## 1. 背景

当前，CORBA、DCOM、RMI等RPC中间件技术已广泛应用于各个领域。但是面对规模和复杂度都越来越高的分布式系统，这些技术也显示出其局限性：

（1）同步通信：客户发出调用后，必须等待服务对象完成处理并返回结果后才能继续执行；（2）客户和服务对象的生命周期紧密耦合：客户进程和服务对象进程都必须正常运行；

如果由于服务对象崩溃或者网络故障导致客户的请求不可达，客户会接收到异常；

（3）点对点通信：客户的一次调用只发送给某个单独的目标对象。

面向消息的中间件（Message Oriented Middleware，MOM）较好的解决了以上问题。发送者将消息发送给消息服务器，消息服务器将消息存放在若干队列中，在合适的时候再将消息转发给接收者。这种模式下，

1）发送和接收是异步的，发送者无需等待；

2）二者的生命周期未必相同：发送消息的时候接收者不一定运行，接收消息的时候发送者

也不一定运行；

3）一对多通信：对于一个消息可以有多个接收者。

已有的 MOM系统包括 IBM的 MQSeries、Microsoft的MSMQ和BEA的MessageQ等。

由于没有一个通用的标准，这些系统很难实现互操作和无缝连接。

Java Message Service（JMS）是 SUN提出的旨在统一各种 MOM系统接口的规范，它包含点对点（Point to Point，PTP）和发布 /订阅（Publish/Subscribe，pub/sub）两种消息模型，提供可靠消息传输、事务和消息过滤等机制。

## 3. Ubuntu上安装Hadoop

### 3.1在VMware Workstation 上安装Linux系统

1、安装虚拟机VMware Workstation，在软件安装目录已经提供

2、在虚拟机上安装Ubuntu系统，这里采用Ubuntu 12.04. 32位系统

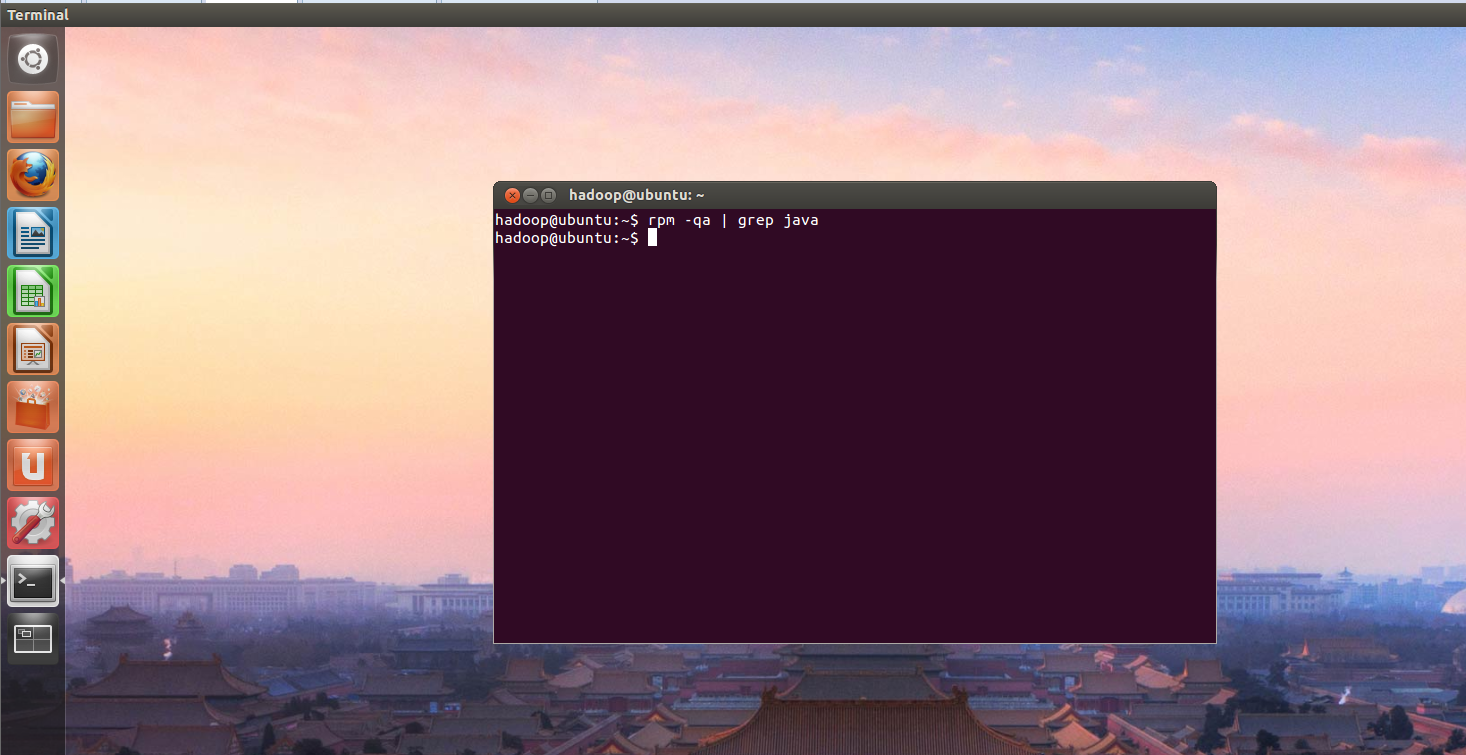
3、在Ubuntu上安装Hadoop2，安装版本是Hadoop2.9.0

【上面所需软件在软件安装目录已经提供】

### 3.2安装Unbutu系统

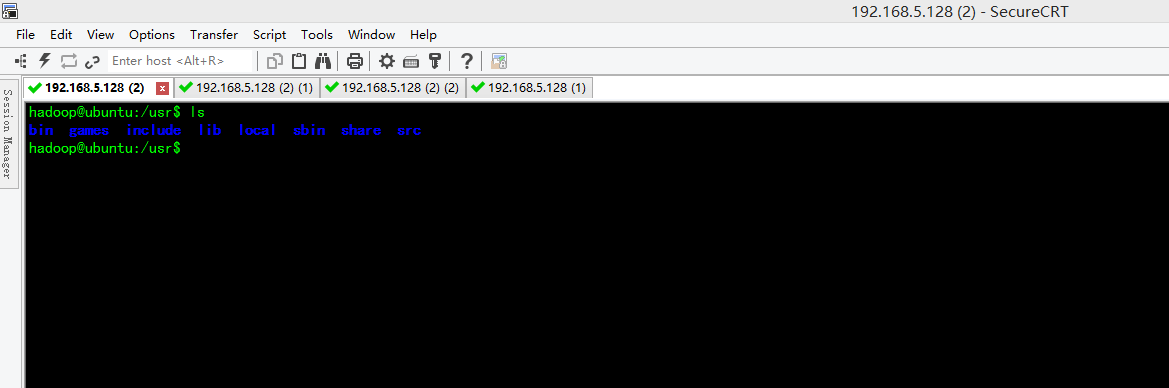
Ubuntu系统安装完后，可采用图形界面里面提供的终端，或者使用Vi、XShell等终端工具进行连接操作，如下图所示：

图形界面中打开终端可通过Ctrl + Alt + T快捷方式打开



Vi终端连接如下：

需要先通过ifconfig找到虚拟机内Ubuntu系统的IP地址。



**使用本教程请确保系统处于联网状态**

装好了Ubuntu系统之后，在安装Hadoop前还需要做一些必备工作

**创建Hadoop用户：sudo useradd -m hadoop -s /bin/bash**

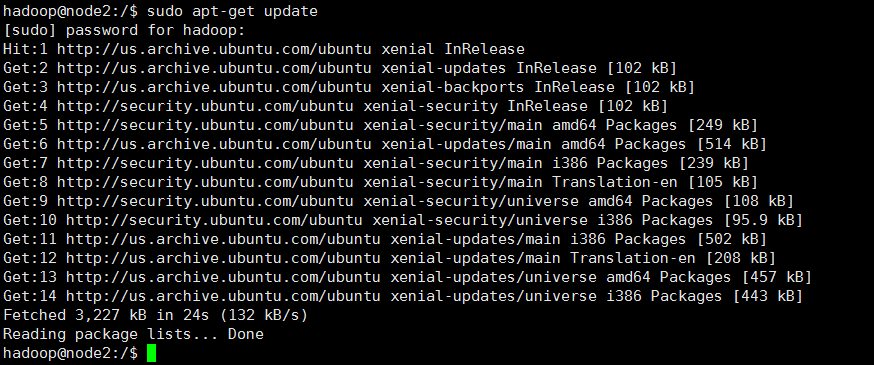
**修改Hadoop用户密码：sudo passwd hadoop，按提示输入两次密码**

**为Hadoop增加管理员权限：sudo adduser hadoop sudo**

【sudo命令：本文中会大量使用到sudo命令。sudo是linux系统管理指令，是允许系统管理员让普通用户执行一些或者全部的root命令的一个工具】

注销后登录Hadoop用户， 或者使用su Hadoop**切换到hadoop用户**

切换到hadoop用户后，先**更新一些apt**，后续会使用apt安装软件，如果没更新可能有一些软件安装不了。按ctrl+alt+t打开终端窗口或者使用xshell建立远程连接，执行如下命令：**sudo apt-get update**

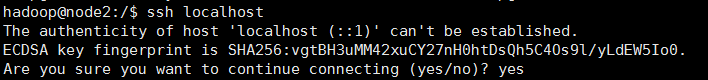


**安装SSH、配置SSH无密码登录**

**集群，单节点模式都需要用到SSH登录（类似于远程登录，你可以登录某台Linux主机，并且在上面运行命令），Ubuntu默认已安装了SSH client，此外还需要安装SSH server，使用命令：sudo apt-get install openssh-server**

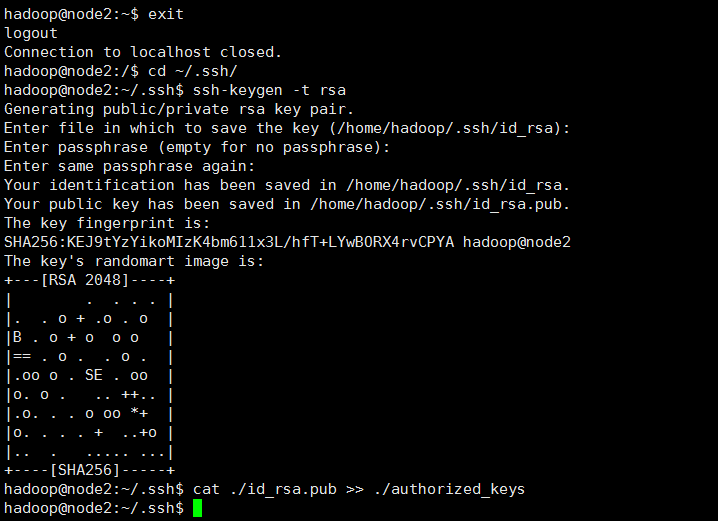
**安装后，可以使用命令登录本机：ssh localhost**

**此时会有如下提示（SSH首次登陆提示），输入yes。然后按提示输入密码hadoop，这样就可以登录到本机。**



**但这样登录是需要每次输入密码的，我们需要配置成SSH无密码登录比较方便。**

**首先推出刚才的ssh，就回到了原先的终端窗口，然后利用ssh-keygen生成密钥，并将密钥加入到授权中：**



**exit 退出刚才的ssh localhost**

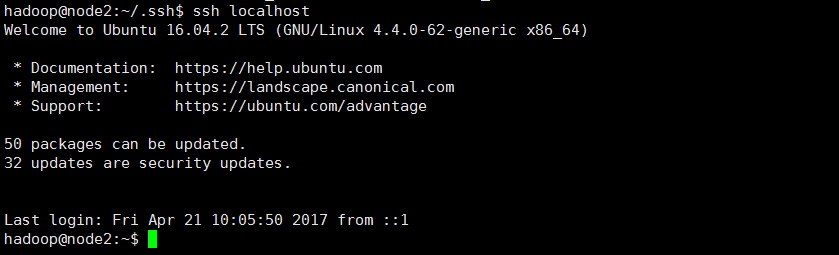
**cd ~/.ssh/ 若没有该目录，请执行一次ssh localhost**

**ssh-keygen -t rsa 会有提示，都按回车就可以**

**cat ./id\_rsa.pub >> ./authorized\_keys 加入授权**

**【注：~的含义   
在Linux中，~代表的是用户的主文件夹，即”/home/用户名”这个目录，如你的用户名为hadoop，则~就代表”/home/hadoop”。此外，命令中的#后面的文字是注释】**

**此时再用ssh localhost命令，无需输入密码就可以直接登录了，如下图所示**。



### 3.3安装 Java环境

Java环境可选择Oracle的JDK，或是OpenJDK，本教程安装的是Oracle的JDK。

1.检查自己的系统是32-bit还是64-bit

在命令行输入：

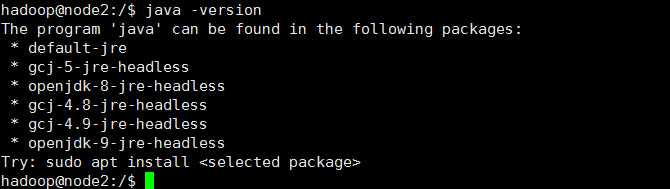
getconf LONG\_BIT

返回32或者是64，可以来查看操作系统的位数是32位还是64位

http://i.imgur.com/PwWzPid.png

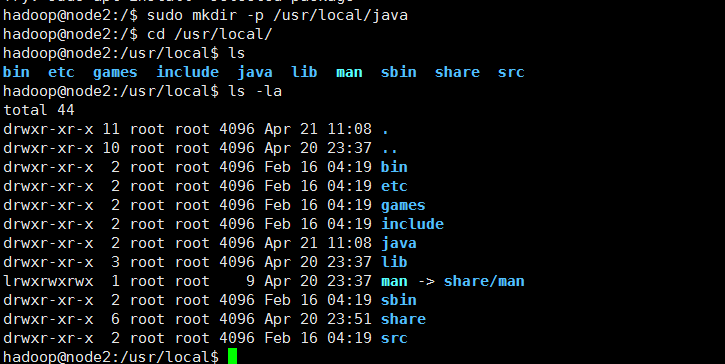
2.检查你的系统是否已经安装了java，输入命令：

java -version

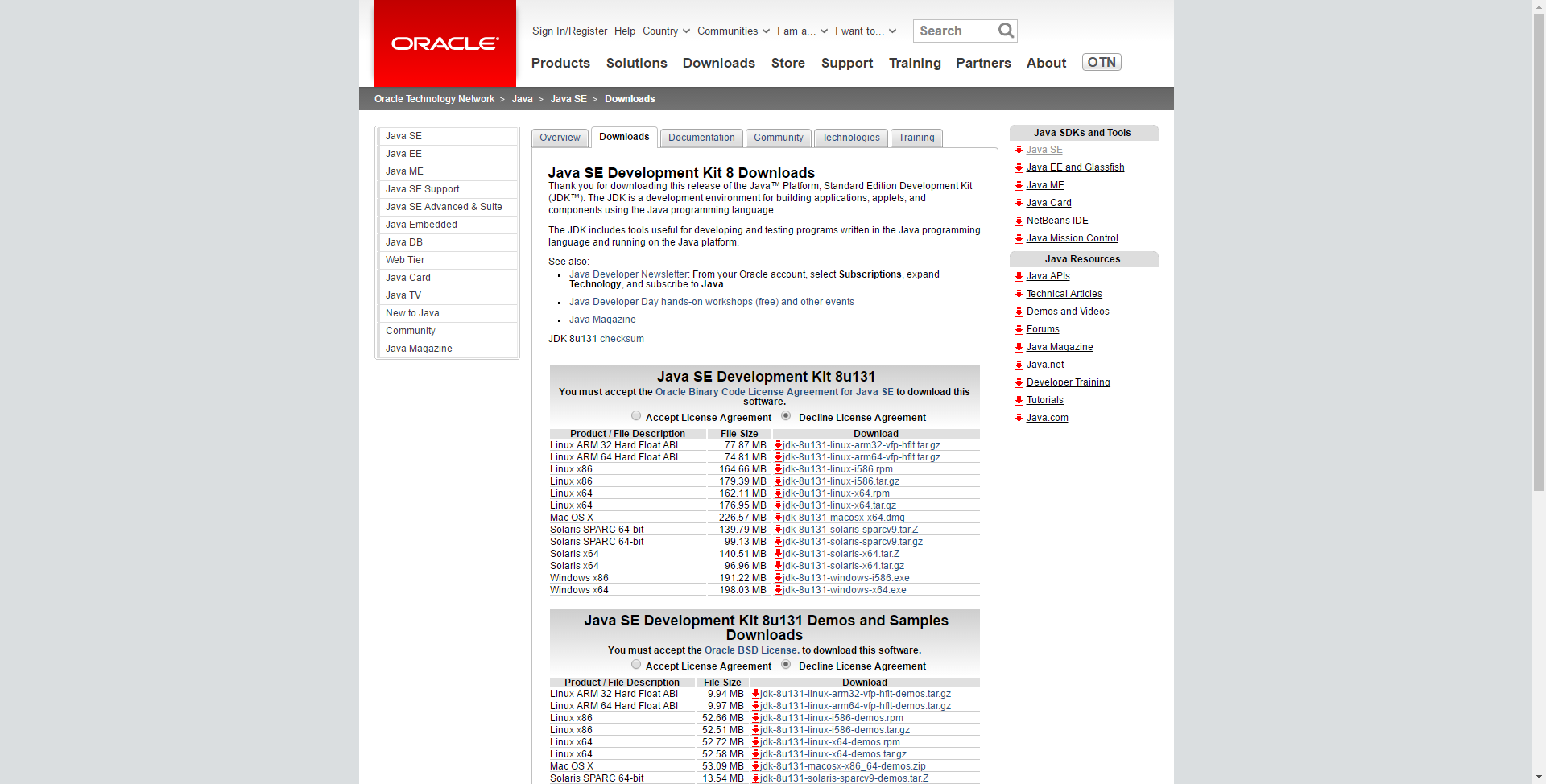


如果有安装OpenJDK，需要先卸载OpenJDK

3.建立Java目录，输入命令: sudo mkdir -p /usr/local/java



4.下载Linux版本的Orcle Java JDK，网址如下：[http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html （软件安装目录中有提供，本教程采用jdk-8u162-linux-i586.tar.gz](http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html%20%20（软件安装目录中有提供，本教程采用jdk-8u162-linux-i586.tar.gz)）



5.复制文件到刚刚创建的Java目录，路径为/usr/local/java;

sudo cp -r jdk-8u131-linux-x64.tar.gz /usr/local/java/

cd /usr/local/java

6.解压下载文件到此目录：

sudo tar xvzf jdk-8u131-Linux-x64.tar.gz

7.下面开始设置环境变量

sudo vi /etc/profile

复制下面代码到文件的末尾处，然后保存并关闭文件：

JAVA\_HOME=/usr/local/java/jdk1.8.0\_131

PATH=$PATH:$HOME/bin:$JAVA\_HOME/bin

export JAVA\_HOME

export PATH

8.配置你的Ubuntu Linux系统Oracle Java JDK的位置:

sudo update-alternatives --install "/usr/bin/java" "java" "/usr/local/java/jdk1.8/bin/java" 1

sudo update-alternatives --install "/usr/bin/javac" "javac" "/usr/local/java/jdk1.8/bin/javac" 1

sudo update-alternatives --install "/usr/bin/javaws" "javaws" "/usr/local/java/jdk1.8/bin/javaws" 1

9.将Orcle java JDK设置为默认:

sudo update-alternatives --set java /usr/local/java/jdk1.8/java

sudo update-alternatives --set javac /usr/local/java/jdk1.8/bin/javac

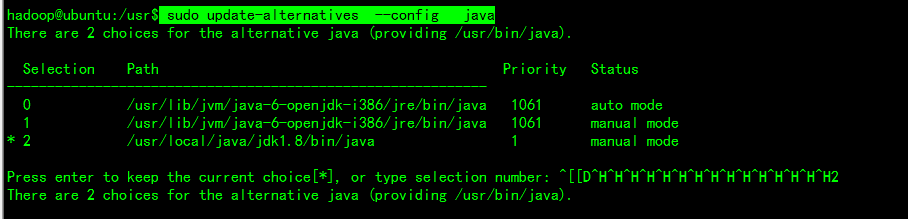
sudo update-alternatives --set javaws /usr/local/java/jdk1.8/bin/javaws

10.重新加载下/etc/profile中的路径:

source /etc/profile

11．删除系统自带的OpenJDK

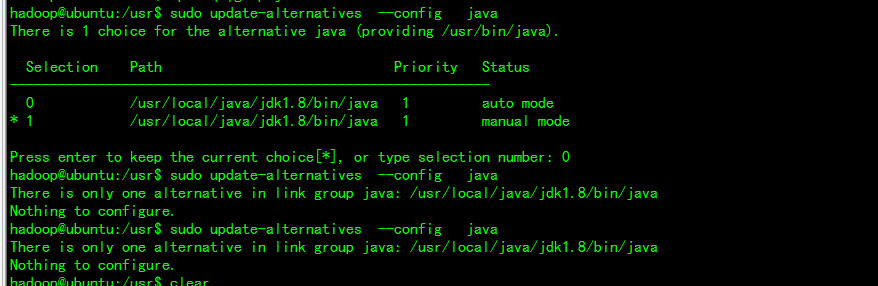
通过命令查看：sudo update-alternatives --config java



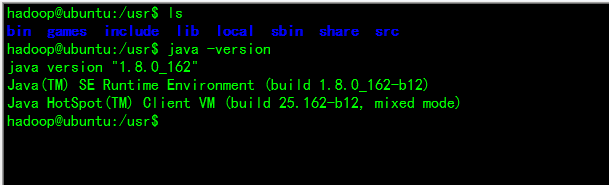
Jdk1.8是我们自己安装的

执行命令：**sudo apt-get remove openjdk\*** 删除有关openjdk的东西

然后执行：sudo update-alternatives --config java重新查看就只剩下自己安装的jdk了



12．输入命令查看是否安装成功：java -version



### 3.4安装Hadoop2

Hadoop2 可以通过<http://mirrors.cnnic.cn/apache/hadoop/common/> 下载，一般选择下载最新的稳定版本，即下载”stable”下的hadoop-2.x.y.tar.gz这个格式的文件，这是编译好的，另一个包含src的则是Hadoop源代码，需要进行编译才可使用

下载完hadoop文件后一般可以直接使用。

我们选择将Hadoop安装至/usr/local/中：

sudo tar -zxf ~/Downloads/hadoop-2.9.0.tar.gz -C /usr/local #解压到/usr/local中

cd /usr/local/

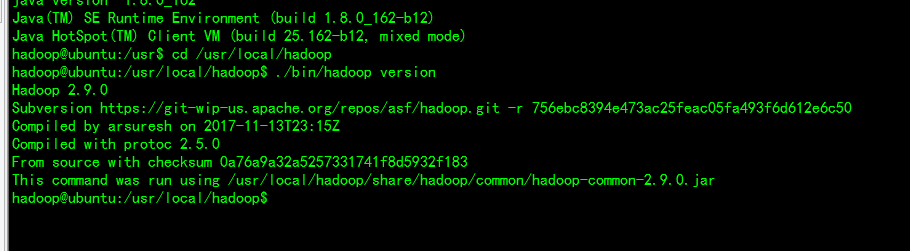
sudo mv ./hadoop-2.9.0/ ./hadoop #将文件夹名改为hadoop

sudo chown -R hadoop ./hadoop #修改文件权限

Hadoop解压后即可使用。输入如下命令来检查Hadoop是否可用，成功则会显示Hadoop版本信息：

cd /usr/local/hadoop

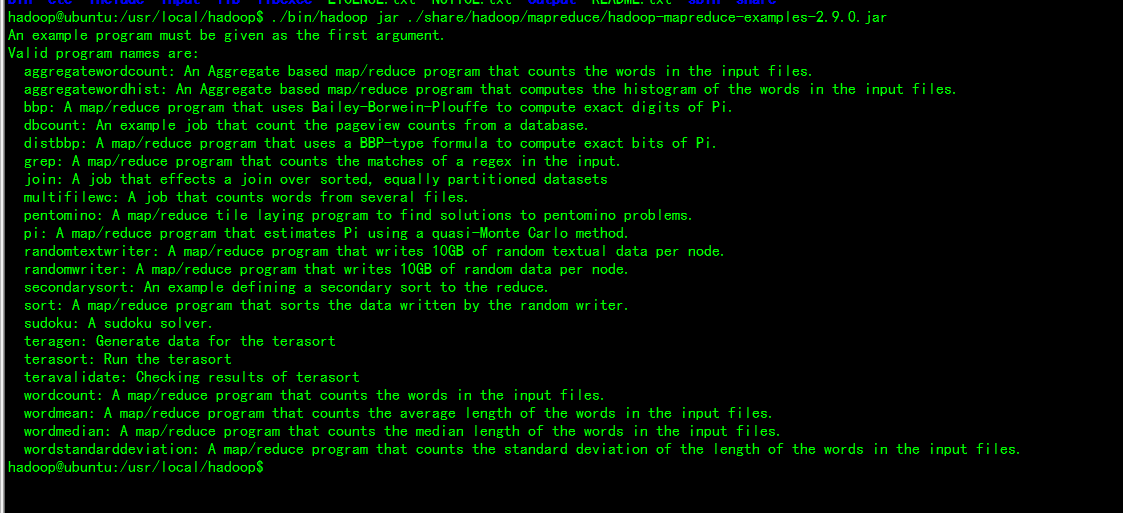
./bin/hadoop version



#### Hadoop单击配置（非分布式）

Hadoop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。非分布式即单java进程，方便进行调试。

现在可以执行例子来感受下Hadoop的运行。Hadoop附带了丰富的例子（运行命令：./bin/hadoop jar ./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.9.0.jar 可以看到所有的例子），包括wordcount、terasort、join、grep等。



在此我们选择运行grep例子，将input文件夹中的所有文件作为输入，筛选当中符合正则表达式dfs[a-z.]+的单词并统计出现的次数，最后输出结果到output文件夹中。

cd /usr/local/hadoop

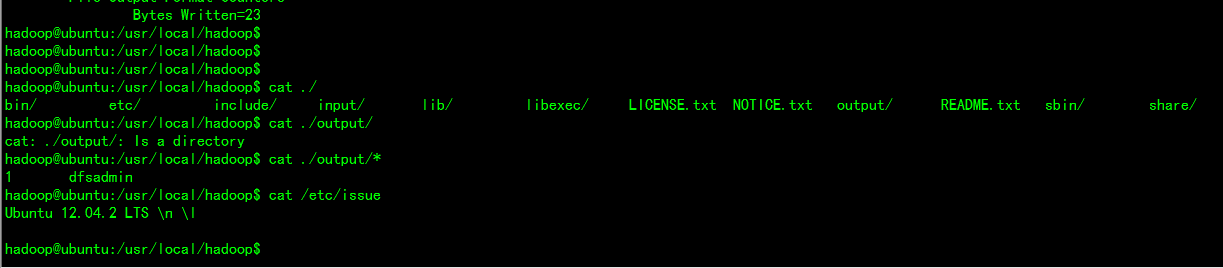
mkdir ./input

cp ./etc/hadoop/\*.xml ./input #将配置文件作为输入文件

./bin/hadoop jar ./share/hadoop/mapreduce/hadoop-mapreduce-examples-2.9.0.jar grep ./input/ ./output 'dfs[a-z.]+'

cat ./output/\*

执行成功后如下所示，输出了作业的相关信息，输出的结果是符合正则的单词dfsadmin出现了1次



【注意，Hadoop默认不会覆盖结果文件，因此再次运行上面实例会提示出错，需要先将./output删除：rm -r ./output】

补充，如果运行Hadoop的时候发现找不到jdk，可以直接将jdk的路径放置在hadoop.env.sh里面，具体如下：

　　　　export JAVA\_HOME="/usr/local/java/jdk1.8"

#### Hadoop伪分布式配置

Hadoop可以在单节点上以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，读取的是HDFS中的文件。

Hadoop的配置文件位于/usr/local/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的配置文件是xml格式，每个配置以声明property的name和value的方式来实现。

修改配置文件core-site.xml（使用命令 vi ./etc/hadoop/core-site.xml）,修改下面的，增加<Configuration>里面的内容：

<property>

<name>hadoop.tmp.dir</name>

<value>/usr/local/hadoop/tmp</value>

<description>Abase for other temporary directories .</description>

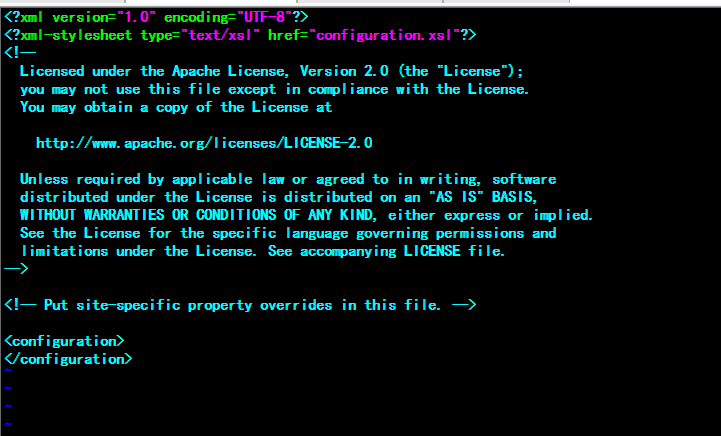
</property>

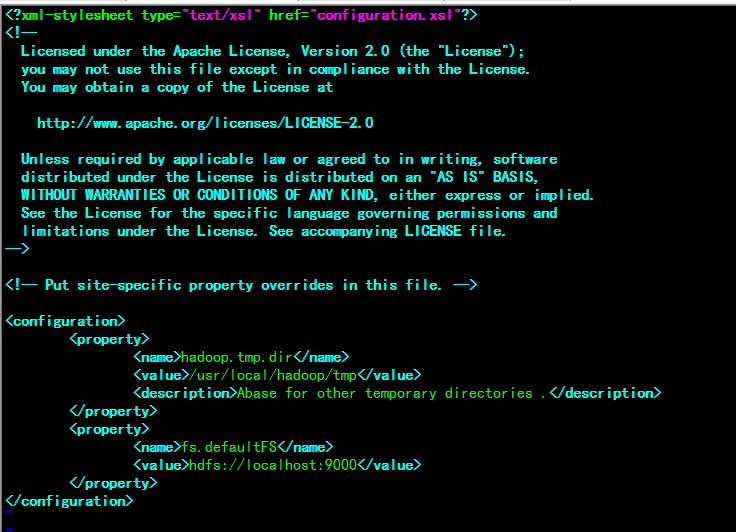
<property>

<name>fs.defaultFS</name>

<value>hdfs://localhost:9000</value>

</property>





同样的，修改配置文件hdfs-site.xml

<property>

<name>dfs.replication</name>

<value>1</value>

</property>

<property>

<name>dfs.namenode.name.dir</name>

<value>/usr/local/hadoop/tmp/dfs/name</value>

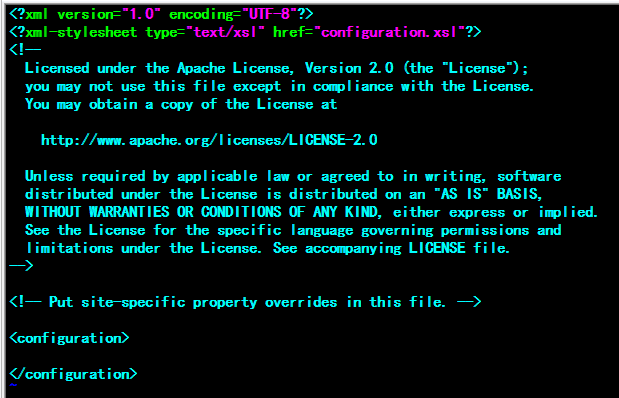
</property>

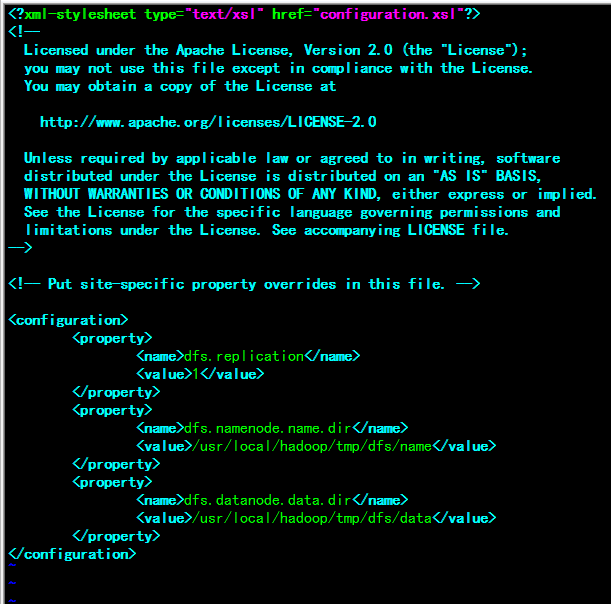
<property>

<name>dfs.datanode.data.dir</name>

<value>/usr/local/hadoop/tmp/dfs/data</value>

</property>

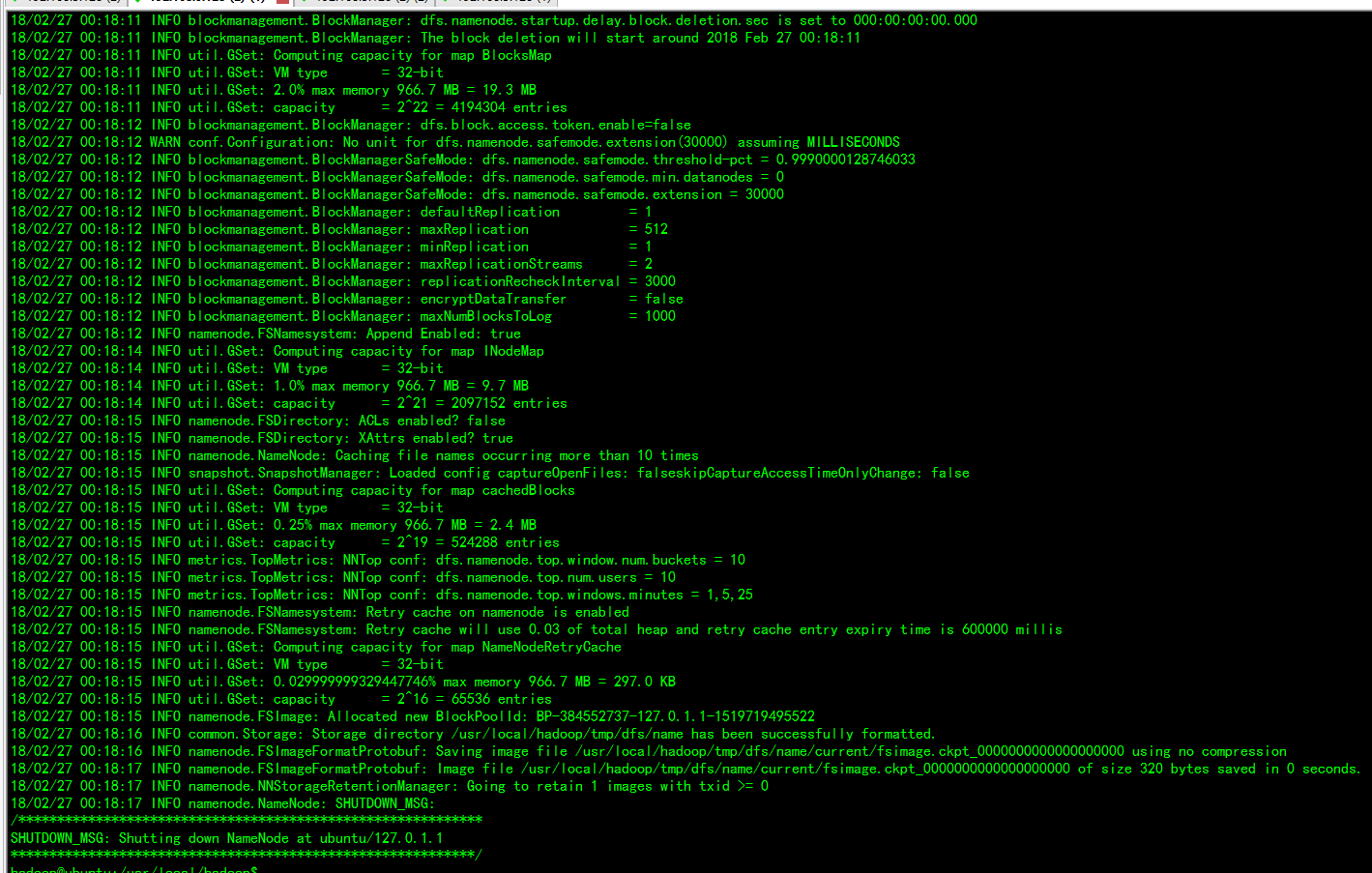




**Hadoop配置文件说明   
Hadoop的运行方式是由配置文件决定的（运行Hadoop时会读取配置文件），因此如果需要从伪分布模式切换到非分布式模式，需要删除core-site.xml中的配置项。   
此外，伪分布式虽然只需要配置fs.defaultFS和dfs.replication就可以运行（官方教程如此)，不过若没有配置hadoop.tmp.dir参数，则默认使用的临时目录为/tmp/hadoop-hadoop，而这个目录在重启时有可能被系统清理掉，导致必须重新执行format才行。所以进行了设置，同时也指定dfs.namenode.name.dir和dfs.datanode.data.dir，否则在接下来的步骤中可能会处错。**

配置完成后，执行NameNode的格式化，命令如下：

sudo ./bin/hdfs namenode -format



接着开启NameNode和DataNode守护进程。

Sudo ./sbin/start-dfs.sh #start-dfs.sh是个完整的执行文件，中间没有空格键