# Mapreduce 实验指导书

## 简介

mapreduce 通过“map”（映射）和“reduce”（归约）的概念，通过hadoop对文件进行批量处理，极大地方便了编程人员在不会分布式并行编程的情况下，将自己的程序运行在分布式系统上。本次实验中我们将在已经配置的hadoop基础上，配置mapreduce所需要的环境，并且运行hadoop的样例中的代码。由于mapreduce是基于hdfs文件系统来进行计算，我们也会用到之前配好的环境，并且第一次尝试使用这些工具运行代码。

## 环境配置(这部分在hadoop配置中已经完成)

### 1.

首先我们对yarn进行配置，即修改~/hadoop-2.7.7/etc/yarn-site.xml

#### yarn-nodemanager

首先配置nodemanager，使得yarn在打开时能够运行起nodemanager，yarn是hadoop的资源管理器，而nodemanager能够处理

- 单个节点上的资源管理和任务管理
- 来自ResourceManager的命令
- 处理来自ApplicationMaster的命令

因此我们在`~/hadoop-2.7.7/etc/hadoop/yarn-site.xml` `<configuration></configuration>`标签间与其他`<property></property>`标签并列地配置

```
<property>
    <name>yarn.nodemanager.resource.memory-mb</name>
    <value>1024</value>
</property>
<property>
    <name>yarn.nodemanager.resource.cpu-vcores</name>
    <value>1</value>
</property>
```

#### yarn-application-classpath

接下来继续配置yarn路径，使得代码可以访问到正确的类路径

```
<property>
  <name>yarn.application.classpath</name>
  <value>
    /home/ubuntu/hadoop-2.7.7/etc/hadoop,
    /home/ubuntu/hadoop-2.7.7/share/hadoop/common/*,
```

```
    /home/ubuntu/hadoop-2.7.7/share/hadoop/common/lib/*,
    /home/ubuntu/hadoop-2.7.7/share/hadoop/hdfs/*,
    /home/ubuntu/hadoop-2.7.7/share/hadoop/hdfs/lib/*,
    /home/ubuntu/hadoop-2.7.7/share/hadoop/mapreduce/*,
    /home/ubuntu/hadoop-2.7.7/share/hadoop/mapreduce/lib/*,
    /home/ubuntu/hadoop-2.7.7/share/hadoop/yarn/*,
    /home/ubuntu/hadoop-2.7.7/share/hadoop/yarn/lib/*
  </value>
</property>
```

## 2.

我们接下来配置`mapred-site.xml`，这是mapreduce这个模块主要的配置文件，按照如下代码配置，主要包括mapreduce的路径的修改以及一些基础设置

```
<configuration>
<property>
  <name>mapreduce.framework.name</name>
  <value>yarn</value>
  <description>Theruntime framework for executing MapReduce jobs. Can be one
of local,    classic oryarn.</description>
</property>

<property>
  <name>mapreduce.jobhistory.address</name>
  <value>master:10020</value>
  <description>MapReduce JobHistoryServer IPC host:port</description>
</property>

<property>
  <name>mapreduce.jobhistory.webapp.address</name>
  <value>master:19888</value>
  <description>MapReduce JobHistoryServer Web UI host:port</description>
</property>

<property>
  <name>mapreduce.application.classpath</name>
  <value>
  /home/ubuntu/hadoop-2.7.7/etc/hadoop,
  /home/ubuntu/hadoop-2.7.7/share/hadoop/common/*,
  /home/ubuntu/hadoop-2.7.7/share/hadoop/common/lib/*,
  /home/ubuntu/hadoop-2.7.7/share/hadoop/hdfs/*,
  /home/ubuntu/hadoop-2.7.7/share/hadoop/hdfs/lib/*,
  /home/ubuntu/hadoop-2.7.7/share/hadoop/mapreduce/*,
  /home/ubuntu/hadoop-2.7.7/share/hadoop/mapreduce/lib/*,
  /home/ubuntu/hadoop-2.7.7/share/hadoop/yarn/*,
  /home/ubuntu/hadoop-2.7.7/share/hadoop/yarn/lib/*
```

```
  </value>
</property>
</configuration>
```

**请注意，以上步骤在三个节点中都需要配置，因此请确认是否进行过前面的安装教程中的`scp`命令，该命令将文件从master复制到其他节点。**

# 样例运行

我们尝试运行hadoop自带的mapreduce例子word count 来对文件中的单词进行计数，来了解map-reduce的基本步骤。该样例能够对多个hdfs上某个文件夹下的所有文本进行读取，统计其中每个单词的个数。

## 1 文件准备(需提前启动hadoop)

- 首先我们创建实验所需要的文件，文件中的内容可以自定。所用到的命令为`touch,vim`两个命令

```
touch a.txt
vim a.txt
```

利用`vim`在文件中输入任意个单词，然后按esc，并输入:wq保存。

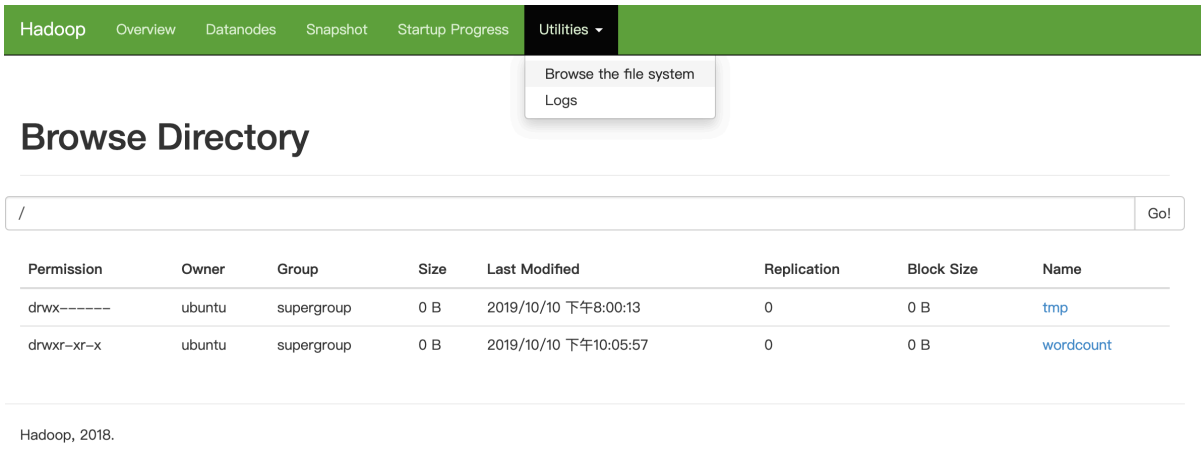
- 步骤同上，可以创建b.txt, c.txt等多个文件
- 在hdfs上建立文件夹input，hdfs上的操作的具体含义在hdfs的实验手册中有

```
hadoop fs -mkdir -p /wordcount/input
```

- 将我们创建的文件从本地上传到hdfs中我们创建的目录下，多个文件可以多次放

```
hadoop fs -put a.txt /wordcount/input
```

- 我们可以登录 http://master的外网ip地址:50070 (例如http://153.136.129.102:50070) 来查看hdfs上的文件是否正确存放，如下图，在Utilities->Browse the file system中可以看到文件的路径和位置




## 2 运行程序

运行MapReduce程序

- 进入jar包所在目录

```
cd /home/ubuntu/hadoop-2.7.7/share/hadoop/mapreduce
```

- 运行

```
hadoop jar hadoop-mapreduce-examples-2.7.7.jar wordcount /wordcount/input
/wordcount/output
```

其中一些具体参数：

  - wordcount: jar 包中需要运行的主类
  - /wordcount/input: wordcount主类需要的参数，指定数据文件目录，统计里面的数据文件
  - /wordcount/output: 统计文件后的结果保存目录，必须要求该目录不存在，运行后会创建该目录

## 3 查看输出

查看统计结果

```
hadoop fs -cat /wordcount/output/part-r-00000
```

```
ubuntu@master:~$ hadoop fs -cat /wordcount/output/part-r-00000
1       2
111     1
12      5
123     6
33      3
333     1
```