使用Eclipse变异运行MapReduce程序

# 环境

Ubuntu 16.04

Hadoop 2.6.0(伪分布式)

jdk1.7+eclipse luna

# 安装Eclipse

在官网上下载eclipse，选择合适的版本，并解压

sudo tar -zvxf ~/Downloads/eclipse-java-\*.tar.gz -C /usr/lib

## 在桌面设置eclipse快捷方式

1.在桌面目录创建一个名为"eclipse.desktop"的文件

并添加以下内容：

[Desktop Entry]   
Encoding=UTF-8   
Name=Eclipse Platfrom   
Comment=Eclipse [IDE](https://www.baidu.com/s?wd=IDE&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4ujwWnHmvny7BuywBuWNW0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHm4njb4rjfvPWRkPHmknH6vr0)   
Exec=/usr/local/eclipse/eclipse #你的eclipse安装目录   
[Icon](https://www.baidu.com/s?wd=Icon&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4ujwWnHmvny7BuywBuWNW0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHm4njb4rjfvPWRkPHmknH6vr0)=/usr/local/eclipse/[icon](https://www.baidu.com/s?wd=icon&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4ujwWnHmvny7BuywBuWNW0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnHm4njb4rjfvPWRkPHmknH6vr0).xpm #eclipse下的图标目录  
Terminal=false   
StartupNotify=true   
Type=Application   
Categories=Application;Development;

2.然后给该文件赋权 ,运行“chmod u+x eclipse.desktop“命令即可。

# 安装Hadoop-Eclipse-Plugin

下载地址：[hadoop2x-eclipse-plugin](https://github.com/winghc/hadoop2x-eclipse-plugin)

下载后，将 release 中的 hadoop-eclipse-kepler-plugin-2.6.0.jar （还提供了 2.2.0 和 2.4.1 版本）复制到 Eclipse 安装目录的 plugins 文件夹中，运行 eclipse -clean 重启 Eclipse 即可（添加插件后只需要运行一次该命令，以后按照正常方式启动就行了）。

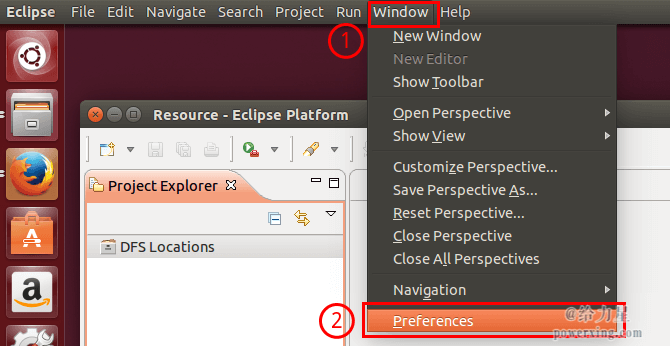
1. unzip -qo ~/下载/hadoop2x-eclipse-plugin-master.zip -d ~/下载 *# 解压到 ~/下载 中*
2. sudo cp ~/下载/hadoop2x-eclipse-plugin-master/release/hadoop-eclipse-plugin-2.6.0.jar /usr/lib/eclipse/plugins/ *# 复制到 eclipse 安装目录的 plugins 目录下*
3. /usr/lib/eclipse/eclipse -clean *# 添加插件后需要用这种方式使插件生效*

# 配置Hadoop-Eclipse-Plugin

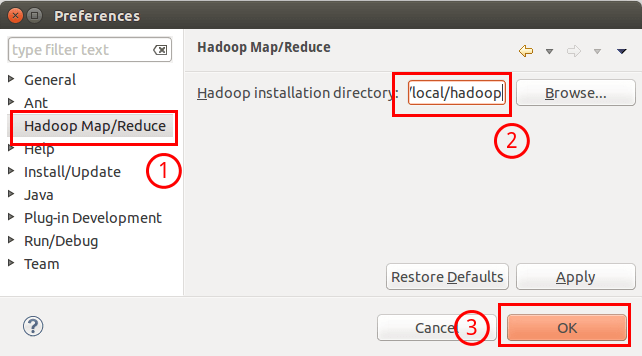
配置前需要确保开启了Hadoop。启动 Eclipse 后就可以在左侧的Project Explorer中看到 DFS Locations。

对插件进行进一步的配置：

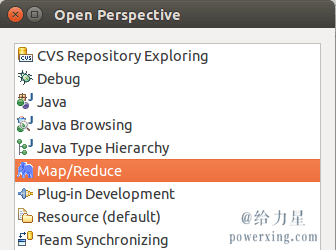
第一步：选择 Window 菜单下的 Preference。

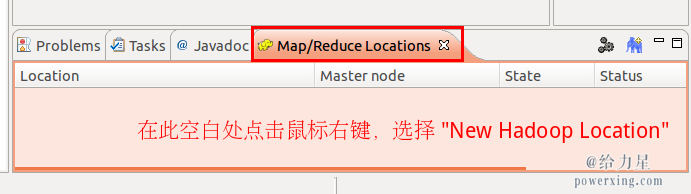


此时会弹出一个窗体，窗体的左侧会多出 Hadoop Map/Reduce 选项，点击此选项，选择 Hadoop 的安装目录（如/usr/local/hadoop，Ubuntu不好选择目录，直接输入就行）。

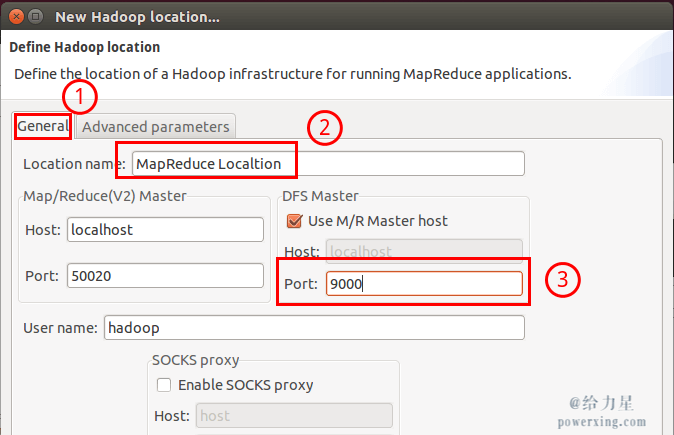


第二步：切换 Map/Reduce 开发视图，选择 Window 菜单下选择 Open Perspective -> Other（CentOS 是 Window -> Perspective -> Open Perspective -> Other），弹出一个窗体，从中选择 Map/Reduce 选项即可进行切换。

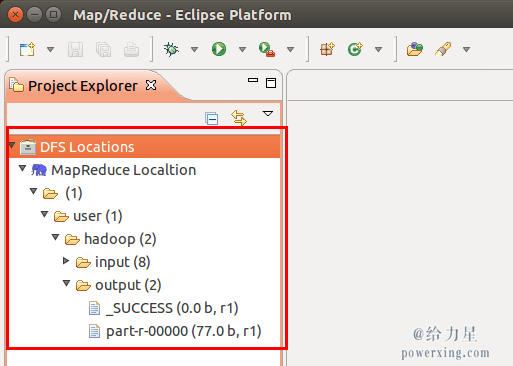


第三步：建立与 Hadoop 集群的连接，点击 Eclipse软件右下角的 Map/Reduce Locations 面板，在面板中单击右键，选择 New Hadoop Location。  


在弹出来的 General 选项面板中，General 的设置要与 Hadoop 的配置一致。一般两个 Host 值是一样的，如果是伪分布式，填写 localhost 即可，另外我使用的[Hadoop伪分布式配置](http://www.powerxing.com/install-hadoop/)，设置 fs.defaultFS 为 hdfs://localhost:9000，则 DFS Master 的 Port 要改为 9000。Map/Reduce(V2) Master 的 Port 用默认的即可，Location Name 随意填写。



# 在Eclipse中操作HDFS中的文件



这是HDFS中的文件列表，右键电机可以上传、下载、删除HDFS中的文件。如果无法查看，可右键点击Location尝试Reconnect或重启Eclipse

**Tips**

HDFS 中的内容变动后，Eclipse 不会同步刷新，需要右键点击 Project Explorer中的 MapReduce Location，选择 Refresh，才能看到变动后的文件。

# 在Eclipse中创建MapReduce项目

新建一个MapReduce项目，命名为WordCount

新建一个类，类中内容如下：

package org.apache.hadoop.examples;

import java.io.IOException;

import java.util.StringTokenizer;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.IntWritable;

import org.apache.hadoop.io.Text;

import org.apache.hadoop.mapreduce.Job;

import org.apache.hadoop.mapreduce.Mapper;

import org.apache.hadoop.mapreduce.Reducer;

import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;

import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

import org.apache.hadoop.util.GenericOptionsParser;

public class WordCount {

public static class TokenizerMapper

extends Mapper<Object, Text, Text, IntWritable>{

private final static IntWritable one = new IntWritable(1);

private Text word = new Text();

public void map(Object key, Text value, Context context

) throws IOException, InterruptedException {

StringTokenizer itr = new StringTokenizer(value.toString());

while (itr.hasMoreTokens()) {

word.set(itr.nextToken());

context.write(word, one);

}

}

}

public static class IntSumReducer

extends Reducer<Text,IntWritable,Text,IntWritable> {

private IntWritable result = new IntWritable();

public void reduce(Text key, Iterable<IntWritable> values,

Context context

) throws IOException, InterruptedException {

int sum = 0;

for (IntWritable val : values) {

sum += val.get();

}

result.set(sum);

context.write(key, result);

}

}

public static void main(String[] args) throws Exception {

Configuration conf = new Configuration();

String[] otherArgs = new GenericOptionsParser(conf, args).getRemainingArgs();

if (otherArgs.length != 2) {

System.err.println("Usage: wordcount <in> <out>");

System.exit(2);

}

Job job = new Job(conf, "word count");

job.setJarByClass(WordCount.class);

job.setMapperClass(TokenizerMapper.class);

job.setCombinerClass(IntSumReducer.class);

job.setReducerClass(IntSumReducer.class);

job.setOutputKeyClass(Text.class);

job.setOutputValueClass(IntWritable.class);

FileInputFormat.addInputPath(job, new Path(otherArgs[0]));

FileOutputFormat.setOutputPath(job, new Path(otherArgs[1]));

System.exit(job.waitForCompletion(true) ? 0 : 1);

}

}

在运行MapReduce程序前，还要执行一项重要操作：将/usr/local/hadoop/etc/hadoop中有修改过的配置文件以及log4j.properties复制到WordCount项目下的src文件夹中

1. cp /usr/local/hadoop/etc/hadoop/core-site.xml ~/workspace/WordCount/src
2. cp /usr/local/hadoop/etc/hadoop/hdfs-site.xml ~/workspace/WordCount/src
3. cp /usr/local/hadoop/etc/hadoop/log4j.properties ~/workspace/WordCount/src

复制完成后，务必右键点击 WordCount 选择 refresh 进行刷新（不会自动刷新，需要手动刷新）

右键点击刚创建的 WordCount.java，选择 Run As -> Run Configurations，在此处可以设置运行时的相关参数（如果 Java Application 下面没有 WordCount，那么需要先双击 Java Application）。切换到 “Arguments” 栏，在 Program arguments 处填写 “input output”

点击工具栏中的 Run 图标，或者右键点击 Project Explorer 中的 WordCount.java，选择 Run As -> Run on Hadoop，就可以运行 MapReduce 程序了。

