# 实验七 开发一个YARN客户端应用

**·实验目标：**

通过该实验后，能开发简单的yarn客户端应用程序，并了解其执行过程。

**·实验概述：**

开发一个yarn客户端应用程序，实现通过客户端发送请求给ResourceManager启动一个ApplicationMaster。

追踪yarn客户端应用程序执行的过程。

**·实验环境：**

三台虚拟机搭建的完全分布，已配置好的hadoop集群环境，server-01为namenode，serveer-02与server-03为DataNode。

**·实验背景：**

自己从头开始开发一个yarn客户端应用程序是复杂的，并且往往没有必要的。Yarn源码中已经自带了两个Application应用程序。尽管这两个示例程序非常简单，但是他们已经具有了一个应用程序具备的所有功能，其他任何应用程序均可在这两个程序基础上扩展而来。

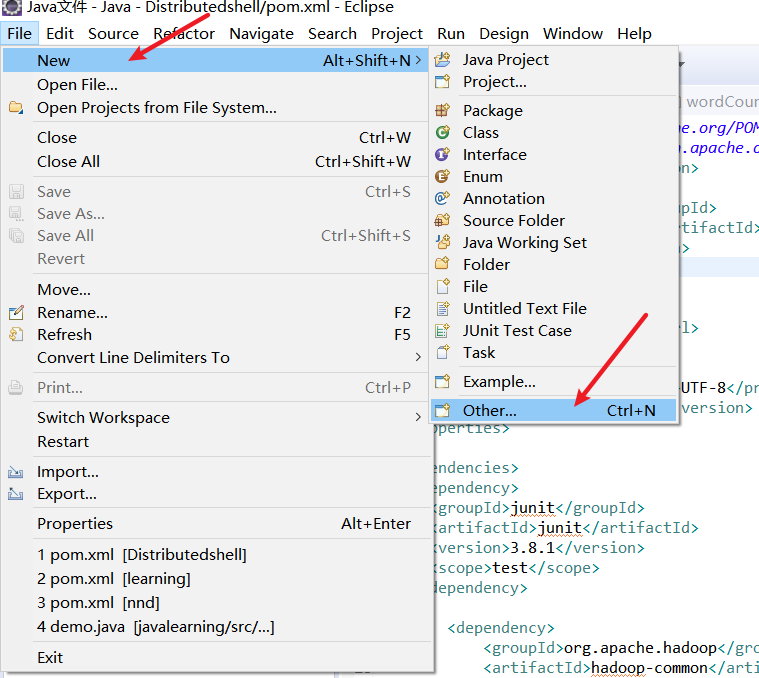
**·实验步骤：**

1、创建Maven项目

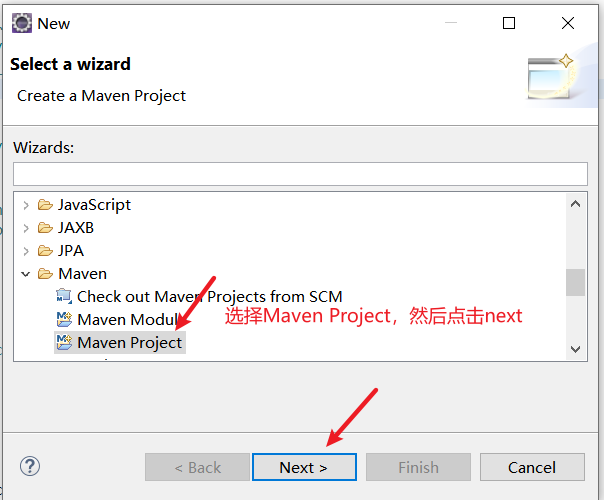
1.1、使用eclipse新建名称为Distributedshell的Maven项目。

打开eclipse工具，新建Distributedshell的Maven项目。

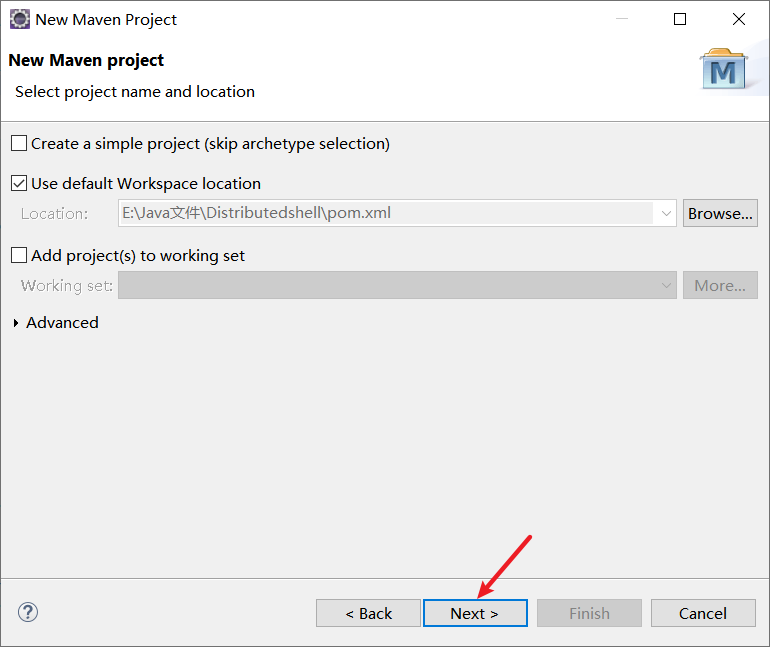
如图：

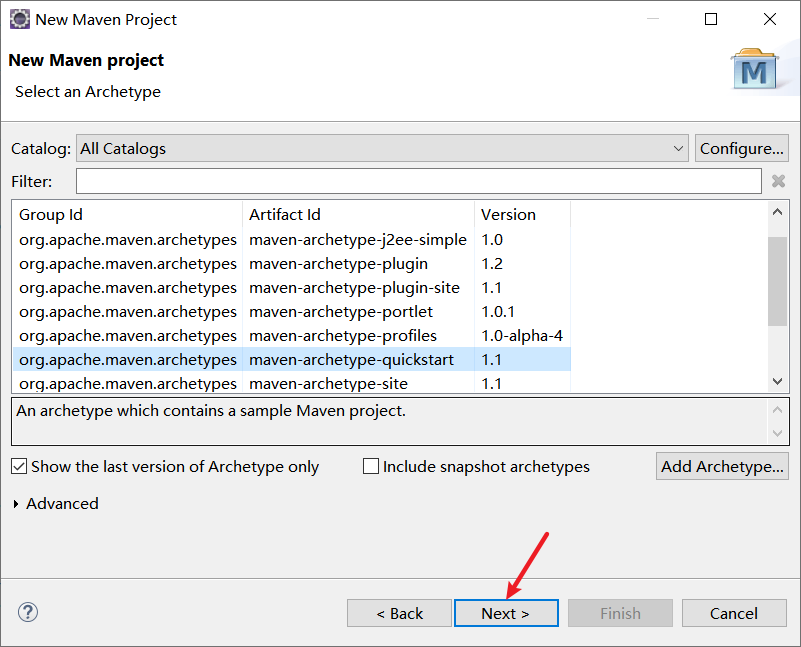


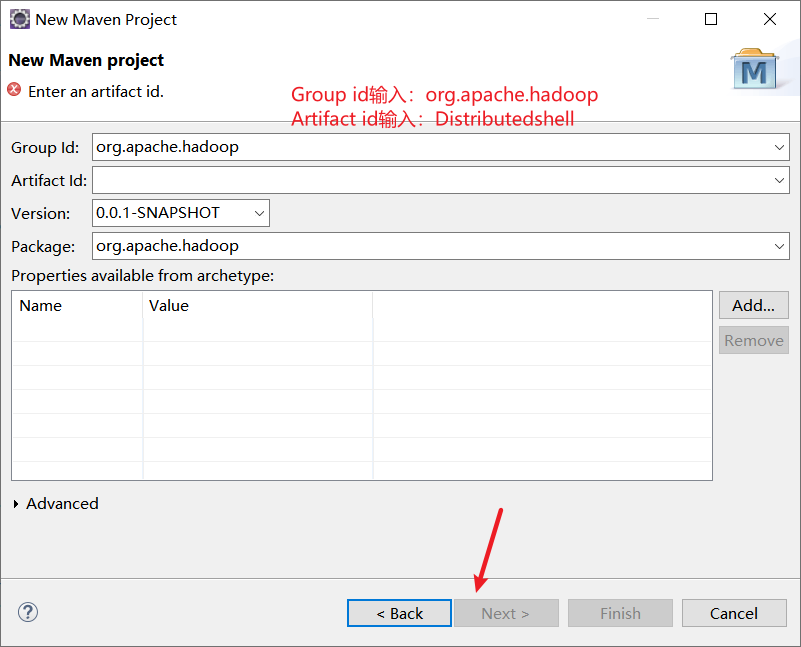
进入如下窗口，点击next按钮：



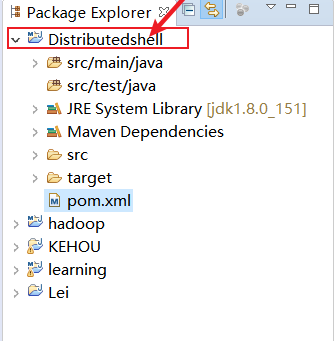
点击next：



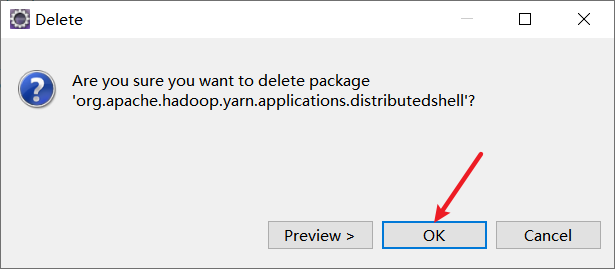
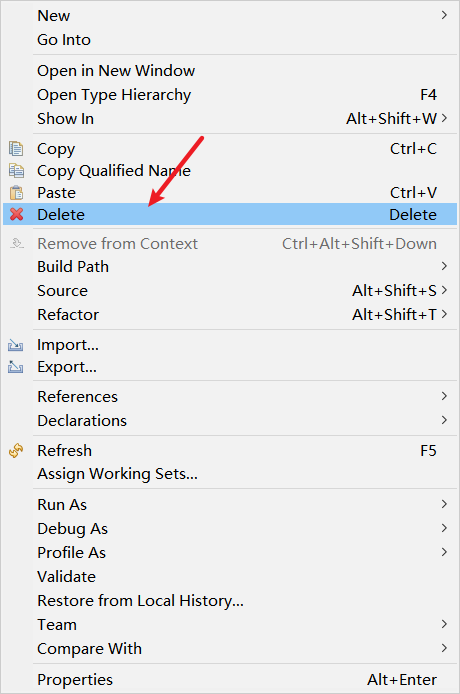




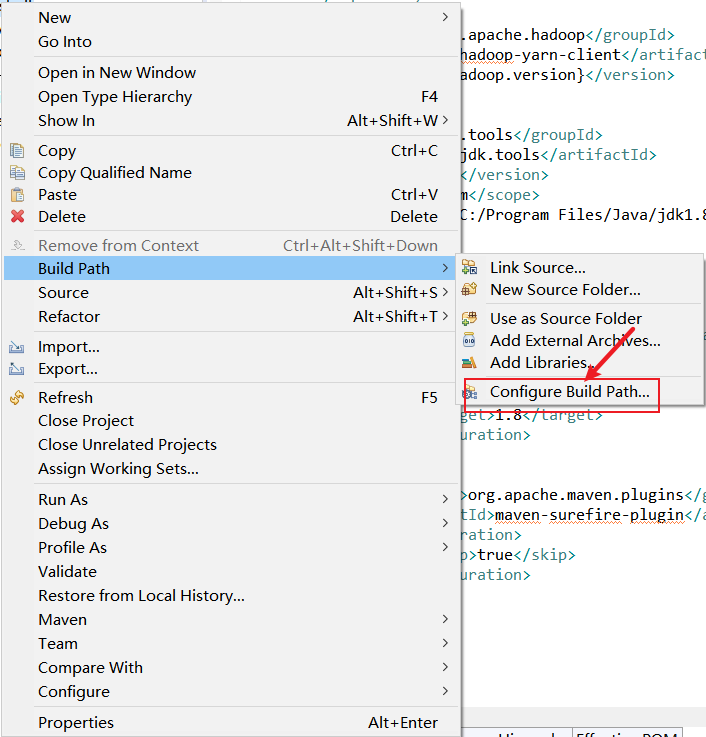
上图点击finish，项目建好，如图：

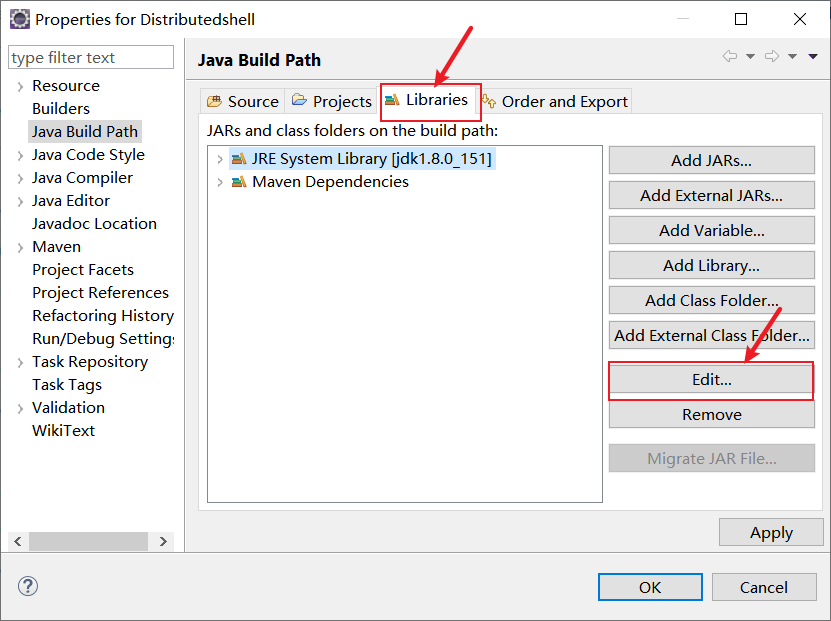


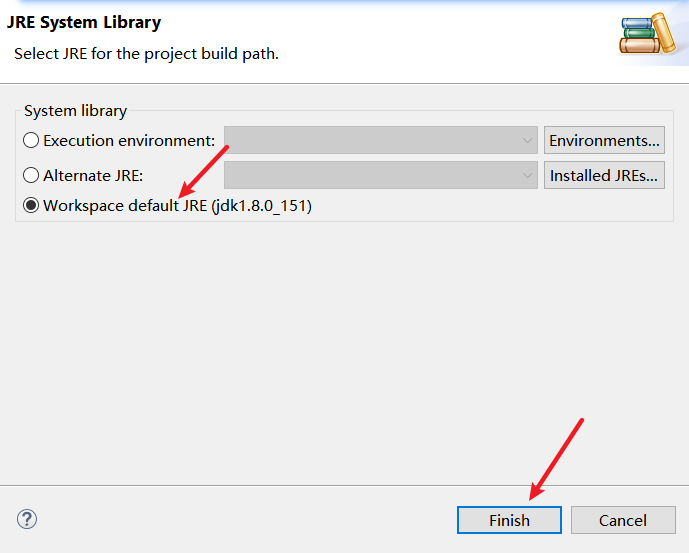
将src/main/java和src/test/java目录下自动创建的package和其包里的内容全部删掉（选择delete，然后“OK”）：



1.2、更改jdk版本为1.8

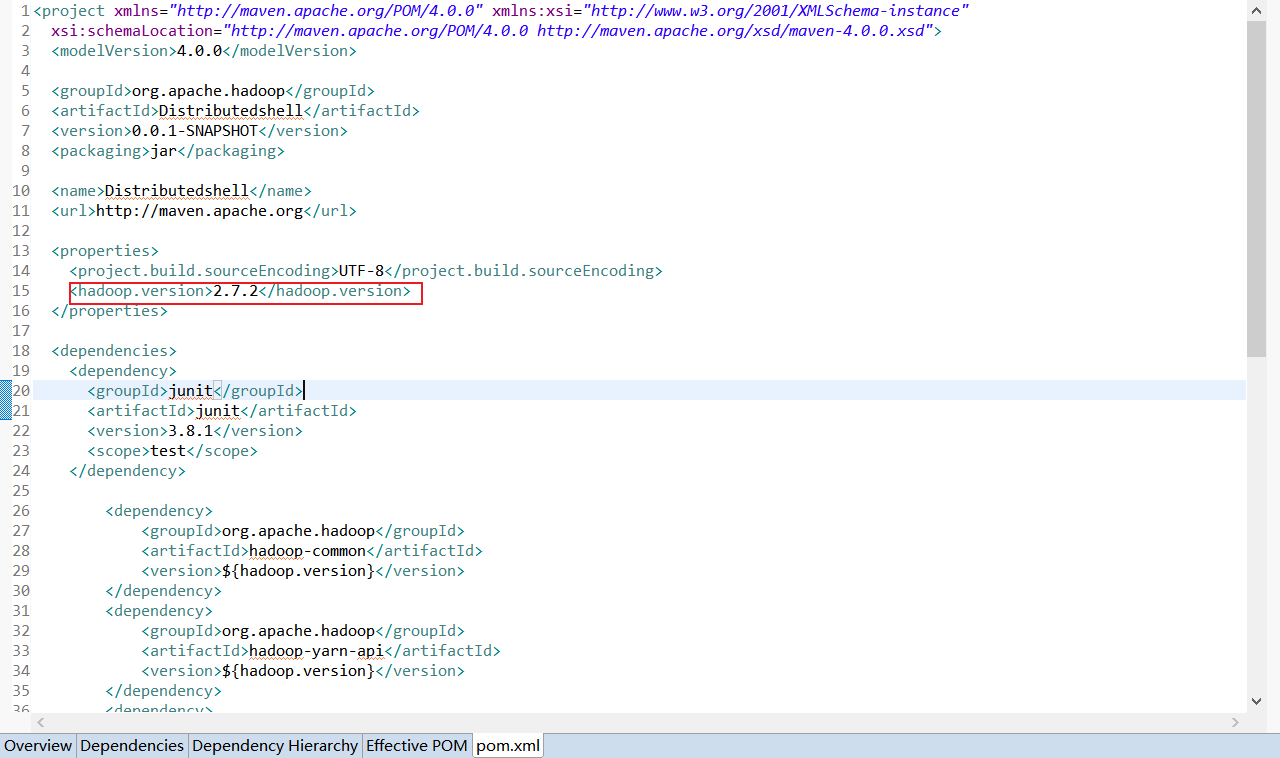


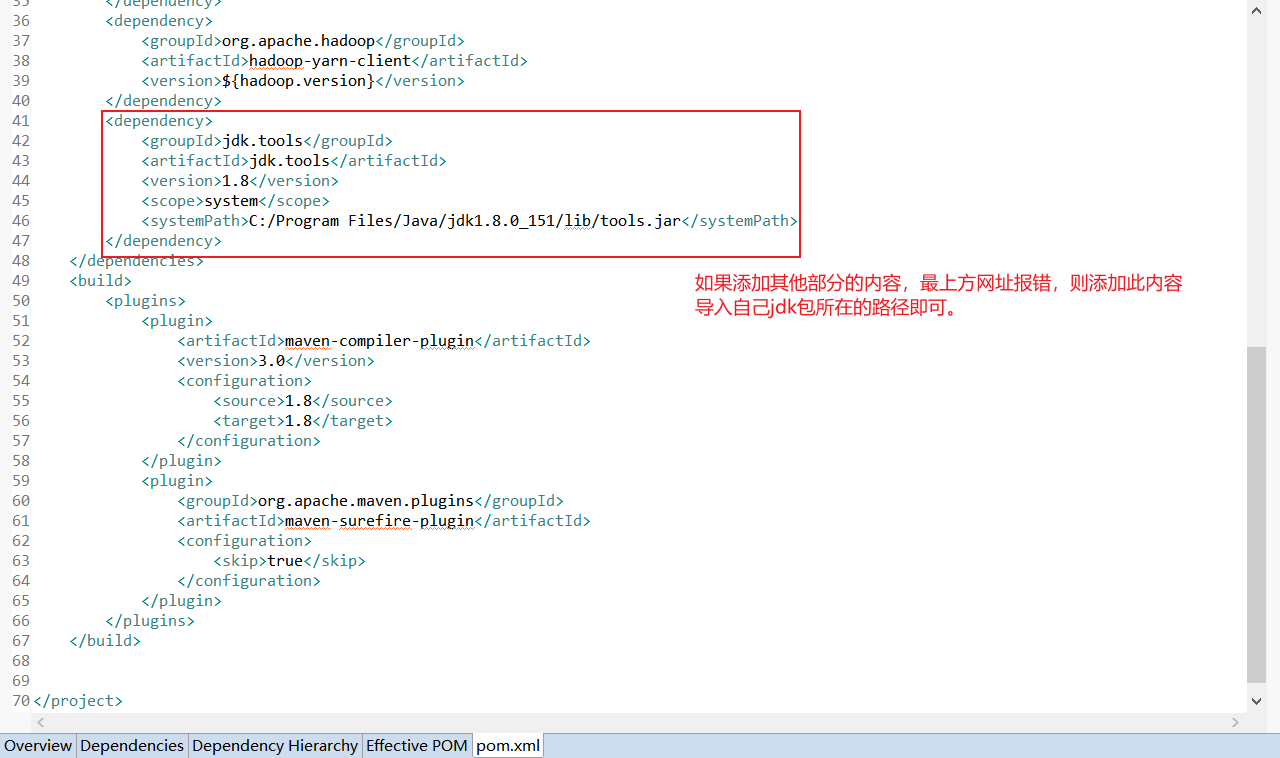




1. 编辑pom.xml 文件

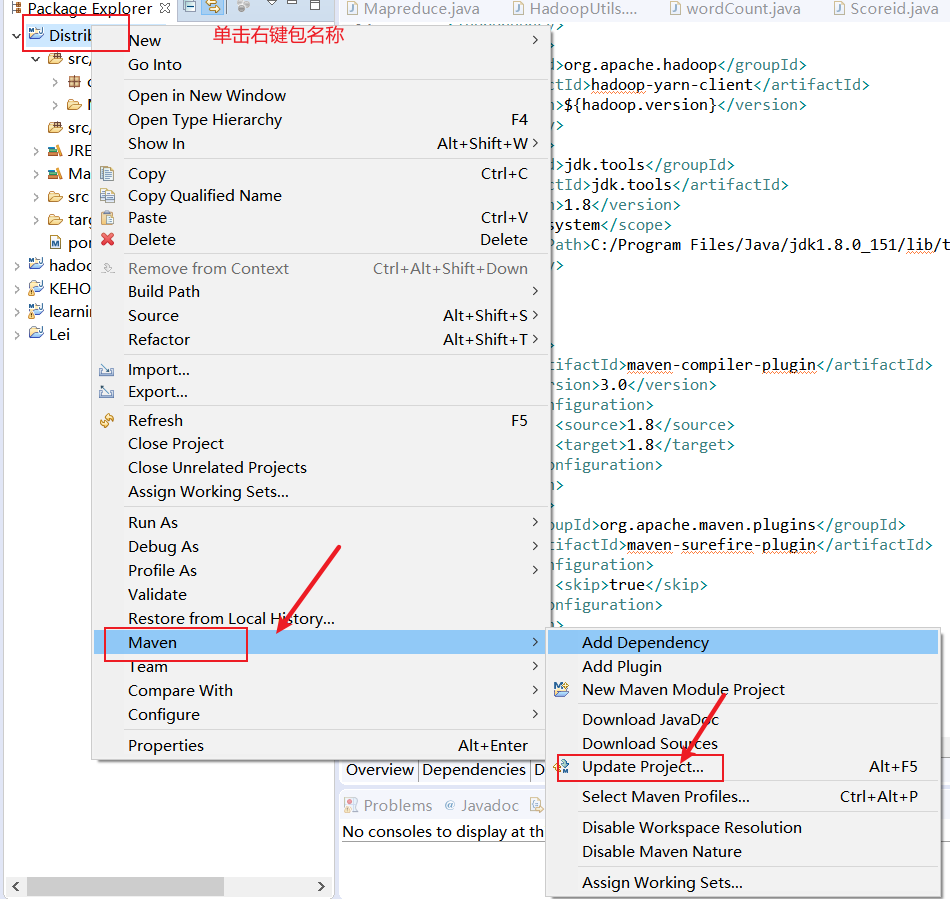
文件内容如下（注意编译的jdk版本与自己本地情况相匹配）：





保存后会自动下载依赖jar包。

此时项目上可能会有一个红叉，在项目上点击右键，选择Maven🡪Update Project...

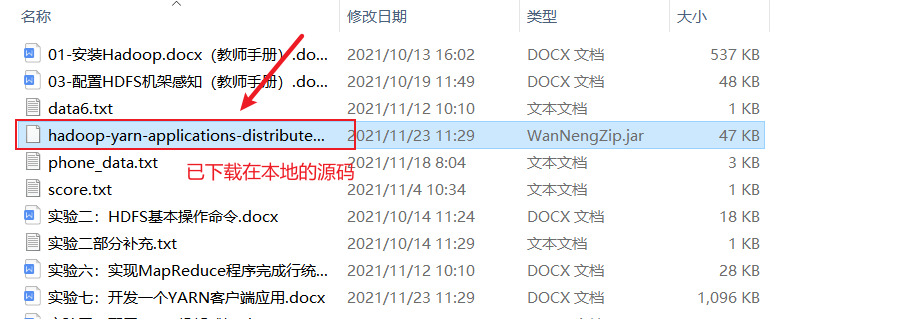


然后在弹出的窗口点击“OK”。

1. 下载源码：

从下方网址下载到本地任意路径：

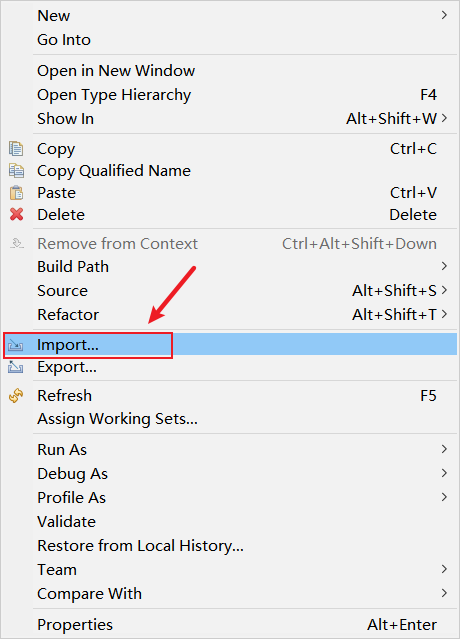
https://repo1.maven.org/maven2/org/apache/hadoop/hadoop-yarn-applications-distributedshell/2.7.3/hadoop-yarn-applications-distributedshell-2.7.3-sources.jar



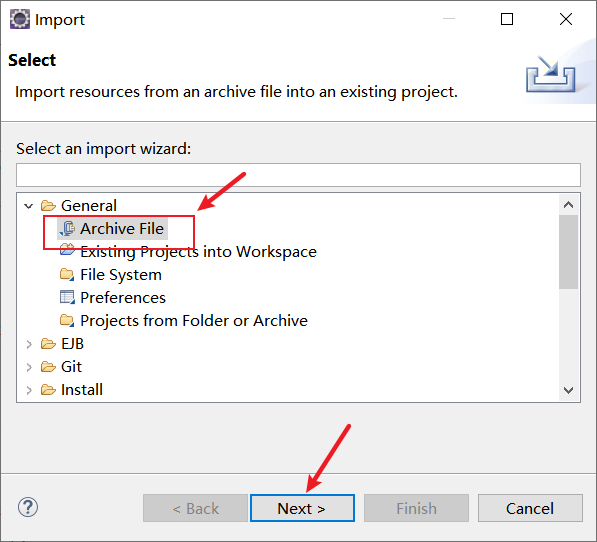
1. 导入源码：

将上一步下载的源码导入到本次项目中：

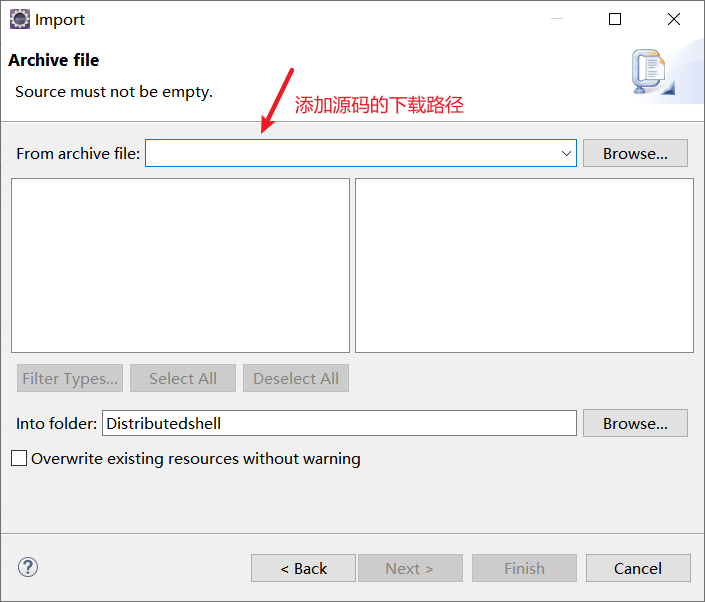
在eclipse工具里，src/main.java上点击右键，选择“import”:



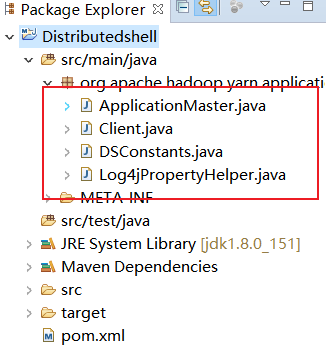
点击“Archive File”:



点击“Next"按钮，进入到如下界面，From archive file:选择上一步我们下载的源码文件，如下图：



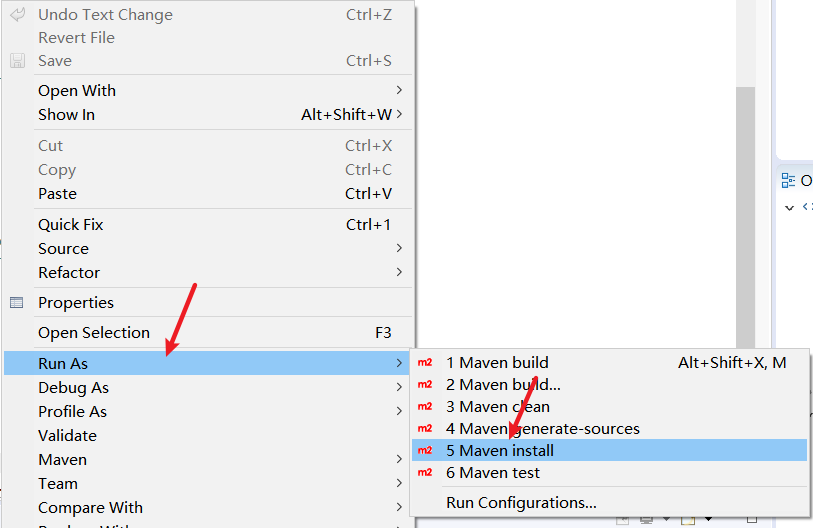
添加路径后，选择“Finish”:



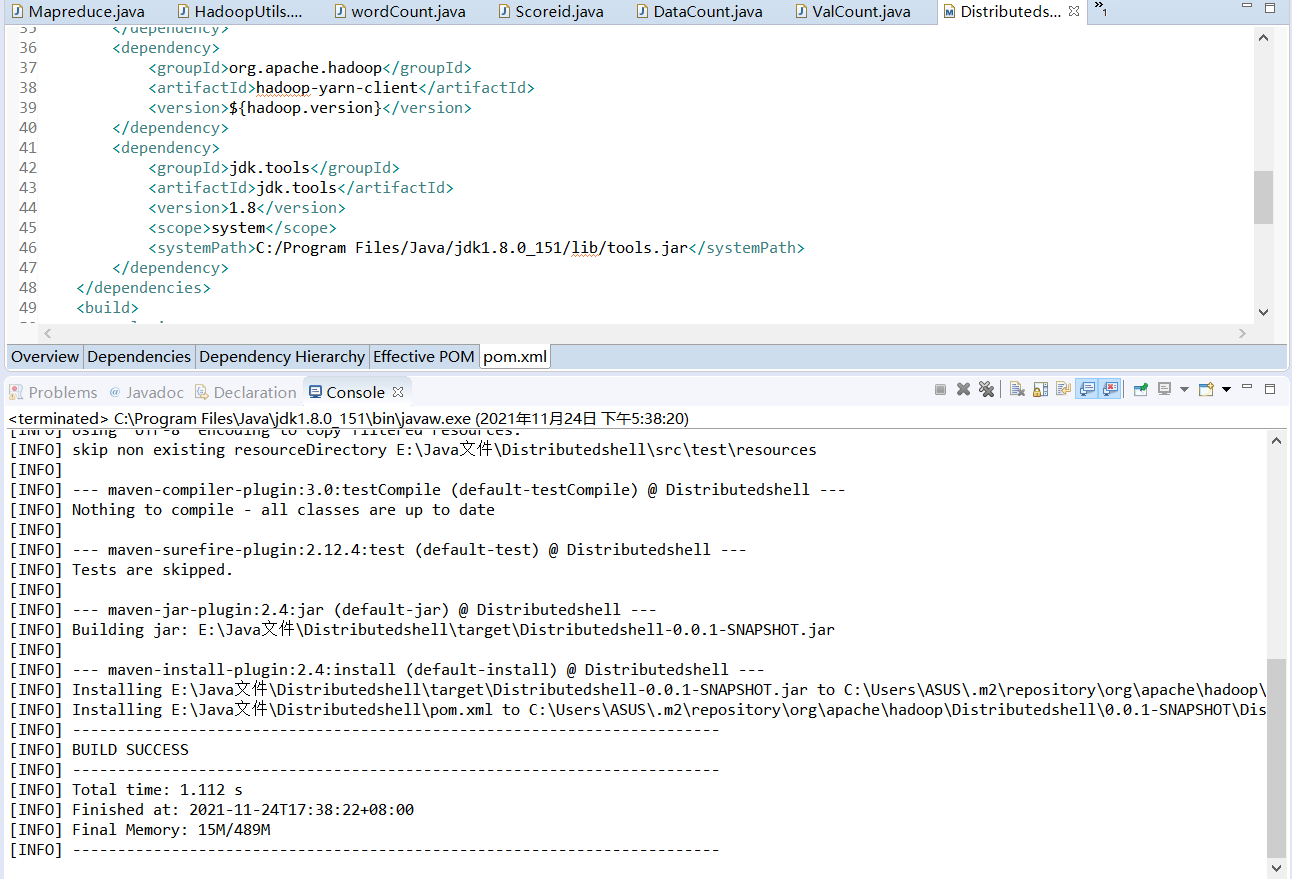
1. 生成jar包：

将上一步中的项目打成jar包命名为：Distributedshell-2.7.3.jar

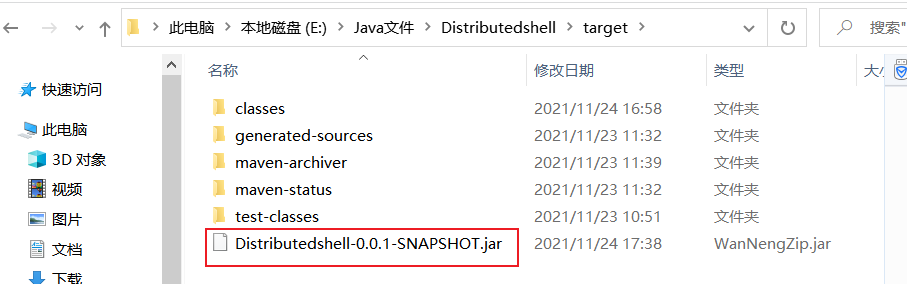
在pom.xml文件上点击右键，选择Run As Maven install：



如图所示运行成功：



如图所示已生成jar包：

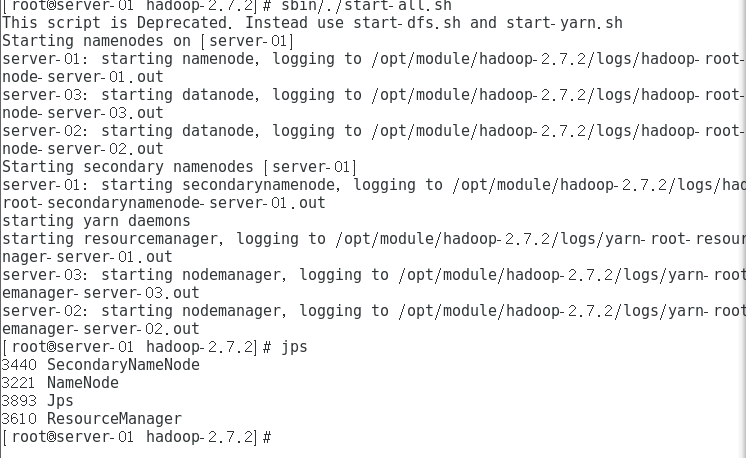


将其jar包改名：



1. 启动hadoop服务

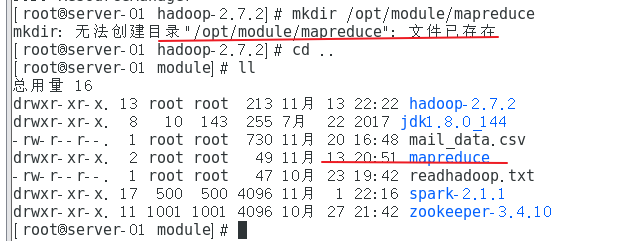




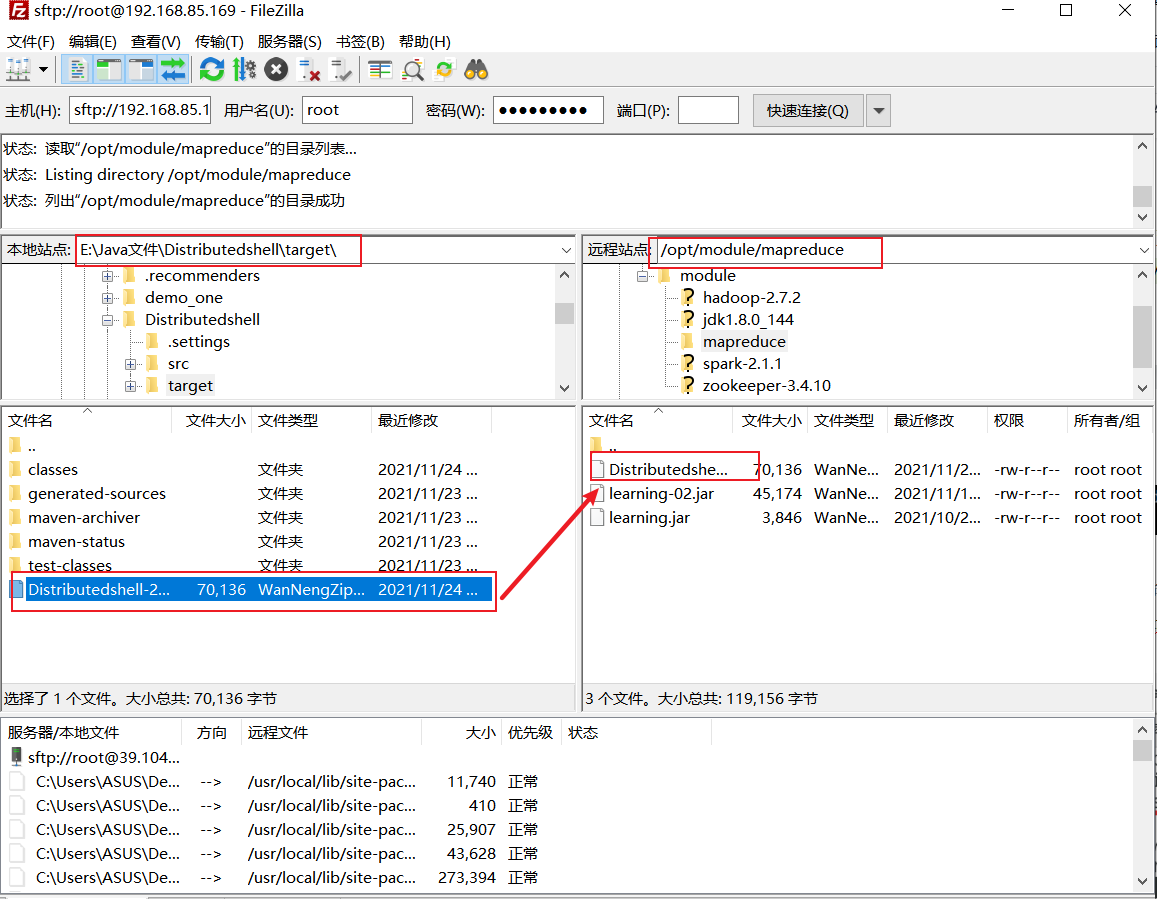
1. 运行yarn客户端应用程序：

将上之前导出的Distributedshell-2.7.3.jar包使用scp命令上传到任意一台服务器上，然后执行hadoop jar命令，在三个容器里分别执行date命令。并查看每个容器运行的结果。

在/opt/module下查看文件夹：

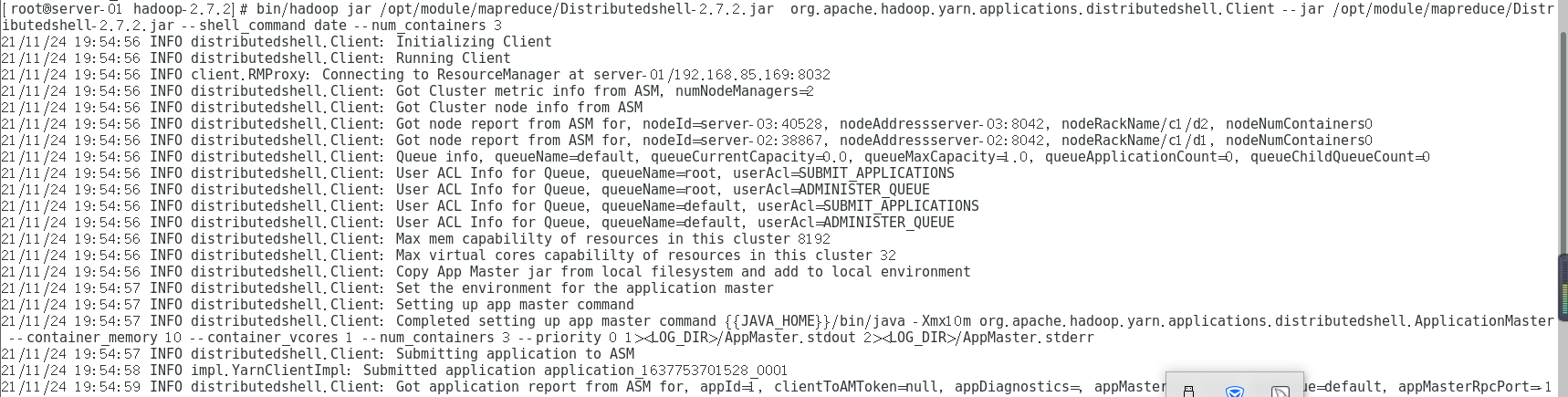


将上一步中生成的jar包重命名为Distributedshell-2.7.3.jar，并使用scp命令上传到主服务器上我们刚建的/usr/local/zhitu/mapreduce 目录下：



运行yarn客户端应用程序:使用ssh工具登陆到**主服务器**，进入到hadoop-2.7.3目录下，使用命令：

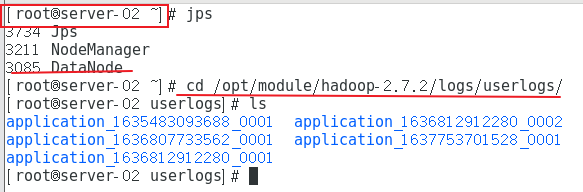
|  |
| --- |
| bin/hadoop jar /usr/local/zhitu/mapreduce/Distributedshell-2.7.3.jar  org.apache.hadoop.yarn.applications.distributedshell.Client  --jar /usr/local/zhitu/mapreduce/Distributedshell-2.7.3.jar  --shell\_command date  --num\_containers 3 |

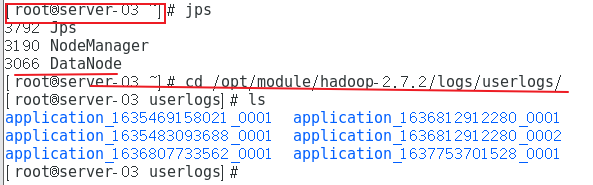


查看运行结果：

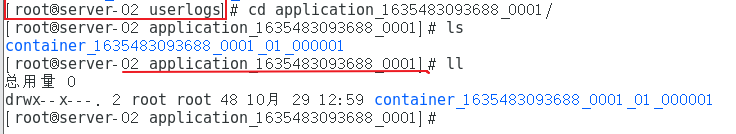
我们请求执行的shell脚本命令date 是被分布到datanode节点上执行的，所以查看结果可以到重服务器上查看**(具体在哪台服务器上执行具有随机性)。**

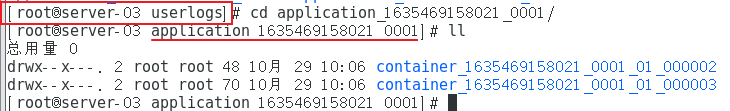
进入到任意一台从服务器上，使用cd命令进入到目录：/opt/module/hadoop-2.7.2/logs/userlogs下面，再使用ls命令查看该目录下的文件夹，如下图：



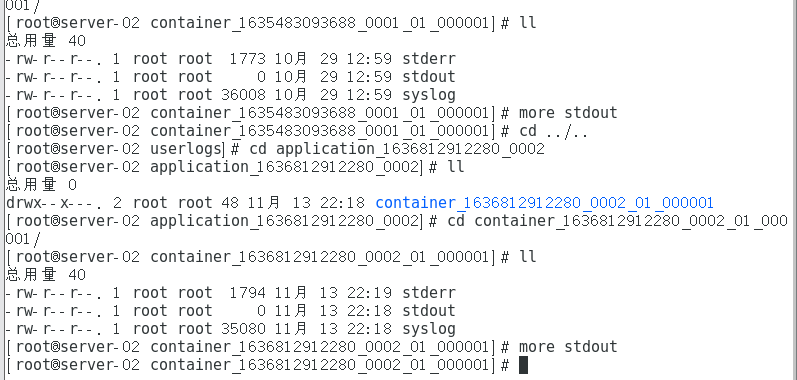


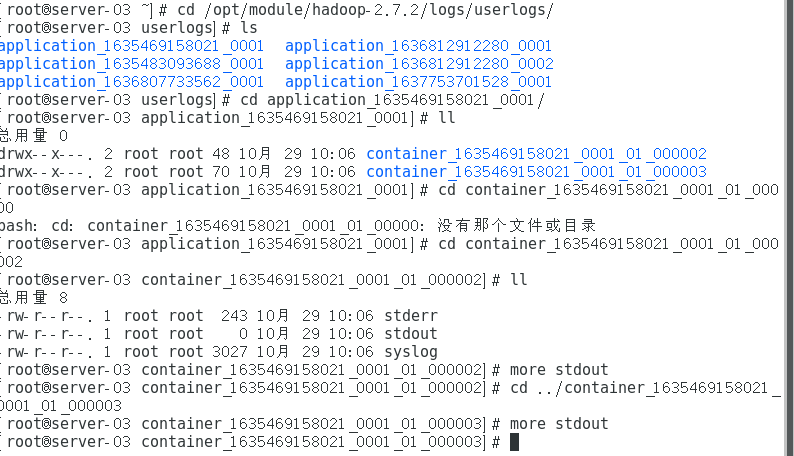
可以看到生成了一个文件夹，使用cd命令进入到该文件夹下，在使用ll命令查看该文件夹下的内容，如下图：





可以看到有三个文件夹，这三个文件夹是三个容器的运行日志。我们分别进入到每个文件夹下，查看文件夹下的stuout文件内容，如下图：





可以看到我们的data命令3个容器均在该从服务器上运行的。

**·实验心得：**

通过本次实验，学会了如何在eclipse中的pom.xml文件添加jdk地址，如何查看、处理相关运行错误，如何在pom.xml文件中打出jar包，进一步了解了yarn服务的运行。