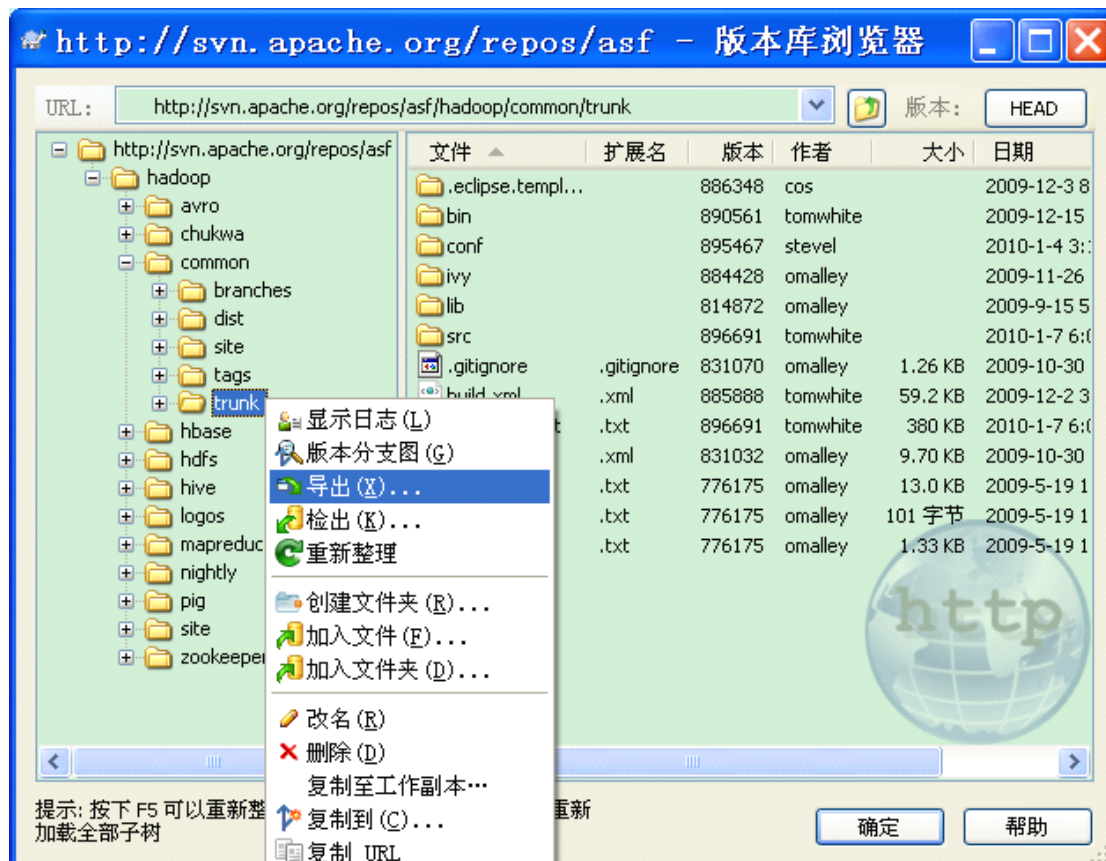


Hadoop 源代码 eclipse 编译教程

一见 2009-8-26 www.hadoopor.com/hadoopor@foxmail.com

1. 下载 Hadoop 源代码

Hadoop 各成员源代码下载地址: <http://svn.apache.org/repos/asf/hadoop>, 请使用 SVN 下载, 在 SVN 浏览器中将 trunk 目录下的源代码 check-out 出来即可:



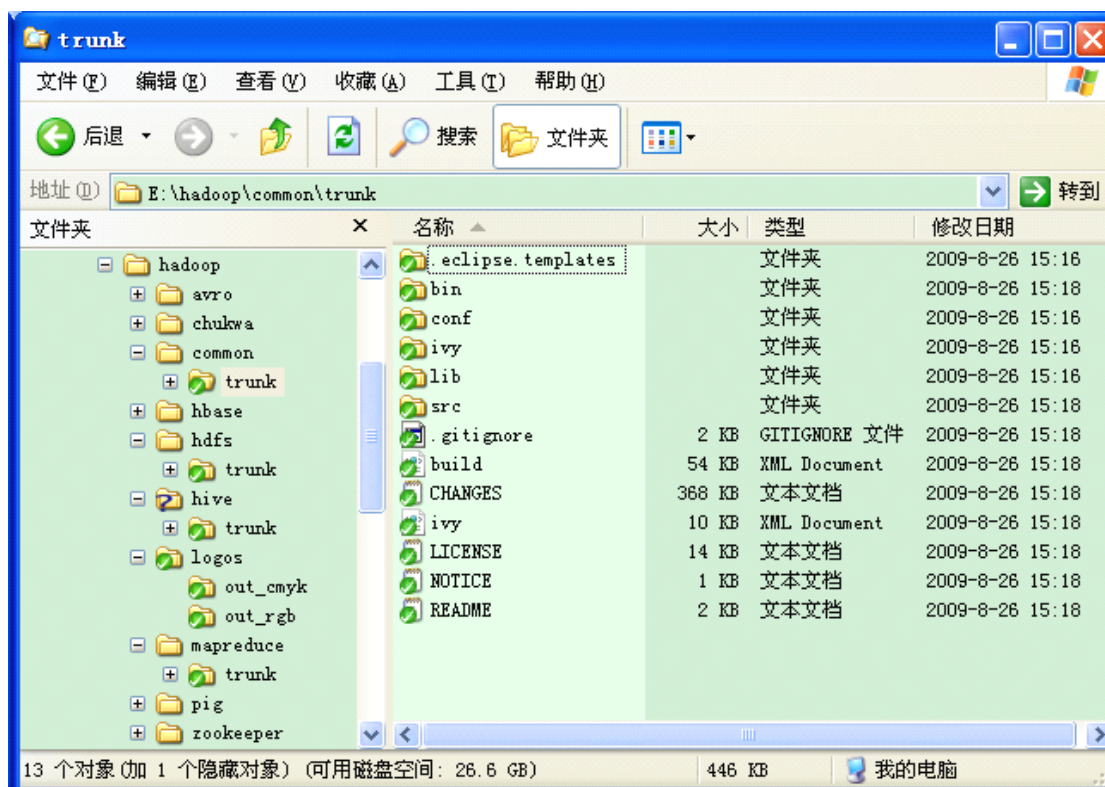
请注意只 check-out 出 SVN 上的 trunk 目录下的内容, 如:

<http://svn.apache.org/repos/asf/hadoop/common/trunk>,

而不是 <http://svn.apache.org/repos/asf/hadoop/common>,

原因是 <http://svn.apache.org/repos/asf/hadoop/common> 目录下包括了很多非源代码文件, 并且很庞大, 会导致需要很长的 check-out 时间。

建议组织成如下图所示的目录结构, 以保持本地的目录结构和 SVN 上的目录结构一致:



2. 准备编译环境

2.1. Hadoop 代码版本

本教程所采用的 Hadoop 是北京时间 2009-8-26 日上午下载的源代码，和 hadoop-0.19.x 版本的差异可能较大。

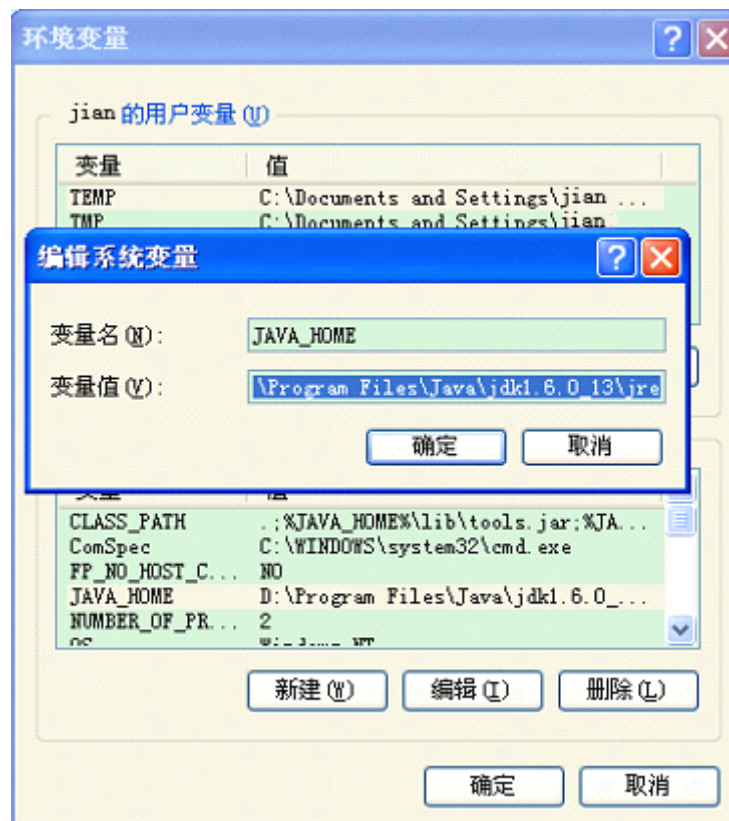
2.2. 联网

编译 Hadoop 会依赖很多第三方库，但编译工具 Ant 会自动从网上下载缺少的库，所以必须保证机器能够访问 Internet。

2.3. java

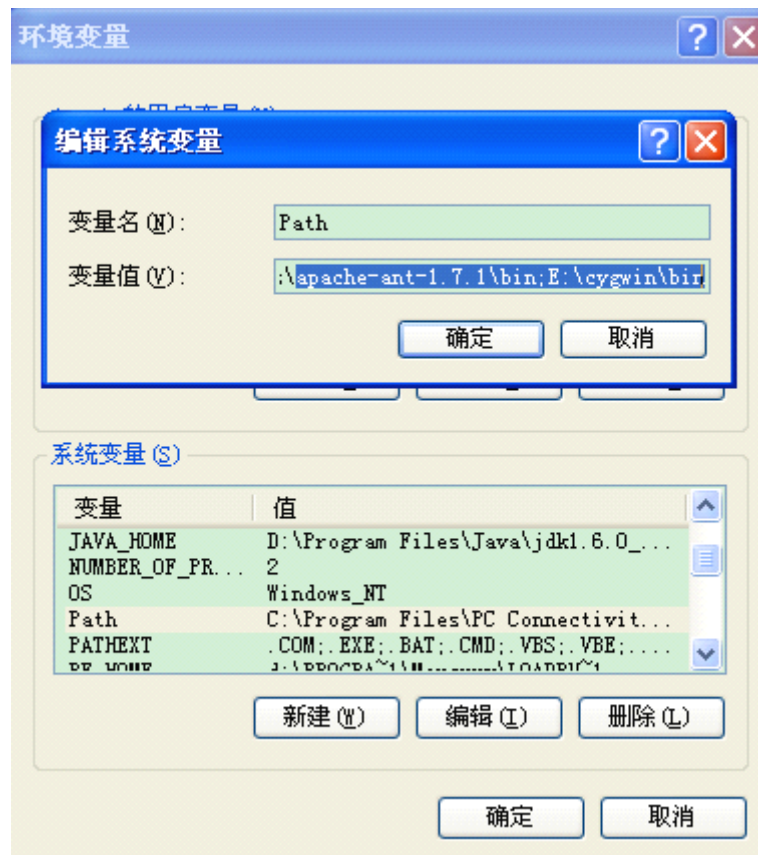
编译 Hadoop 要求使用 1.6 或更新的 JDK，可以从：

<http://java.sun.com/javase/downloads/index.jsp> 上下载 JDK。安装好之后，请设置好 JAVA_HOME 环境变量，如下图所示：



2.4. Ant 和 Cygwin

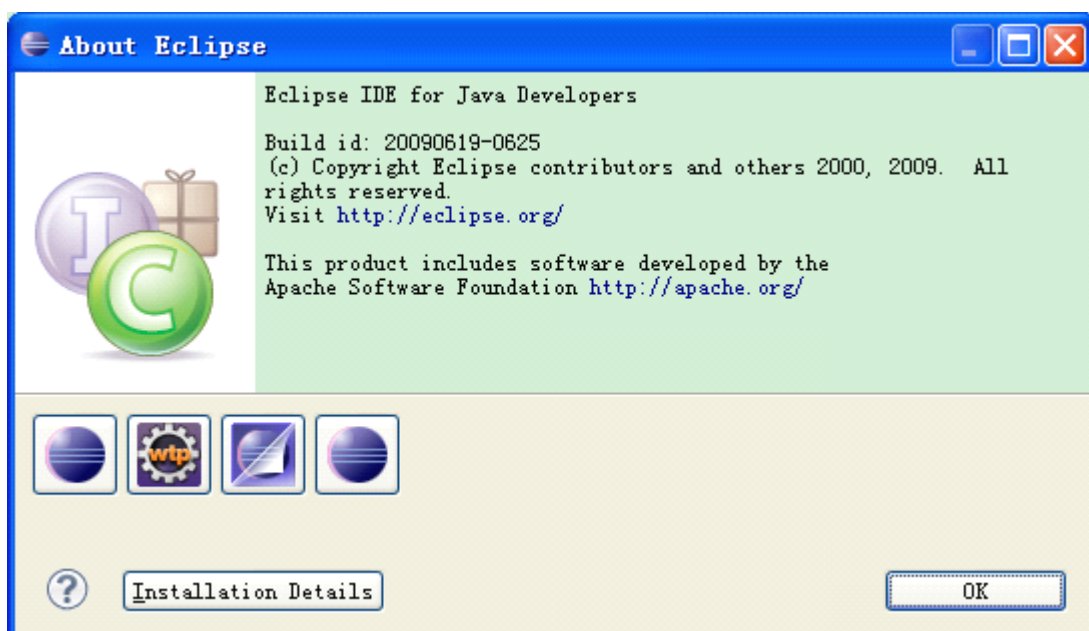
需要使用 Ant 工具来编译 Hadoop, 而 Ant 需要使用到 Cygwin 提供的一些工具, 如 sed 等, 可以从: <http://ant.apache.org/ivy/download.cgi> 下载 Ant, 从 <http://www.cygwin.cn/> 下载 Cygwin (Cygwin 的安装, 请参考《在 Windows 上安装 Hadoop 教程》一文)。安装好之后, 需要将 Ant 和 Cygwin 的 bin 目录加入到环境变量 PATH 中, 如下图所示:



在安装 Cygwin 时，建议将 **SVN** 安装上，因为在 Ant 编译过程中会通过 SVN 下载些文件，但这个不是必须的，下载不成功时，并未见出错，编译仍然可以成功。

2.5. Eclipse

Eclipse 则可以从 <http://www.eclipse.org/downloads/> 上下载，本教程使用的版本如下图所示：

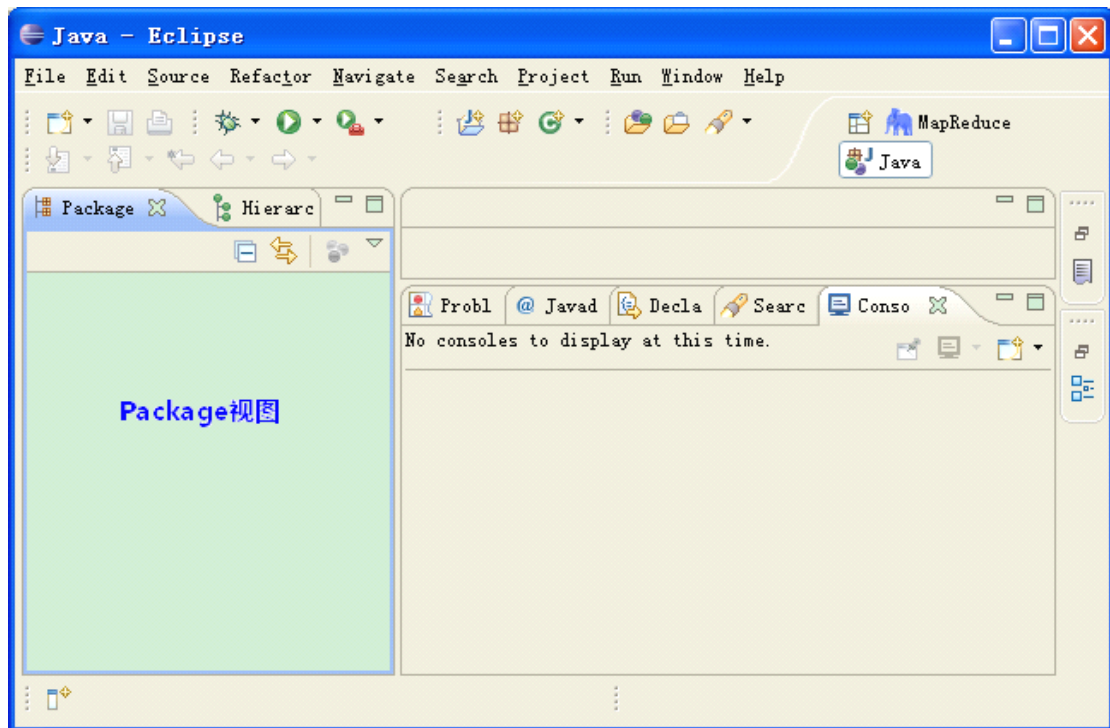


3. 编译 Hadoop

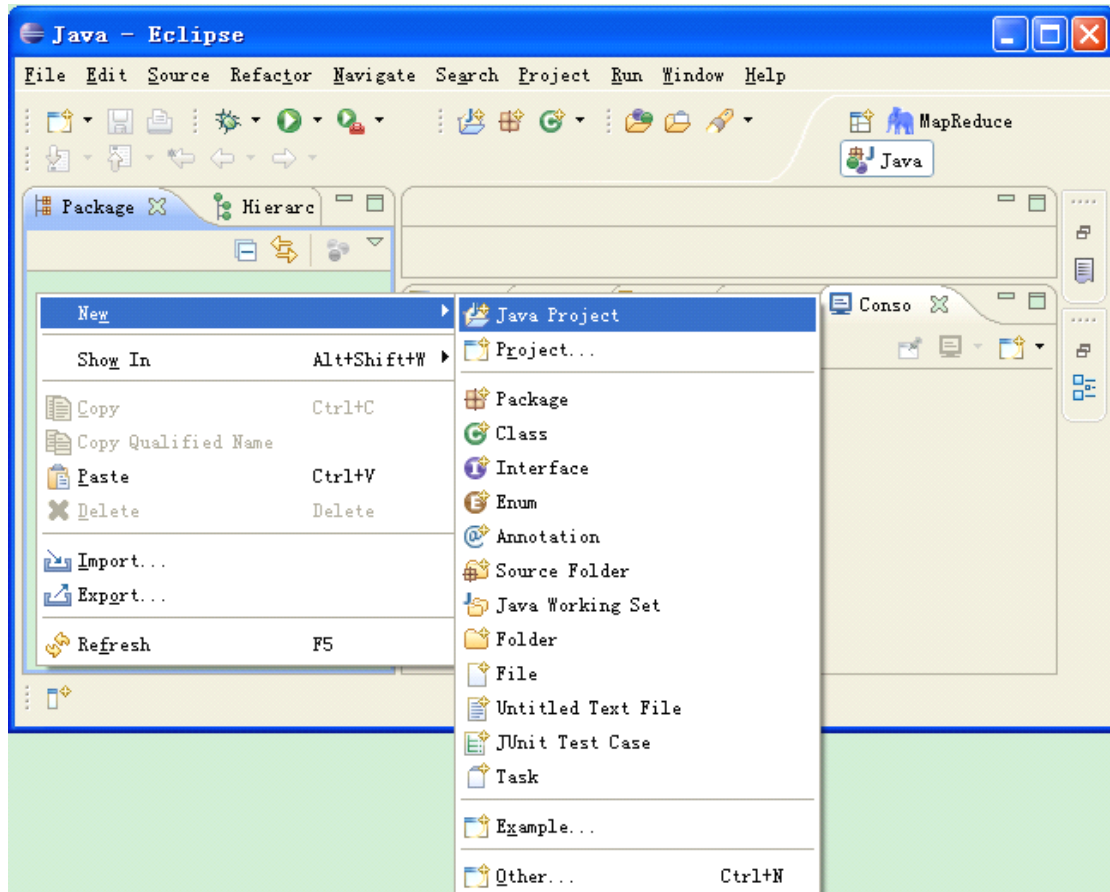
在这里，我们以编译 Hadoop 家庭成员 common 为例，对 Hadoop 其它成员的编译方法是类似的。

3.1. 编译 common 成员

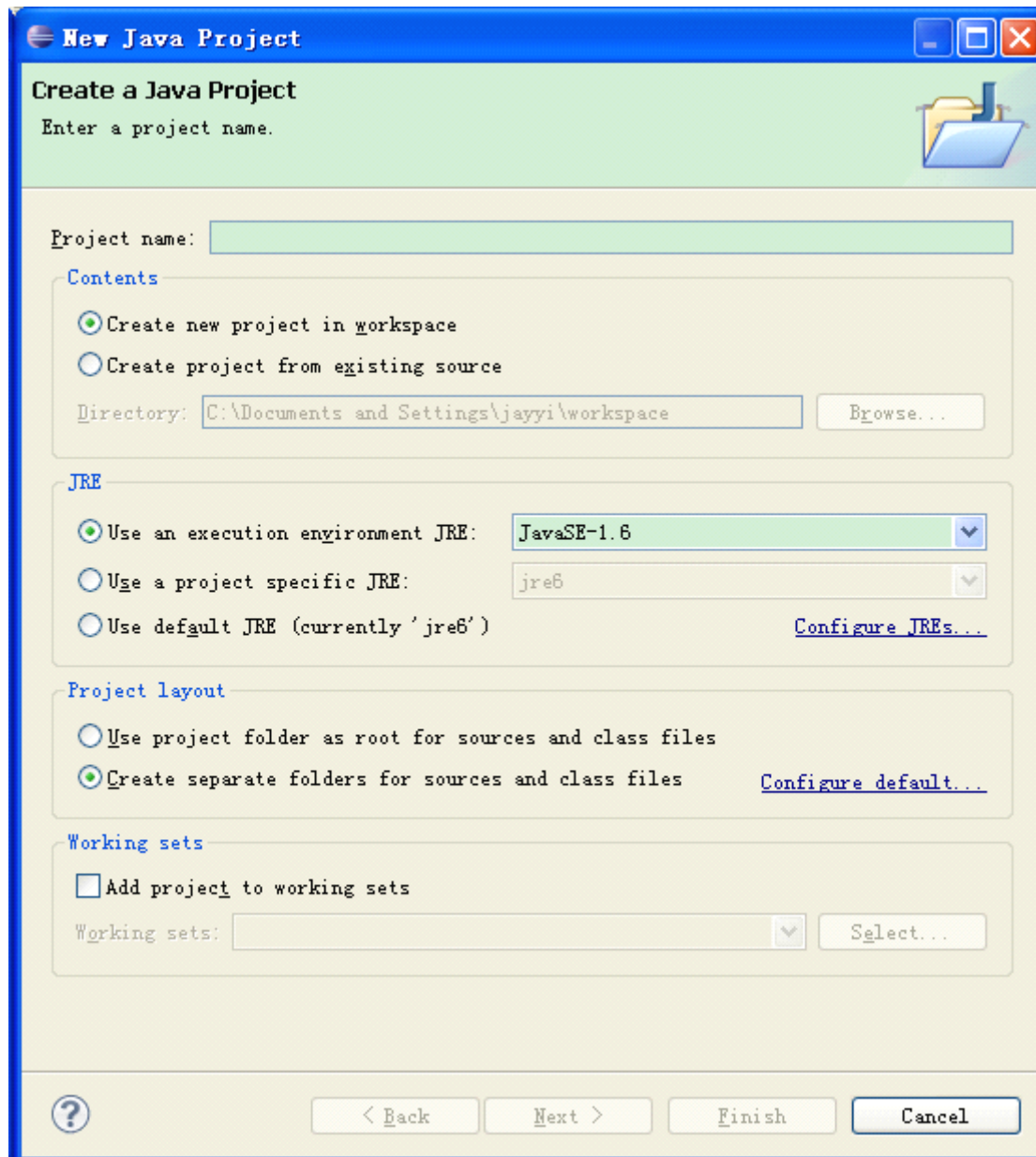
步骤 1) 启动 Eclipse，进入 Eclipse 主界面，如下图所示：



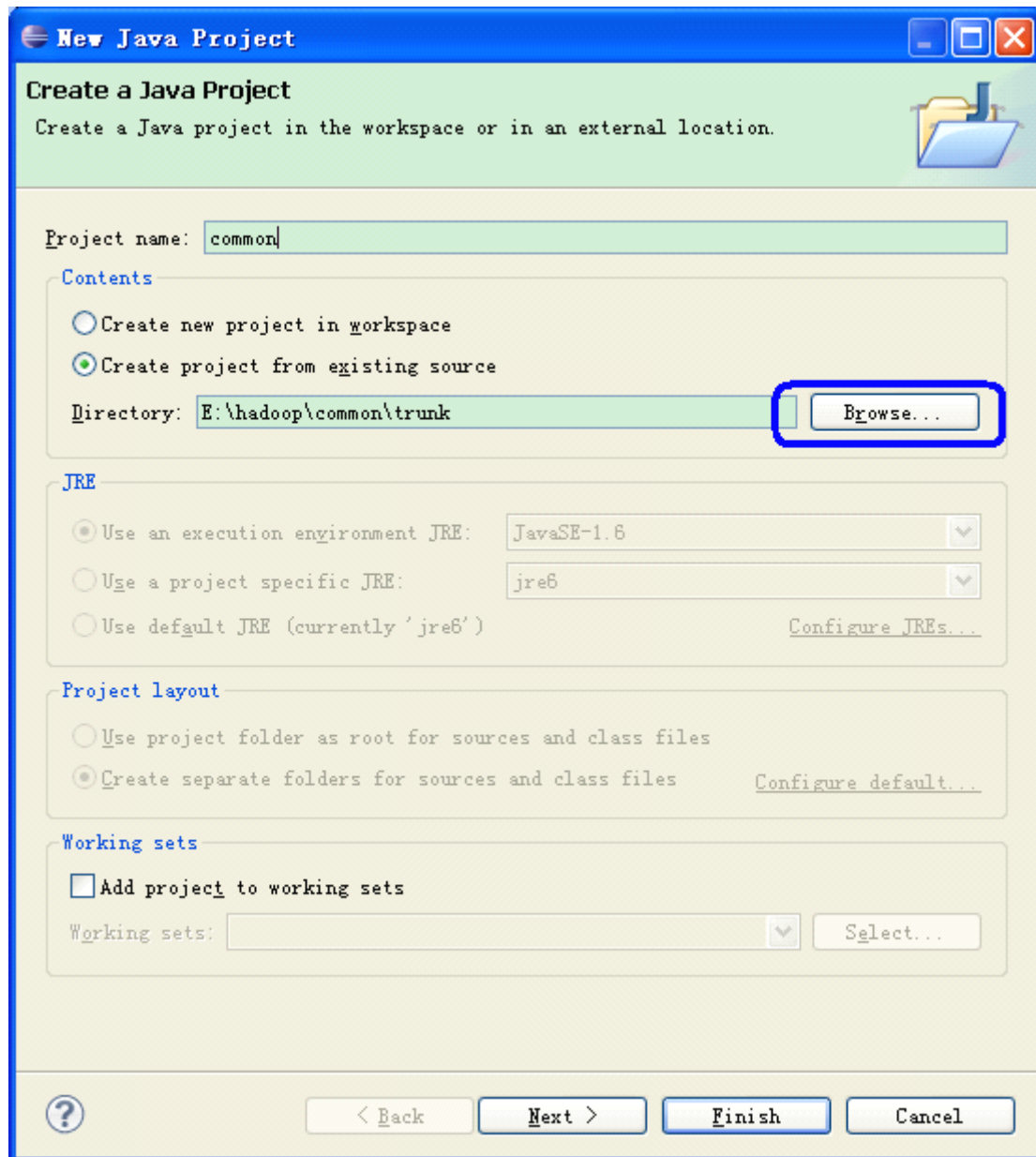
步骤 2) 在 Eclipse 的 Package 视图中单击右键，选择 New->Java Project，如下图所示：



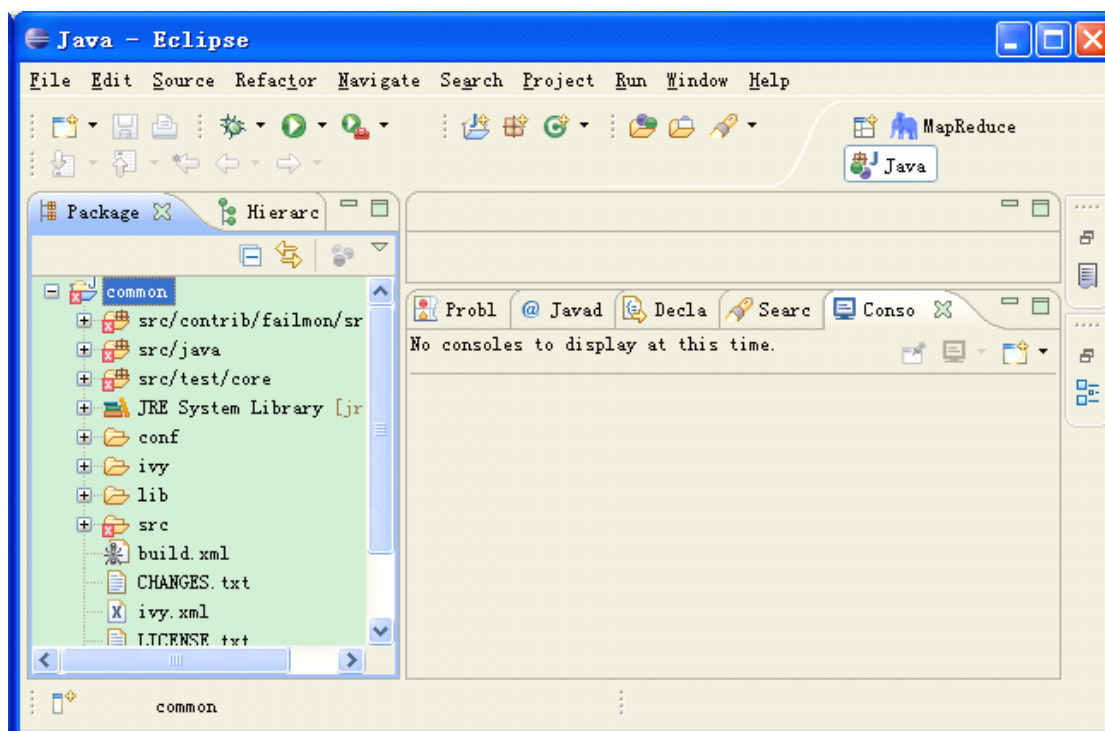
这个时候会打开如下图所示的对话框：



步骤 3) 选择源代码目录和设置 Project 名。在下图所示的对话框中，点击 Browse 按钮，选择 common 源代码目录，并设置 Project name 为 common，然后点击 Finish 按钮。

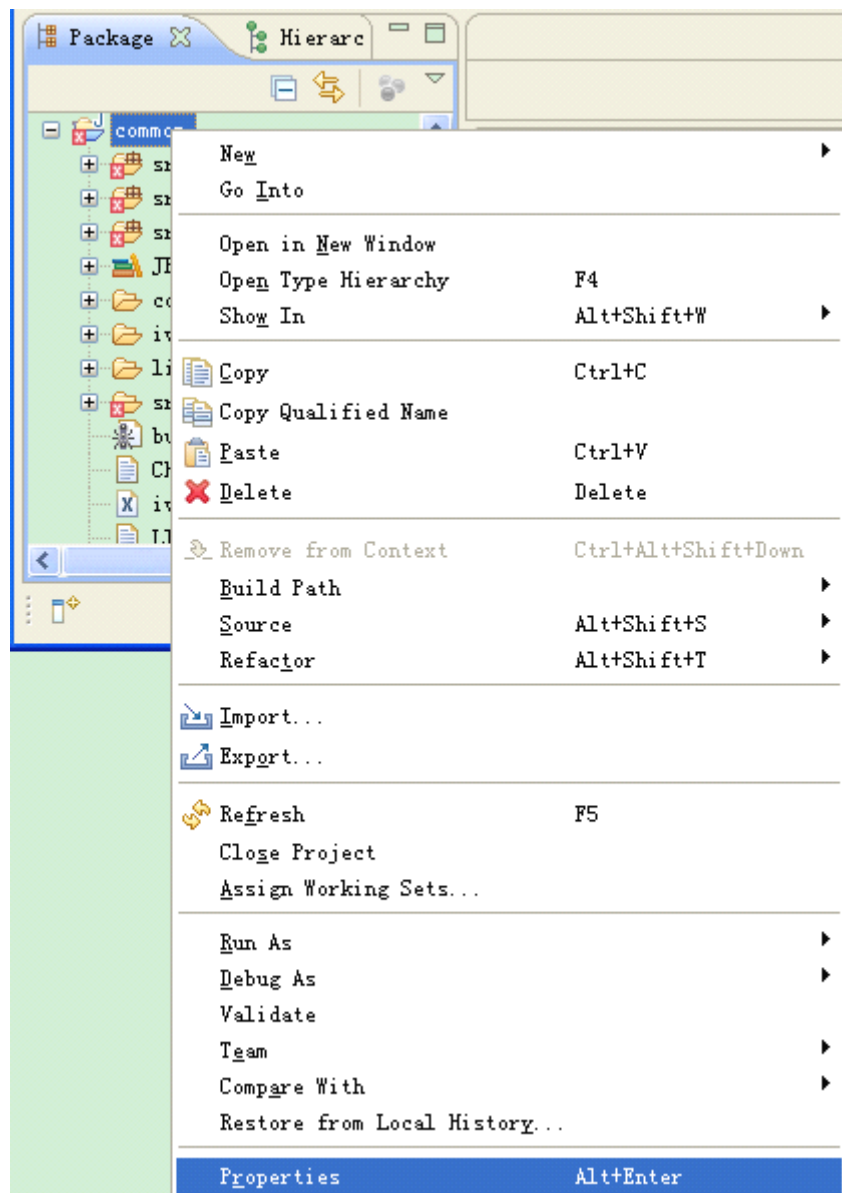


当工程导入完成后，上图所示的对话框会自动关闭，并重新进入 Eclipse 主界面，如下图所示，可以看到 common 已经导入进来，但可以看到 common 上有红叉叉，原因是因为 Eclipse 默认使用了 Java Builder，而不是 Ant Builder，所以下一步就是设置使用 Ant Builder。

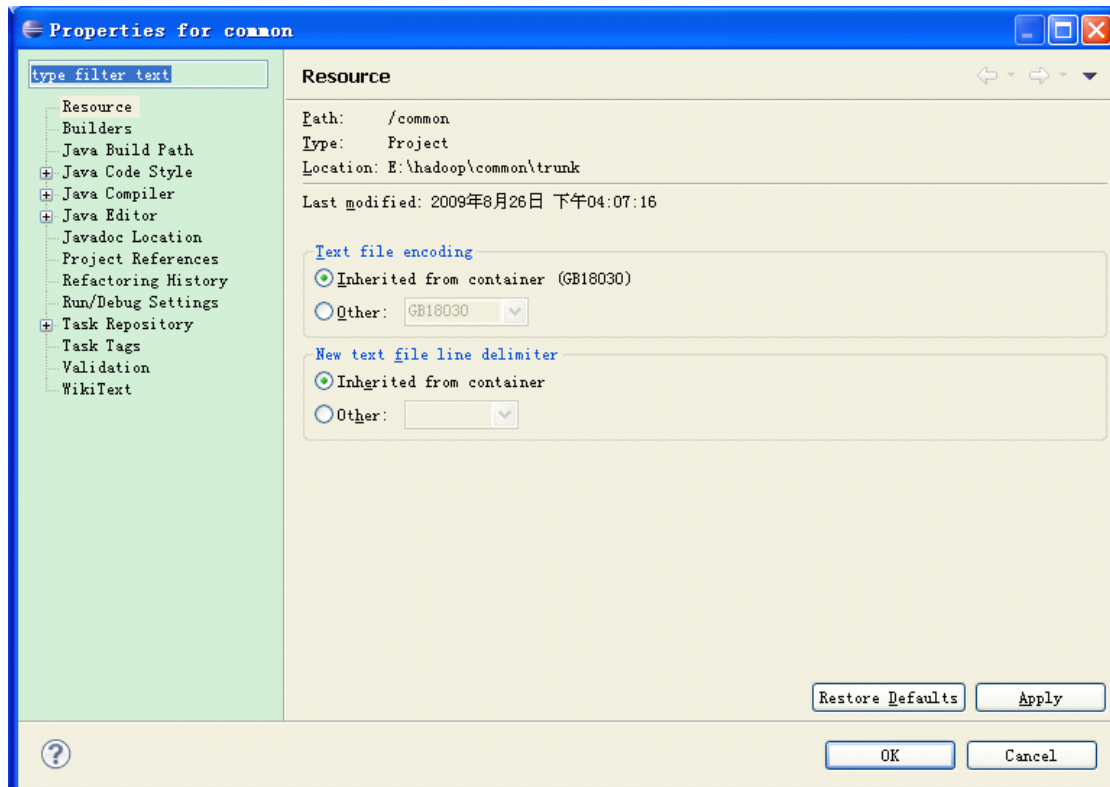


步骤 4) 设置 Builder 为 Ant

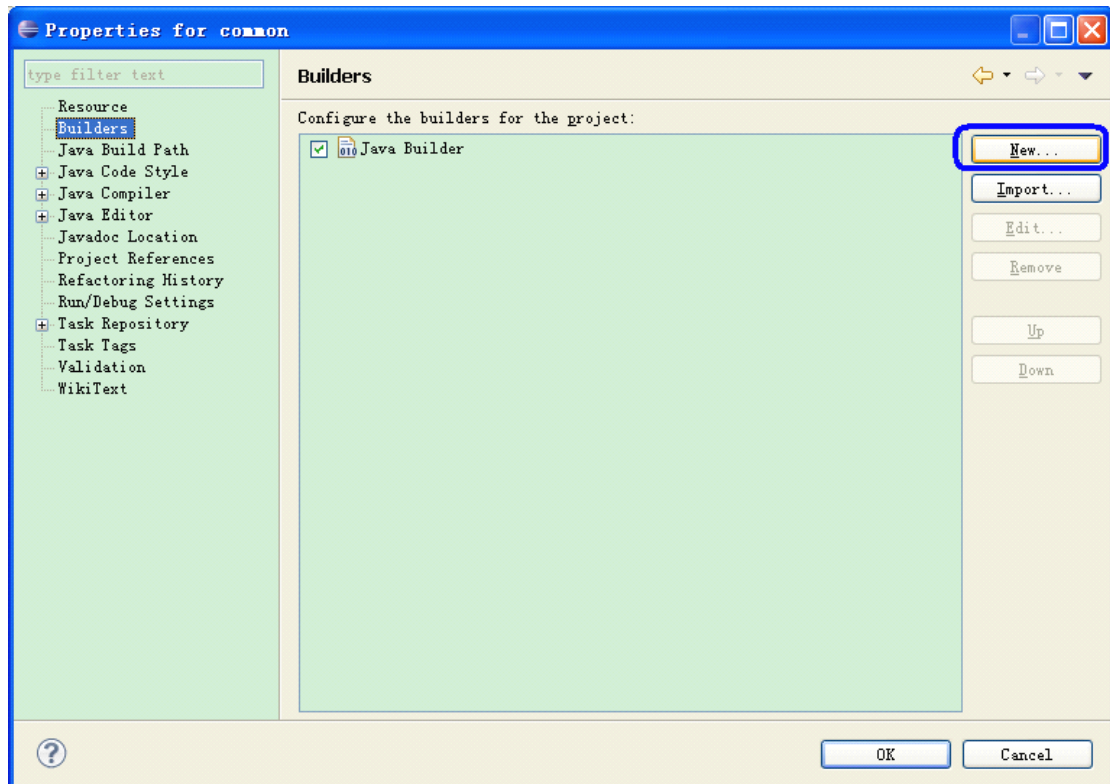
在上图所示对话框中，右单击 `common`，弹出如下图所示的菜单：



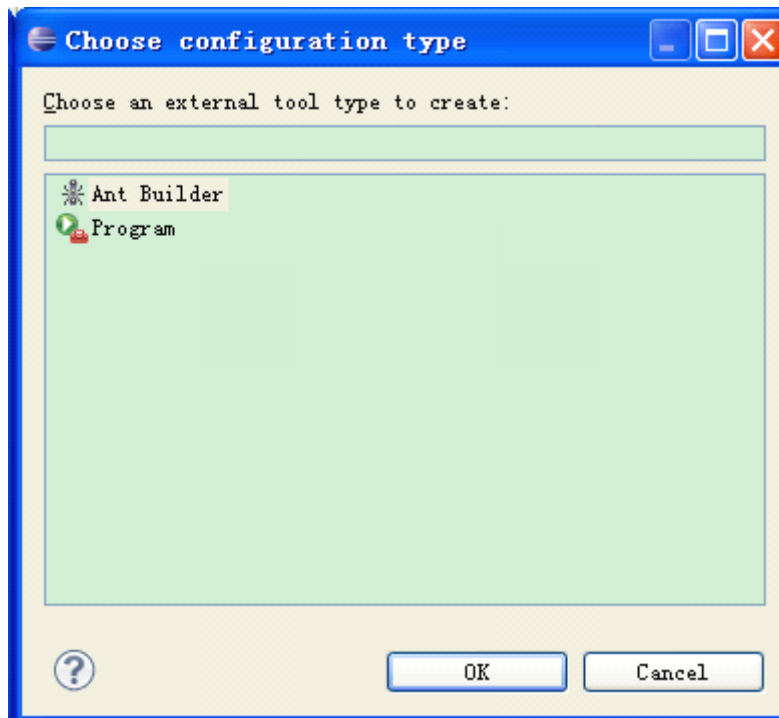
选择 Properties 子菜单，这个时候会弹出如下图所示的对话框：



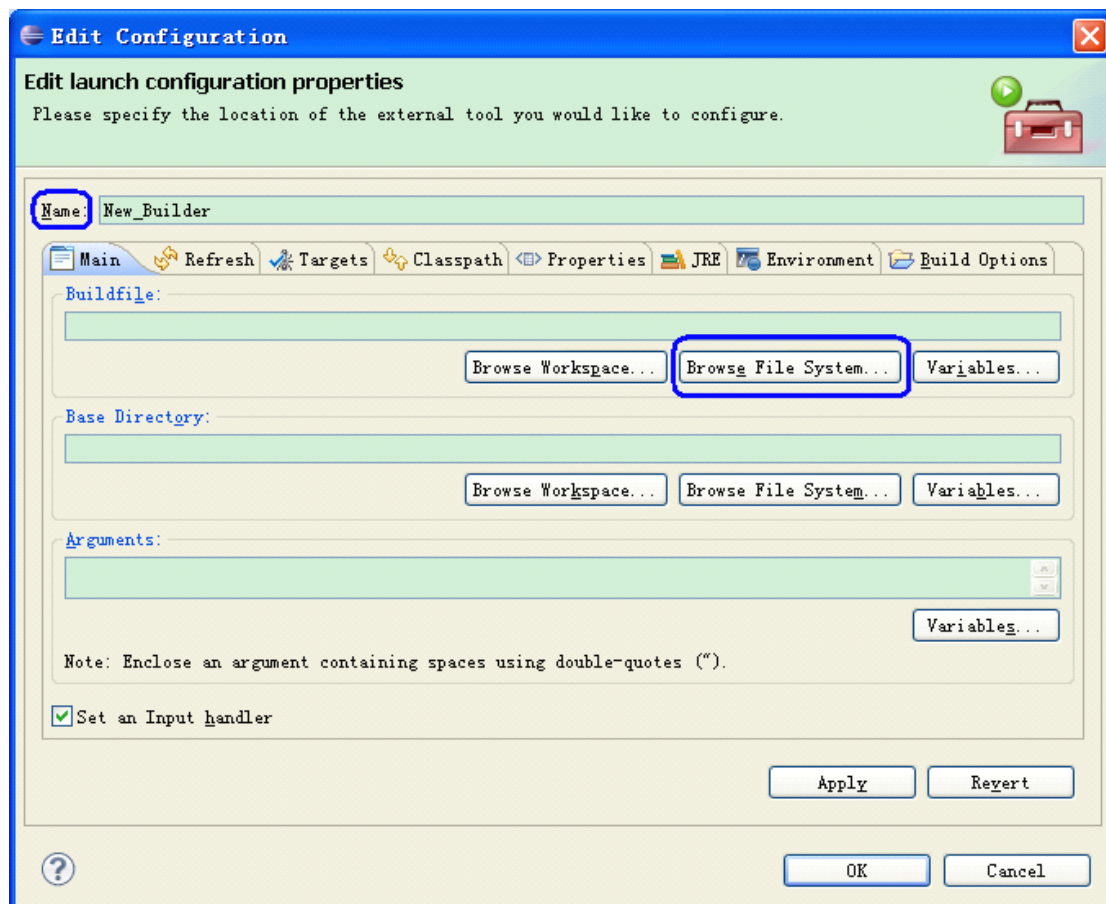
在上图所示的对话框中，点击 Builders，会出现如下图所示界面：



在上图所示的界面中，点击 New 按钮，弹出如下图所示的对话框：

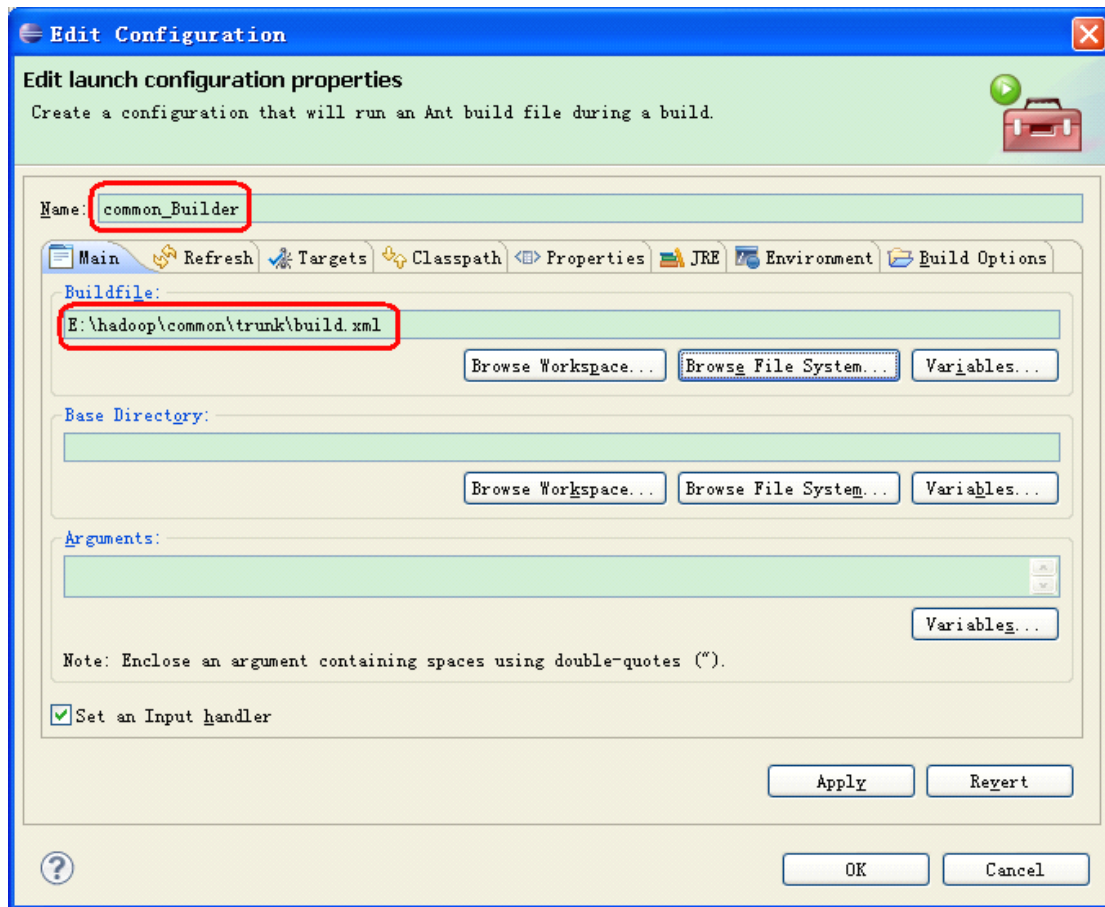


在上图所示的对话框中选中 Ant Builder，然后点击 OK 按钮，这个时候会弹出如下图所示的新对话框：

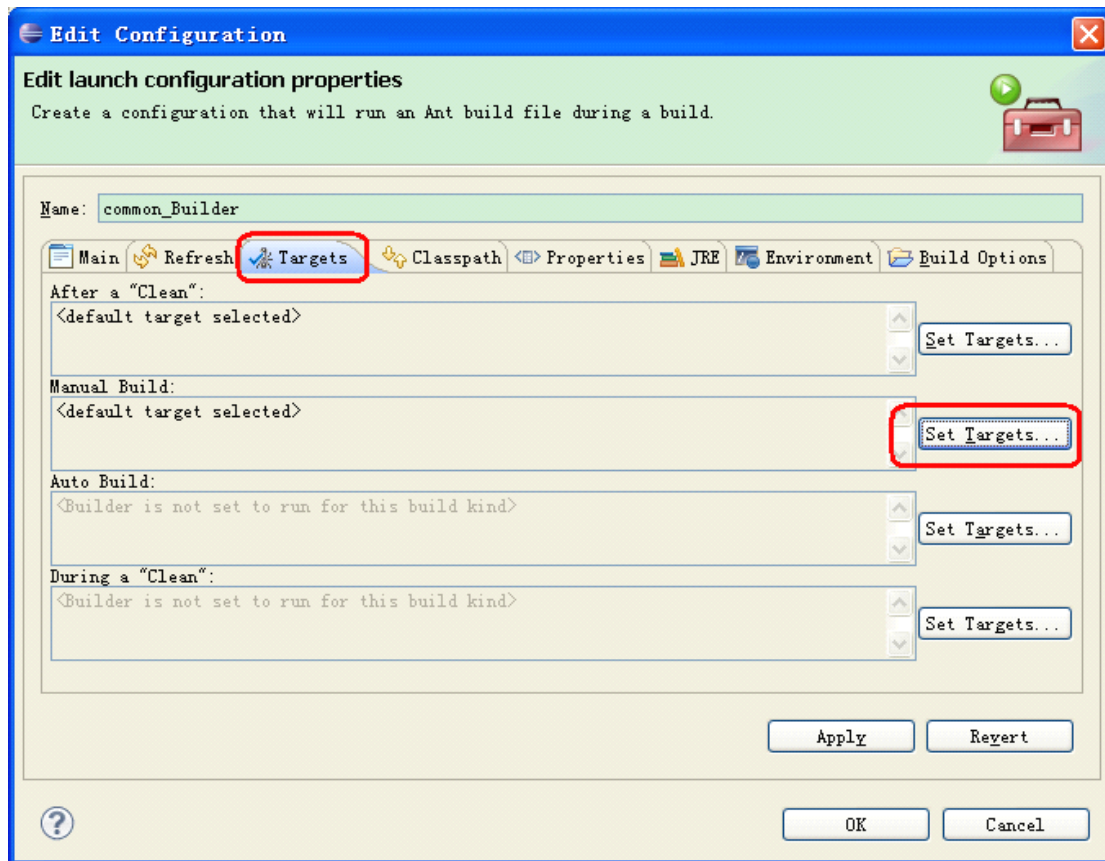


点击上图所示的对话框中的 Browse File System 按钮，选择 common 源代码目录下的 build.xml 文件，并设置 Name 为 common_Builder（Name 可以改成其它的，但建议使用

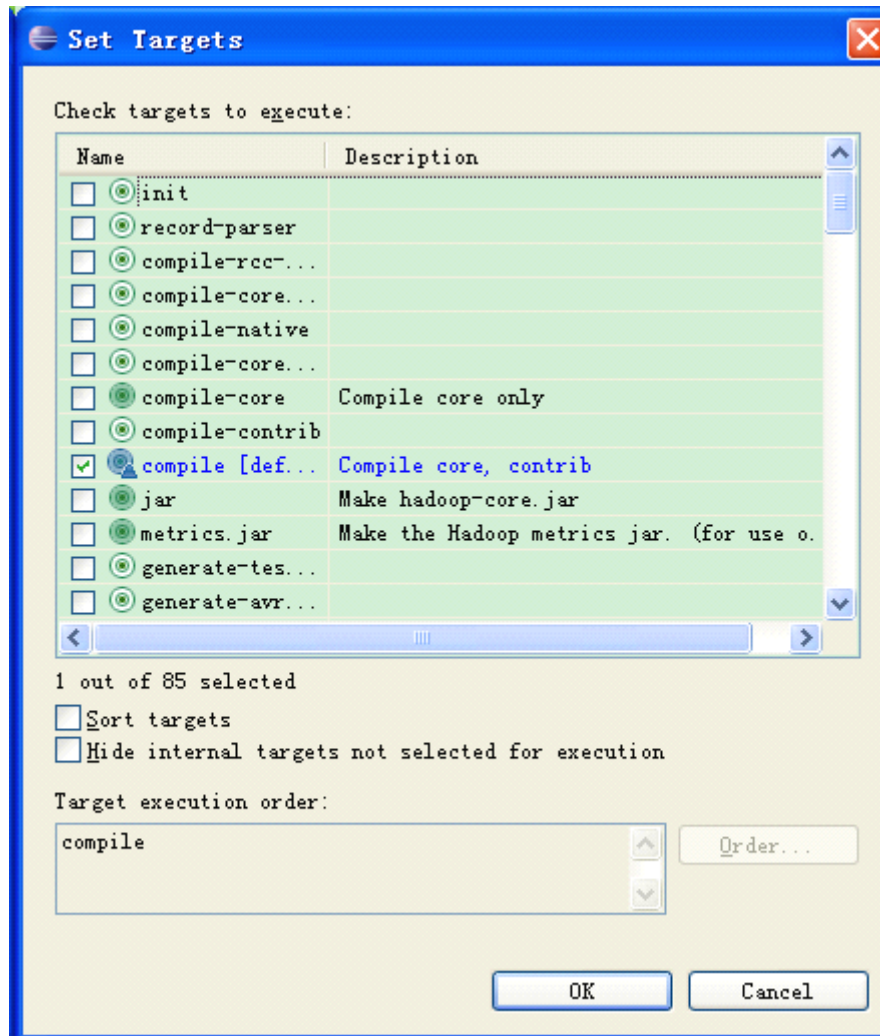
common_Builder，因为这样名副其实)，操作结果如下图所示：



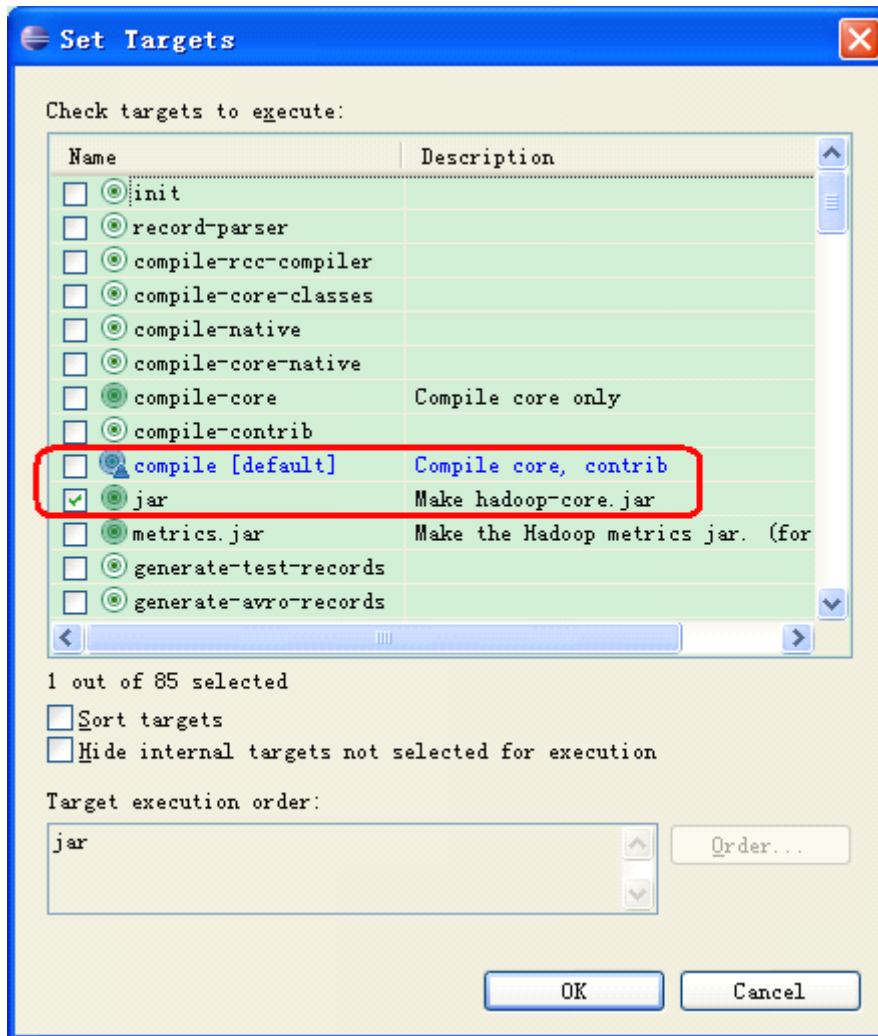
除上图所示的设置外，还需要设置如下图所示的 Targets，建议设置成 Manual Build 编译方式，而不是 Auto Build 编译方式。原因是在 Auto Build 模式下，任何修改都会触发编译，而 Manual Build 模式下，只在需要的时候，点击编译按钮或菜单编译即可。



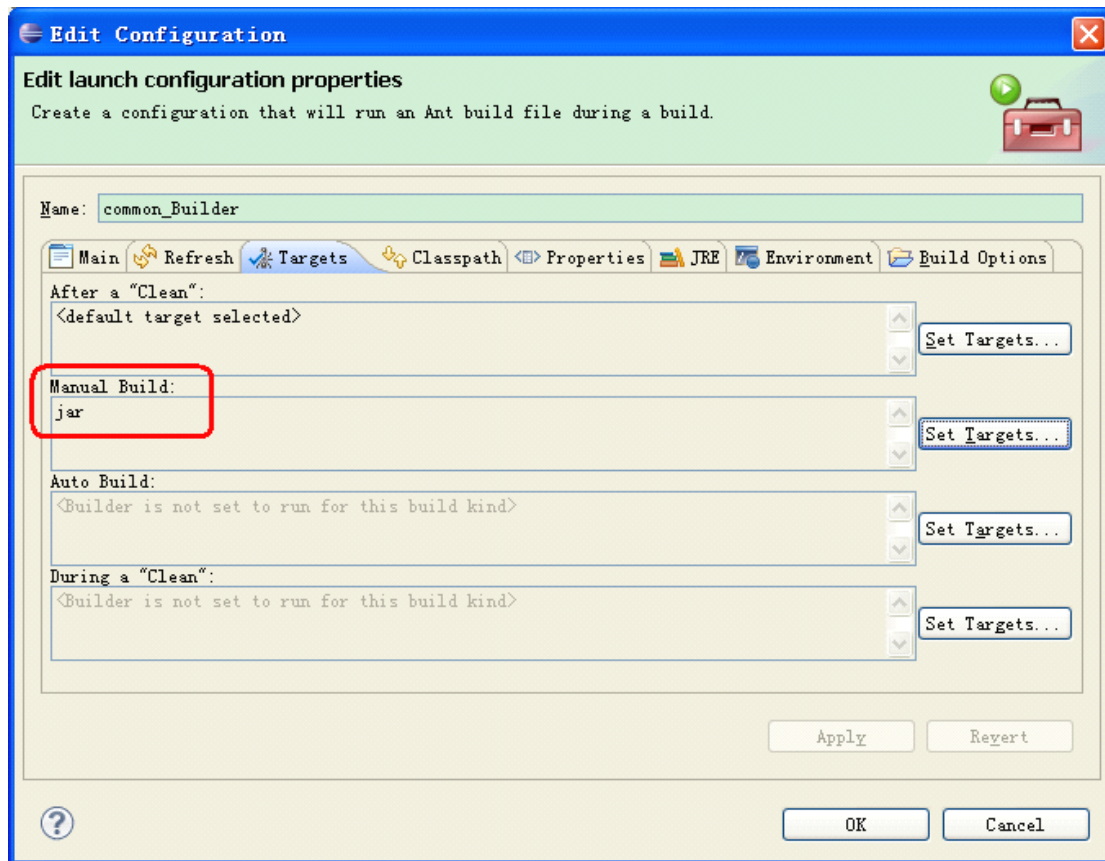
点击上图所示对话框中的 Manual Build 栏的 Set Targets 按钮，弹出如下图所示对话框：



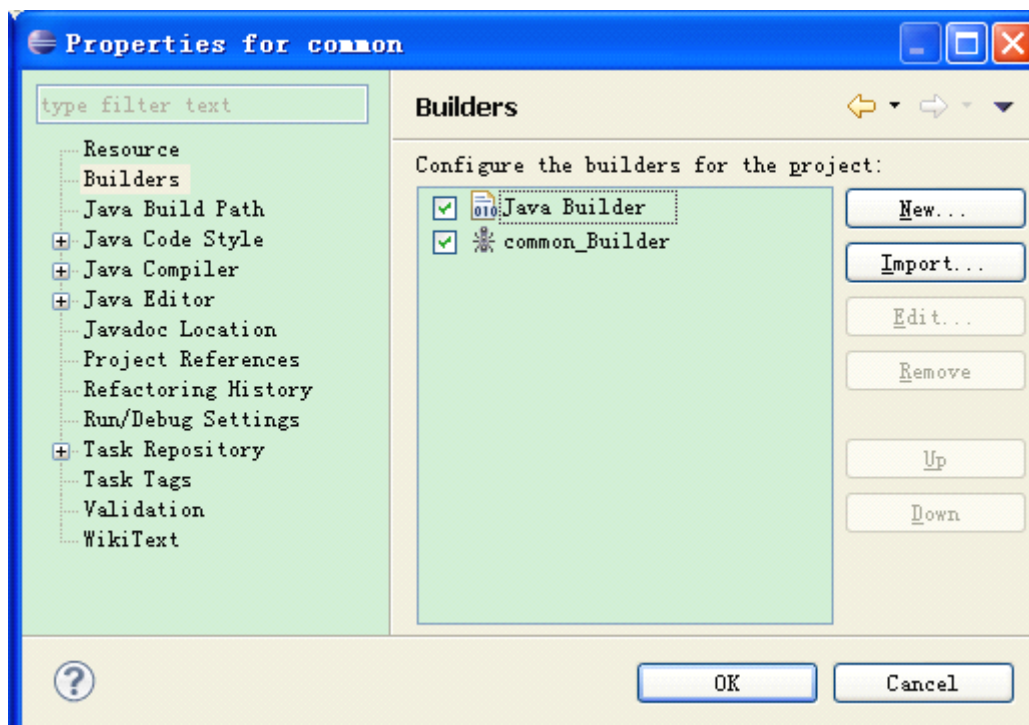
Hadoop 各成员都需要编译成 jar，所以做如下图所示的一个修改：



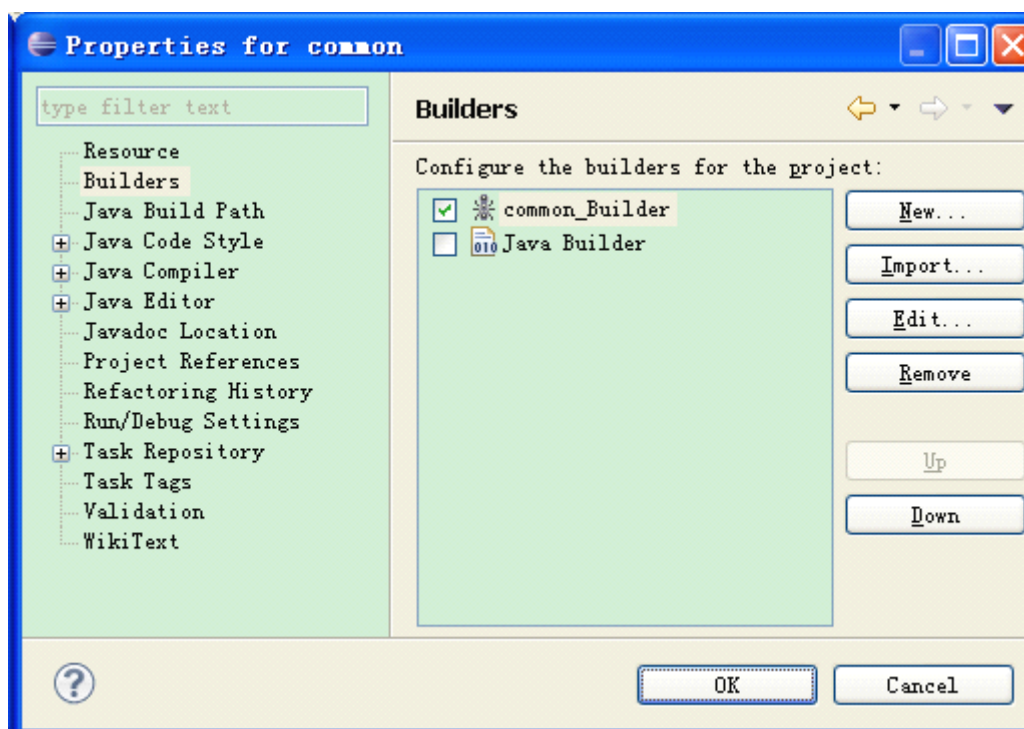
点击上图所示对话框中的 OK 按钮，返回如下图所示的 Edit Configuration 对话框：



点击上图所示对话框中的 OK 按钮，回到如下图所示的对话框：

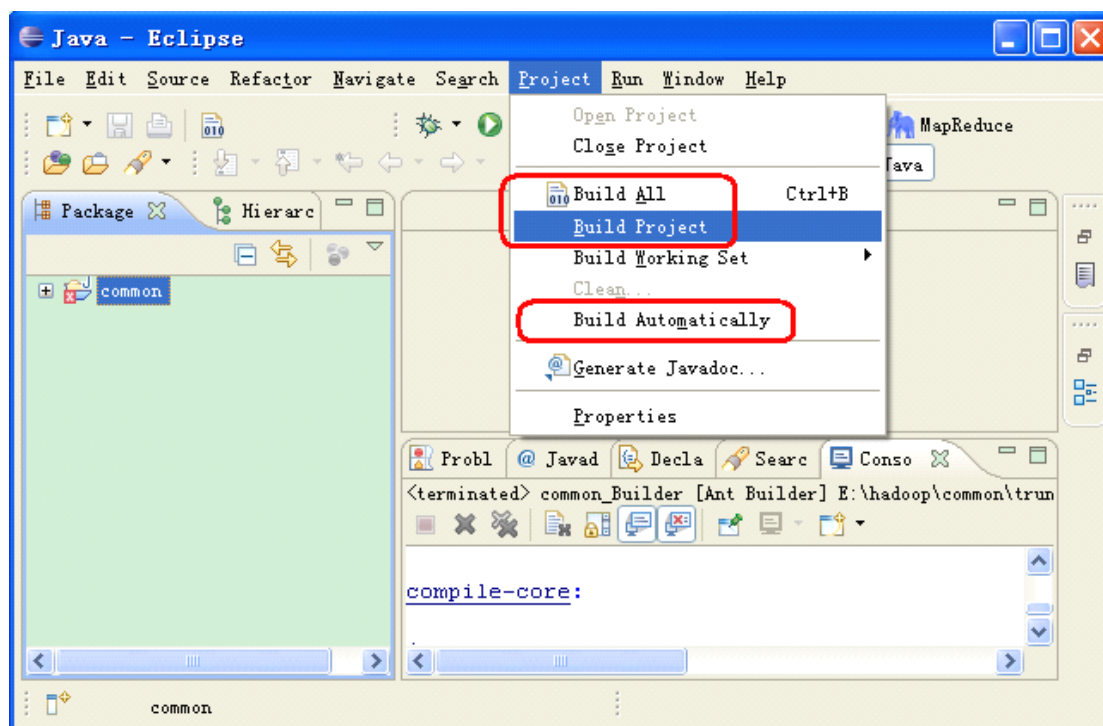


将上图对话框中的 Java Builder 下移，并它前面的勾去掉，结果如下图所示：



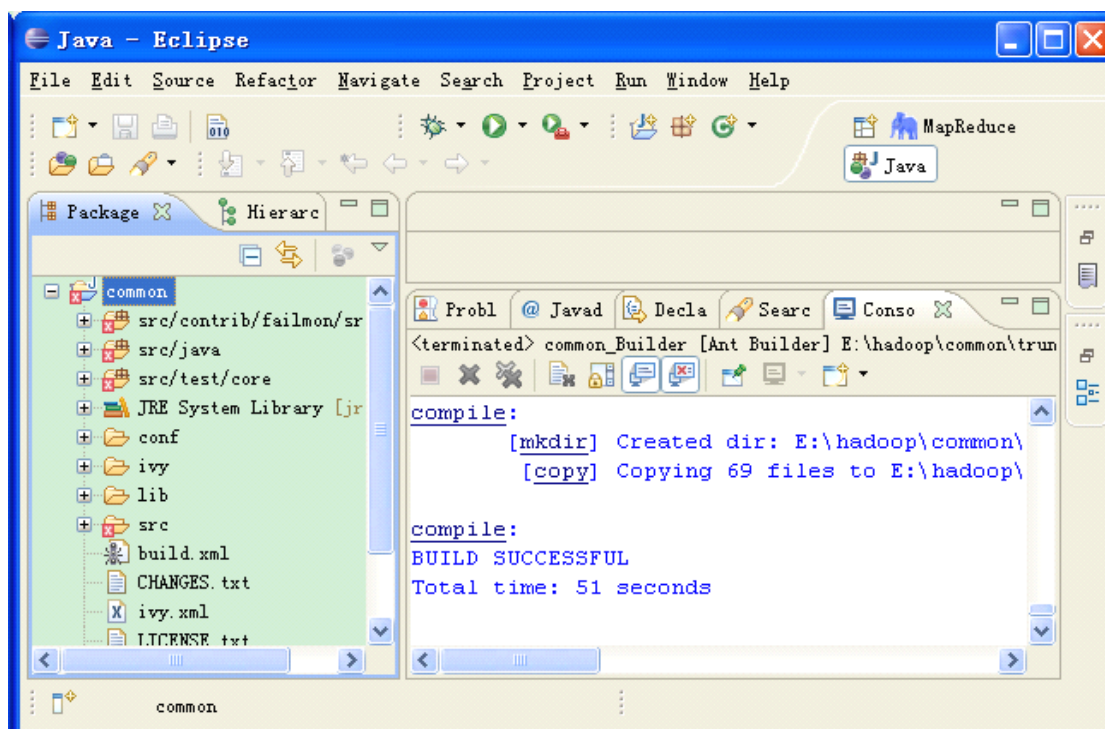
点击上图所示对话框中的 OK 按钮，上图所示的对话框会自动关闭，这个时候会进入 Eclipse 主界面。

由于之前选择了 Manual Build，所以需要人工方式驱动编译：

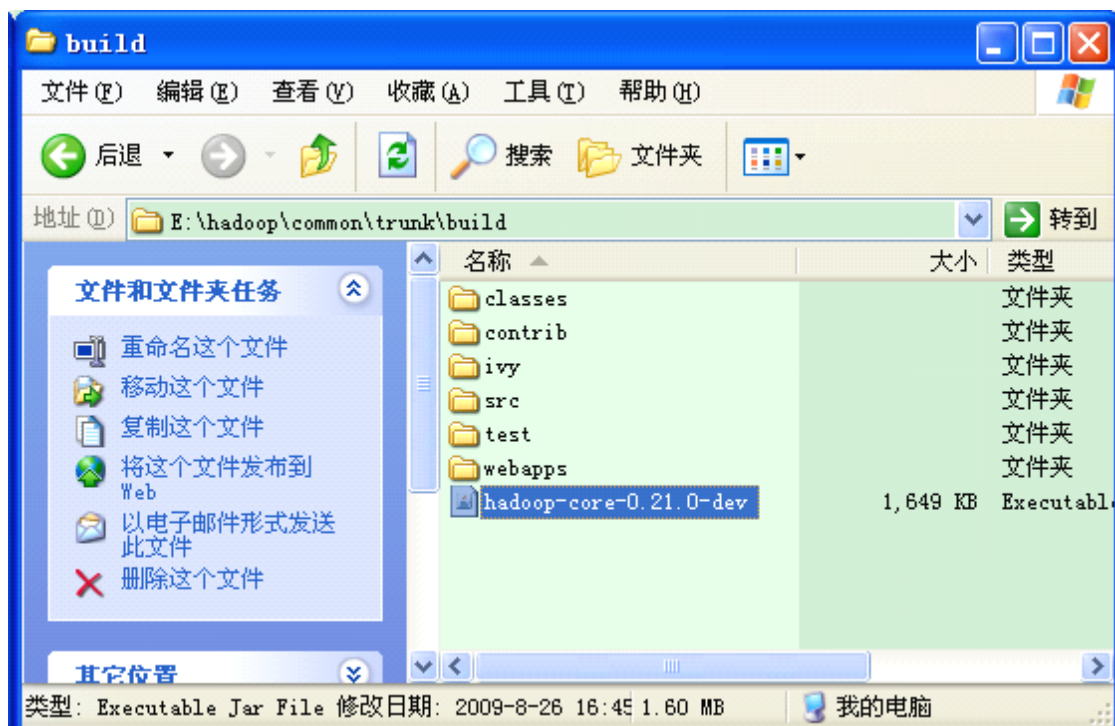


请注意，如果上图所示的菜单中的 Build Automatically 被勾中，则在 common 的右键菜单中可能不会出现 Build 子菜单。

编译成功后，可以看到如下图所显示的 BUILD SUCCESSFUL 字样：



在编译通过中，Ant 会自动从网上下载所依赖的库。从上图可以看到，编译只花了 51 秒，至此 common 的编译就成功结束了，可以在 build 目录下找到编译后生成的文件 hadoop-core-0.21.0-dev.jar。



3.2. 编译 Hadoop 其它成员

hdfs、mapreduce 和 hive 的编译方式同 common。

4. FAQ

4.1. 联网

确保可以上 internet, Ant 需要自动下载很多第三方依赖库, 如果不能上 internet, 编译会复杂很多。

4.2. 编译 hive

hive 的编译相对复杂些, 而且默认它会编译多个版本的 hive, 建立修改 shims 目录下的 ivy.xml 文件, 去掉不必要版本的编译, 默认的 ivy.xml 文件内容如下图所示。

```

1  <ivy-module version="2.0">
2      <info organisation="org.apache.hadoop.hive" module="shims"/>
3      <dependencies>
4          <dependency org="hadoop" name="core" rev="0.17.2.1">
5              <artifact name="hadoop" type="source" ext="tar.gz"/>
6          </dependency>
7          <dependency org="hadoop" name="core" rev="0.18.3">
8              <artifact name="hadoop" type="source" ext="tar.gz"/>
9          </dependency>
10         <dependency org="hadoop" name="core" rev="0.19.0">
11             <artifact name="hadoop" type="source" ext="tar.gz"/>
12         </dependency>
13         <dependency org="hadoop" name="core" rev="0.20.0">
14             <artifact name="hadoop" type="source" ext="tar.gz"/>
15         </dependency>
16         <conflict manager="all" />
17     </dependencies>
18 </ivy-module>

```

4.3. 编译生成文件位置

common 编译后生成 `build\hadoop-core-0.21.0-dev.jar`

hdfs 编译后生成 `build\hadoop-hdfs-0.21.0-dev.jar`

mapreduce 编译后生成 `build\hadoop-mapred-0.21.0-dev.jar`

hive 编译后生成 `build\service\hive_service.jar` 等 请注意并没有直接放在 **build** 目录下

hbase 编译后生成 `build\hbase-0.21.0-dev.jar`

有时候, 在编译某个时, 可能会出错, 这个时候可以跳过, 先编译其它的, 然后再 Refresh 后, 再编译, 可能就会成功了, 我第一把编译 mapreduce 就没有成功, 后 Refresh 后再编译就 OK 了。