2018 年 12 月

哈爾濱工業大學

实验环境配置书

题　　目：多源的特定属性的社区查找算法设计(64)

专　　业：大数据专业

学　　号：1160300610

姓　　名：李思睿

课程类别：必修

本次实验重要在 linux 系统下完成，通过在 linux 系统下搭建 hadoop 来完成实验。

首先通过在 Windows 操作系统下使用开源虚拟机软件 VirtualBox 安装 Ubuntu18.04。Ubuntu18.04 系统的安装和配置在此不详细叙述。

下面主要详细介绍 hadoop 的安装及配置过程:

①安装 Java 环境。

为了方便起见直接安装 OpenJDk1.7，通过命令 sudo apt-get install openjdk-7-jre openjdk-7-jdk。安装完成之后，需要为 java 配置相应的环境变量，通过 vim ~/.bashrc 中增加 export JAVA_HOME=JDK 安装路径，来为 java 配置环境变量，最后通过 source ~/.bashrc 来使得配置生效。

②安装 Hadoop2.8.5（本人选用）。

首先到官网下载相应版本的安装包，然后通过命令 sudo tar -zxf ~/下载/hadoop-2.6.0.tar.gz -C /usr/local 将安装包解压到用户的本地目录上面去，并且通过命令 sudo chown -R hadoop ./hadoop 来修改文件夹的权限。

③配置 Hadoop 的模式（本人配置了单机模式已经伪分布式模式）

Hadoop 的默认模式为非分布式模式（本地模式），无需进行其他配置即可运行，在非分布式即单 Java 进程，方便进行调试。

Hadoop 的伪分布式配置，Hadoop 可以在单节点上通过伪分布式的方式运行，Hadoop 进程以分离的 Java 进程来运行，节点既作为 NameNode 也作为 DataNode，同时，读取 HDFS 中的文件。Hadoop 的配置文件位于/hadoop/etc/hadoop/中，伪分布式需要修改 2 个配置文件 core-site.xml 和 hdfs-site.xml。Hadoop 的配置文件时 xml 格式，每个配置以声明 property 的 name 和 value 的方式来实现。

首先修改配置文件./etc/hadoop/core-site.xml，将当中添加配置信息:

```
<configuration>
    <property>
        <name>hadoop.tmp.dir</name>
        <value>file:/usr/local/hadoop/tmp</value>
        <description>Abase for other temporary directories.
        </description>
    </property>
    <property>
        <name>fs.defaultFS</name>
        <value>hdfs://localhost:9000</value>
    </property>
</configuration>
```

其中第一个配置项为配置 hadoop 的临时存放目录，不过若没有配置 hadoop.tmp.dir 参数，则默认使用的临时目录为 /tmp/hadoop-hadoop，而这个目录在重启时有可能被系统清理掉，导致必须重新执行 format 才行。（也可以换到本地的某个文件夹下，避免有可能因设备问题而使得数据丢失）第二个配置项为 HDFS 文件系统的默认存放路径，其中 9000 为 fileSystem 的默认端口号。

然后修改配置文件 hdfs-site.xml，将当中添加配置信息：

```
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>1</value>
    </property>
    <property>
        <name>dfs.namenode.name.dir</name>
        <value>file:/usr/local/hadoop/tmp/dfs/name</value>
    </property>
    <property>
        <name>dfs.datanode.data.dir</name>
        <value>file:/usr/local/hadoop/tmp/dfs/data</value>
    </property>
</configuration>
```

其中配置了 HDFS 中主节点 namenode 的文件存放路径以及数据节点 datanode 的文件存放路径。（※在本次实验过程中，突然发现 HDFS 的文件系统空间不足，主节点 namenode 进入了 safe 模式，不能对 HDFS 文件系统中的任何文件进行操作，已经了文件系统的扩充，扩充的主要步骤就是要对此配置文件进行修改，改变主节点和数据节点的存储路径，在此我又给虚拟机额外分配了一块 30G 磁盘，并修改了配置文件）

配置完成后，执行 NameNode 的格式化./bin/hdfs namenode –format。

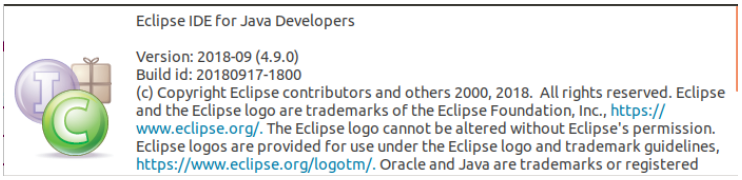
```
<property>
    <name>dfs.namenode.name.dir</name>
    <value>file:/home/hadoop/sda3/hadoop/tmp/dfs/name</value>
</property>
<property>
    <name>dfs.datanode.data.dir</name>
    <value>file:/home/hadoop/sda3/hadoop/tmp/dfs/data</value>
</property>
```

接着就可以开启 NameNode 和 DataNode 守护进程./sbin/start-dfs.sh。

成功启动后，可以访问 Web 界面 http://localhost:50070 来查看 NameNode 和 DataNode 的信息，还可以在线查看 HDFS 中的文件。

④下载 eclipse

首先去官网下载 Eclipse IDE for Java Developers 版本。



配置相应的环境变量：

```
export JRE_HOME=${JAVA_HOME}/jre
export CLASSPATH=.:${JAVA_HOME}/lib:${JRE_HOME}/lib
export PATH=.:${JAVA_HOME}/bin:$PATH
export HADOOP_HOME=/usr/local/hadoop
export CLASSPATH=$($HADOOP_HOME/bin/hadoop classpath):$CLASSPATH
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:/usr/local/hadoop/sbin:/usr/local/hadoop/bin
export GIRAPH_HOME=/usr/local/giraph
export HAMA_HOME=/usr/local/hama
export classpath=$classpath:/usr/local/hama/bin/:/usr/local/hama/lib/
```

④配置 eclipse 运行 map-reduce 程序

（在配置 eclipse 之前必须确保已经开启了 Hadoop）

首先需要先下载 hadoop-eclipse-plugin 的插件，并将其复制到 Eclipse 安装目录的 plugins 文件夹中，运行 eclipse –clean 来重启 eclipse 即可。

启动 eclipse 之后，就可以在左侧的 project Explorer 中看到 DFS Locations 了，然后选择 Windows 菜单下的 Preference，此时窗口会多出一个 Hadoop Map/Reduce 的选项，点击之后，选择 Hadoop 的安装目录即可。然后切换到 Map/Reduce 开发视图。

然后建立与 Hadoop 集群的连接，点击 Eclipse 软件右下角的 Map/Reduce Locations 面板，在面板中选择 New Hadoop Loction，在弹出来的 General 面板中，General 的设置要与 Hadoop 的设置一致，在伪分布式的情况下，填写 localhost，设置 fs.defaultFS 为 hdfs://localhost:9000，并且要将 DFS Master 的 Port 更改为 9000。Map/Reduce(V2) Master 的 port 用默认的即可。Advanced parameters 选项面板是对 Hadoop 参数进行配置，实际上就是填写 Hadoop 的配置项（/usr/local/hadoop/etc/hadoop 中的配置文件）。最后，点击 Finish，Map/Reduce Location 就创建好了。然后就可以创建 Map-Reduce 程序了。

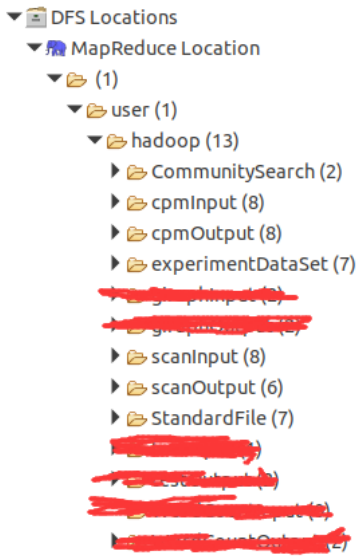
最后，在运行 Map-Reduce 程序之前，还需要执行一项重要的操作，就是将 /usr/local/hadoop/etc/hadoop 中修改过的配置文件（core-site.xml、hdfs-site.xml），以及 log4j.property 复制到程序项目的源文件夹 src 中，其中前两个文件使用于让程序跑在伪分布式环境下的，最后一个 log4j 文件适用于记录程序中的输出日志。

⑤创建 HDFS 目录，用于程序读取文件

首先在 HDFS 中创建用户目录。首先，执行 cd /usr/local/Hadoop 命令，然后执行./bin/hdfs dfs –mkdir –p /user/hadoop 来创建 hdfs 目录。

创建好目录之后，就需要将实验所需要的文件全部导入进去，首先通过命令./bin/hdfs dfs –put xxx 即可。

最后的 hdfs 效果如图所示：



每个代码文件中的 Executor.java 文件为程序的主函数文件，通过修改开头的 file 文件名称，来运行不同的程序。

至此，本次实验的系统就搭建完毕了！