r ./hadoop/logs/\* *# 删除日志文件*
4. tar -zcf ~/hadoop.master.tar.gz ./hadoop *# 先压缩再复制*
5. cd ~
6. scp ./hadoop.master.tar.gz Slave1:/home/hadoop

Shell 命令

在 Slave1 节点上执行：

1. sudo rm -r /usr/local/hadoop *# 删掉旧的（如果存在）*
2. sudo tar -zxf ~/hadoop.master.tar.gz -C /usr/local
3. sudo chown -R hadoop /usr/local/hadoop

Shell 命令

同样，如果有其他 Slave 节点，也要执行将 hadoop.master.tar.gz 传输到 Slave 节点、在 Slave 节点解压文件的操作。

首次启动需要先在 Master 节点执行 NameNode 的格式化：

1. hdfs namenode -format *# 首次运行需要执行初始化，之后不需要*

Shell 命令

**CentOS系统需要关闭防火墙**

CentOS系统默认开启了防火墙，在开启 Hadoop 集群之前，**需要关闭集群中每个节点的防火墙**。有防火墙会导致 ping 得通但 telnet 端口不通，从而导致 DataNode 启动了，但 Live datanodes 为 0 的情况。

在 CentOS 6.x 中，可以通过如下命令关闭防火墙：

1. sudo service iptables stop *# 关闭防火墙服务*
2. sudo chkconfig iptables off *# 禁止防火墙开机自启，就不用手动关闭了*

Shell 命令

若用是 CentOS 7，需通过如下命令关闭（防火墙服务改成了 firewall）：

1. systemctl stop firewalld.service *# 关闭firewall*
2. systemctl disable firewalld.service *# 禁止firewall开机启动*

Shell 命令

如下图，是在 CentOS 6.x 中关闭防火墙：

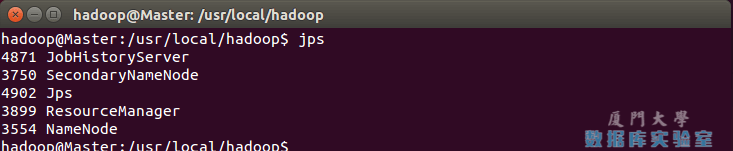
CentOS6.x系统关闭防火墙

接着可以启动 hadoop 了，启动需要在 Master 节点上进行：

1. start-dfs.sh
2. start-yarn.sh
3. mr-jobhistory-daemon.sh start historyserver

Shell 命令

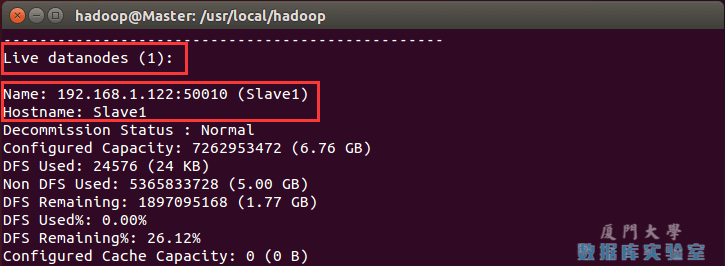
通过命令 jps 可以查看各个节点所启动的进程。正确的话，在 Master 节点上可以看到 NameNode、ResourceManager、SecondrryNameNode、JobHistoryServer 进程，如下图所示：

通过jps查看Master的Hadoop进程

在 Slave 节点可以看到 DataNode 和 NodeManager 进程，如下图所示：

通过jps查看Slave的Hadoop进程

缺少任一进程都表示出错。另外还需要在 Master 节点上通过命令 hdfs dfsadmin -report 查看 DataNode 是否正常启动，如果 Live datanodes 不为 0 ，则说明集群启动成功。例如我这边一共有 1 个 Datanodes：

通过dfsadmin查看DataNode的状态

也可以通过 Web 页面看到查看 DataNode 和 NameNode 的状态：[http://master:50070/](http://master:50070/" \t "_blank)。如果不成功，可以通过启动日志排查原因。

**伪分布式、分布式配置切换时的注意事项**

1, 从分布式切换到伪分布式时，不要忘记修改 slaves 配置文件；  
2, 在两者之间切换时，若遇到无法正常启动的情况，可以删除所涉及节点的临时文件夹，这样虽然之前的数据会被删掉，但能保证集群正确启动。所以如果集群以前能启动，但后来启动不了，特别是 DataNode 无法启动，不妨试着删除所有节点（包括 Slave 节点）上的 /usr/local/hadoop/tmp 文件夹，再重新执行一次 hdfs namenode -format，再次启动试试。

## 执行分布式实例

执行分布式实例过程与伪分布式模式一样，首先创建 HDFS 上的用户目录：

1. hdfs dfs -mkdir -p /user/hadoop

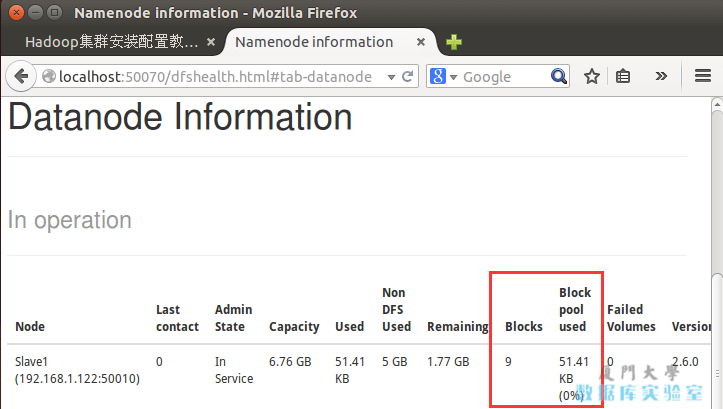
Shell 命令

将 /usr/local/hadoop/etc/hadoop 中的配置文件作为输入文件复制到分布式文件系统中：

1. hdfs dfs -mkdir input
2. hdfs dfs -put /usr/local/hadoop/etc/hadoop/\*.xml input

Shell 命令

通过查看 DataNode 的状态（占用大小有改变），输入文件确实复制到了 DataNode 中，如下图所示：

通过Web页面查看DataNode的状态

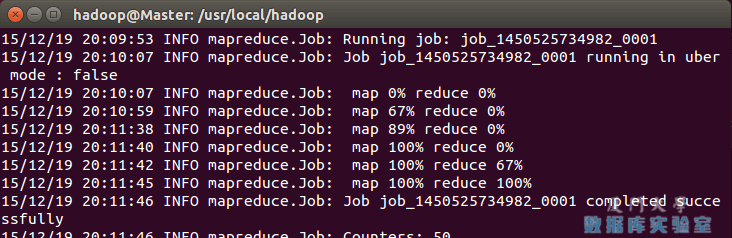
接着就可以运行 MapReduce 作业了：

1. hadoop jar /usr/local/hadoop/share/hadoop/mapreduce/hadoop-mapreduce-examples-\*.jar grep input output 'dfs[a-z.]+'

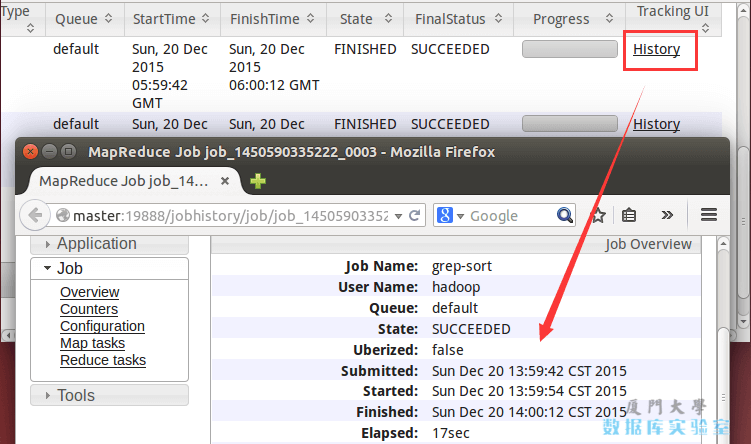
Shell 命令

运行时的输出信息与伪分布式类似，会显示 Job 的进度。

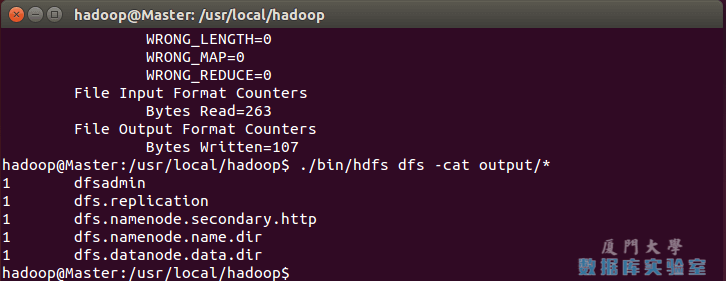
可能会有点慢，但如果迟迟没有进度，比如 5 分钟都没看到进度，那不妨重启 Hadoop 再试试。若重启还不行，则很有可能是内存不足引起，建议增大虚拟机的内存，或者通过更改 YARN 的内存配置解决。

显示MapReduce Job的进度

同样可以通过 Web 界面查看任务进度 <http://master:8088/cluster>，在 Web 界面点击 “Tracking UI” 这一列的 History 连接，可以看到任务的运行信息，如下图所示：

通过Web页面查看集群和MapReduce作业的信息

执行完毕后的输出结果：

MapReduce作业的输出结果

关闭 Hadoop 集群也是在 Master 节点上执行的：

1. stop-yarn.sh
2. stop-dfs.sh
3. mr-jobhistory-daemon.sh stop historyserver

Shell 命令

此外，同伪分布式一样，也可以不启动 YARN，但要记得改掉 mapred-site.xml 的文件名。

自此，你就掌握了 Hadoop 的集群搭建与基本使用了。

## 相关教程

* [使用Eclipse编译运行MapReduce程序](http://dblab.xmu.edu.cn/blog/hadoop-build-project-using-eclipse/): 用文本编辑器写 Java 程序是不靠谱的，还是用 Eclipse 比较方便。
* [使用命令行编译打包运行自己的MapReduce程序](http://dblab.xmu.edu.cn/blog/hadoop-build-project-by-shell/): 有时候需要直接通过命令来编译 MapReduce 程序。

## 参考资料

* <http://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/ClusterSetup.html>

本文作者



[**给力星**](http://dblab.xmu.edu.cn/blog/author/powerxing/)

你有多渴望，你有多付出

[www.powerxing.com](http://www.powerxing.com/)moc.621@gnixnimiac

<http://dblab.xmu.edu.cn/blog/install-hadoop-cluster/> [大数据](http://dblab.xmu.edu.cn/blog/category/big-data/)[Hadoop](http://dblab.xmu.edu.cn/blog/tag/hadoop/)

###### 相关文章

1. [**使用Eclipse编译运行MapReduce程序\_Hadoop2.6.0\_Ubuntu/CentOS**](http://dblab.xmu.edu.cn/blog/hadoop-build-project-using-eclipse/)
2. [**Hadoop 2.4.1单机版 自定义实现类以及编译运行**](http://dblab.xmu.edu.cn/blog/wordcount/)
3. [**使用命令行编译打包运行自己的MapReduce程序 Hadoop2.6.0**](http://dblab.xmu.edu.cn/blog/hadoop-build-project-by-shell/)
4. [**Hadoop安装配置简略教程**](http://dblab.xmu.edu.cn/blog/install-hadoop-simplify/)
5. [**Hadoop安装教程\_伪分布式配置\_CentOS6.4/Hadoop2.6.0**](http://dblab.xmu.edu.cn/blog/install-hadoop-in-centos/)
6. [**Linux安装hadoop-2.7.1**](http://dblab.xmu.edu.cn/blog/247/)