

一、软件准备 见附件

1. 下载将jar包转化为EXE的工具EXE4J，下载地址为：

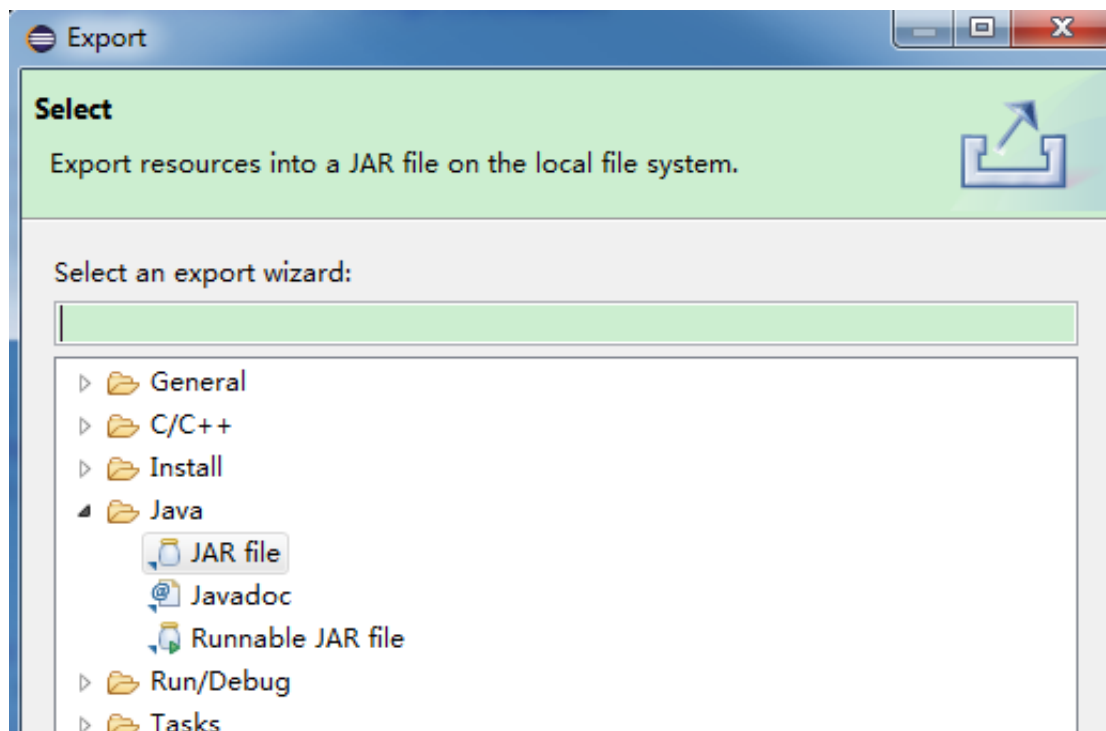
<https://www.ej-technologies.com/download/exe4j/files>

2. 下载完成之后，按照如下步骤安装EXE4J。

二、生产jar文件

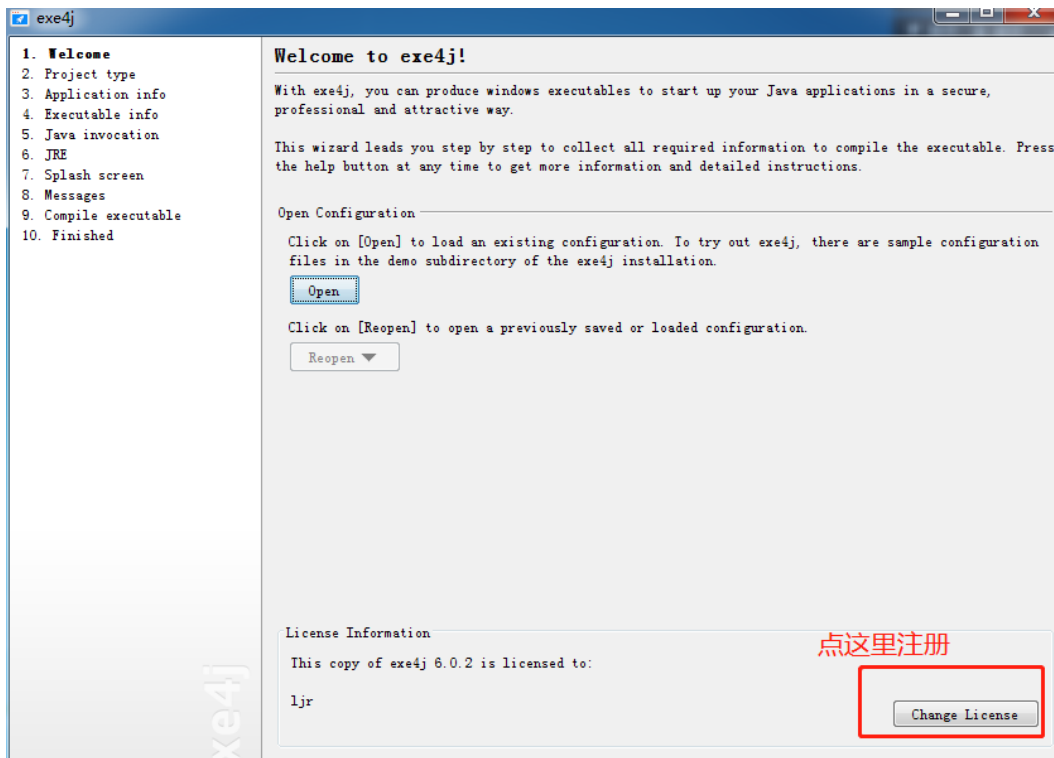
在将写好的Java程序转换为EXE程序前，需要先将Java程序导出为jar包，然后才能通过EXE4J生成想要的EXE程序。在这里，笔者使用的工具为eclipse，具体步骤如下：

1. 打开一个含有main方法且已调试可运行的Java工程，右键点击后选择export，再选择Java选项下的JAR file选项。



三、将jar文件转化为EXE文件

1. 打开exe4j Wizard.exe，在首页先进行License注册，否则生成的exe文件在运行时会提示：“this executable was create with an evaluation exe4j”。注册时可以使用如下的注册号：



A-XVK258563F-1p4lv7mg7sav

A-XVK209982F-1y0i3h4ywx2h1

A-XVK267351F-dpurrrhnyarva

A-XVK204432F-1kkoilo1jy2h3r

A-XVK246130F-1l7msieqiwnq

A-XVK249554F-pllh351kcke50

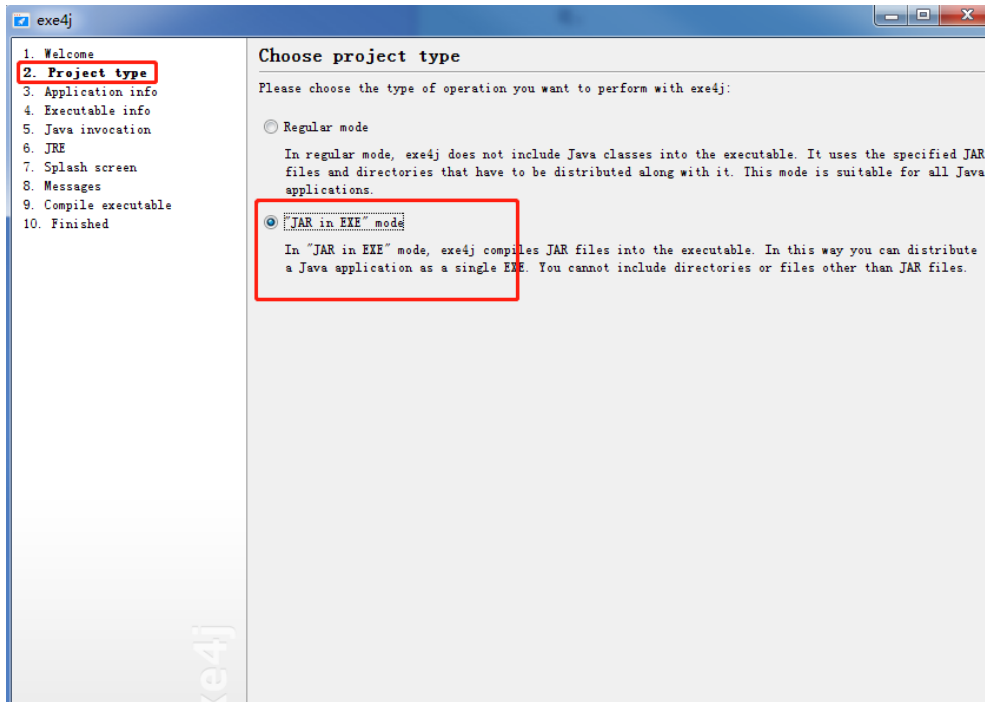
A-XVK238729F-25yn13iea25i

A-XVK222711F-134h5ta8yxbm0

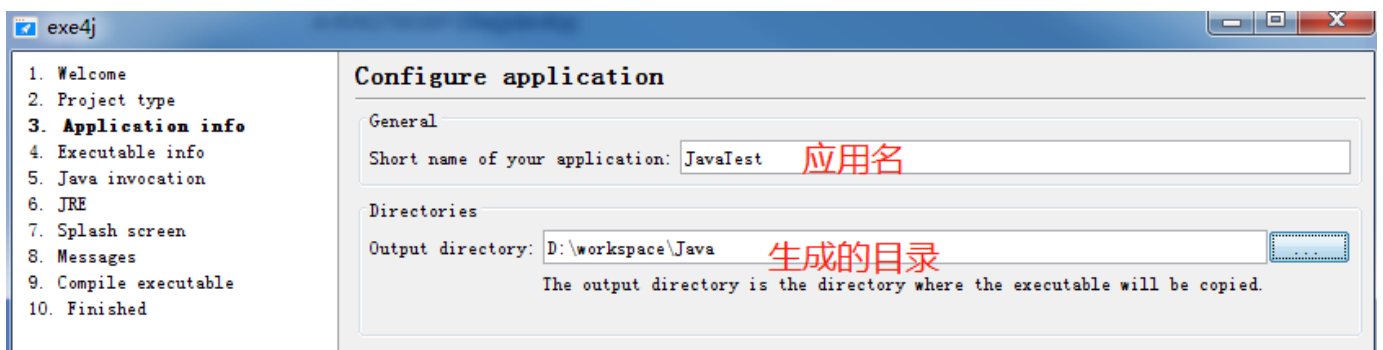
A-XVK275016F-15wjcbn4tpj

A-XVK275016F-15wjcbn4tpj

2. 点击next , 选择”JAR in EXE” mode。



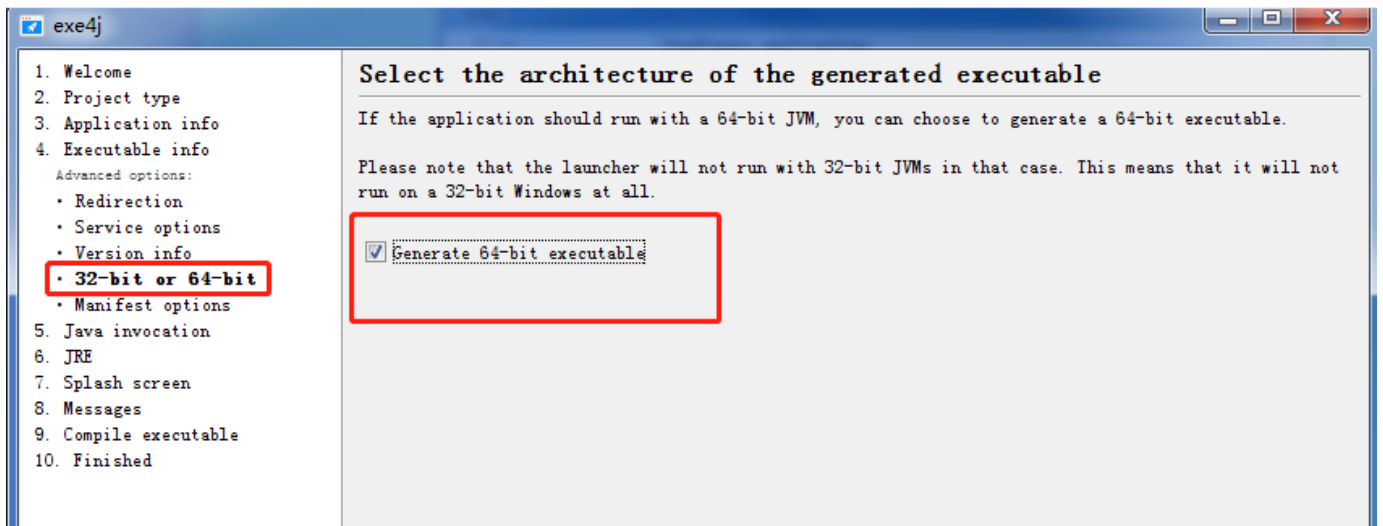
3. 点击next，填写生成的应用名和输出目录。



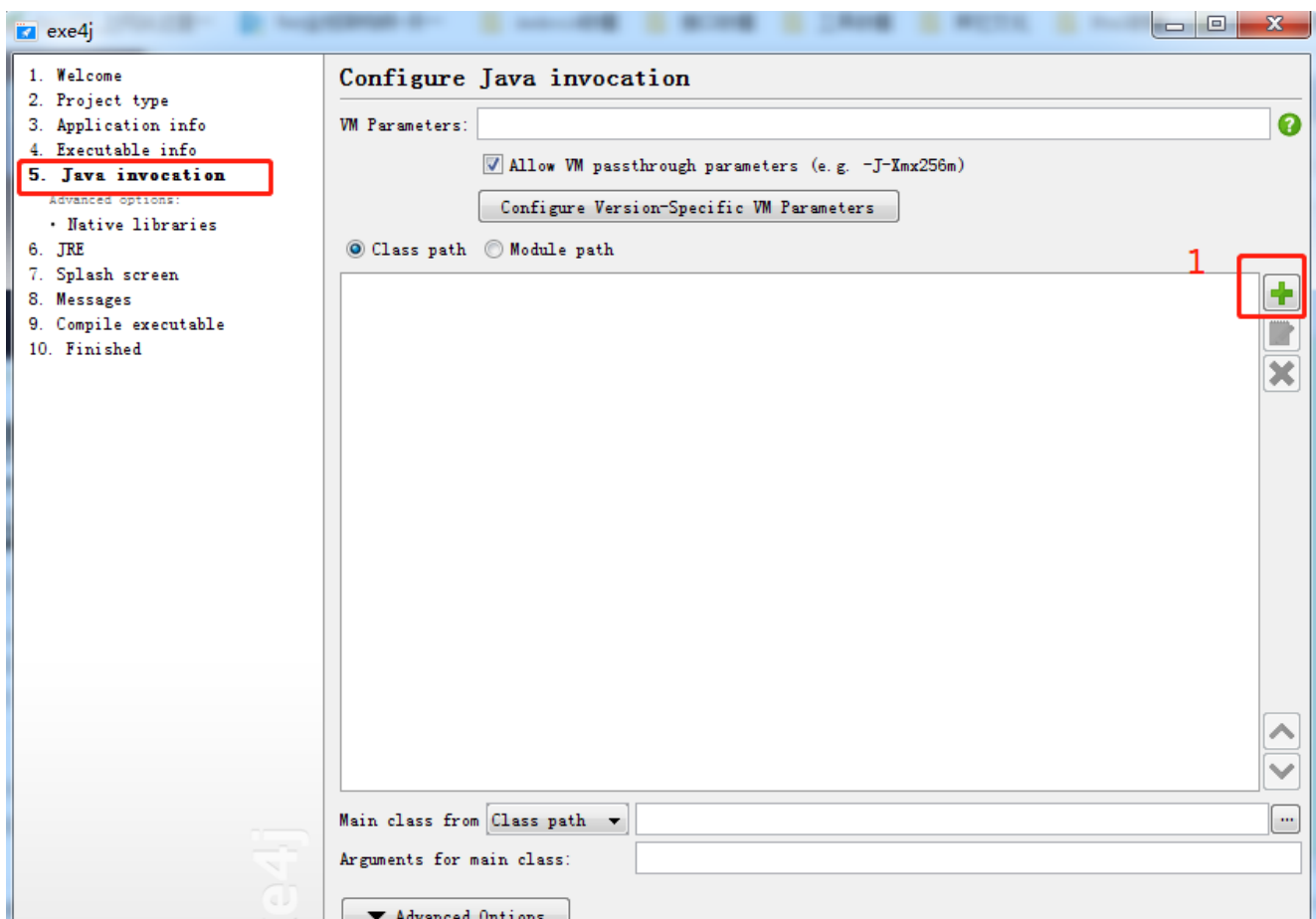
4. 点击next，根据自己的应用类别进行选择，并填入可执行文件的名称。其余选项可根据自己的需要进行设置。

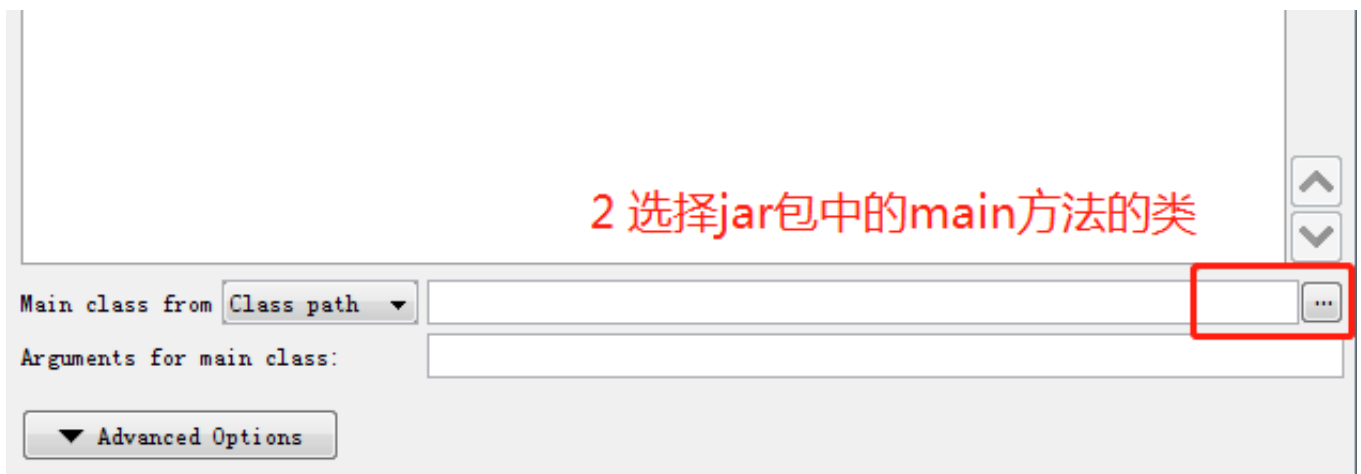
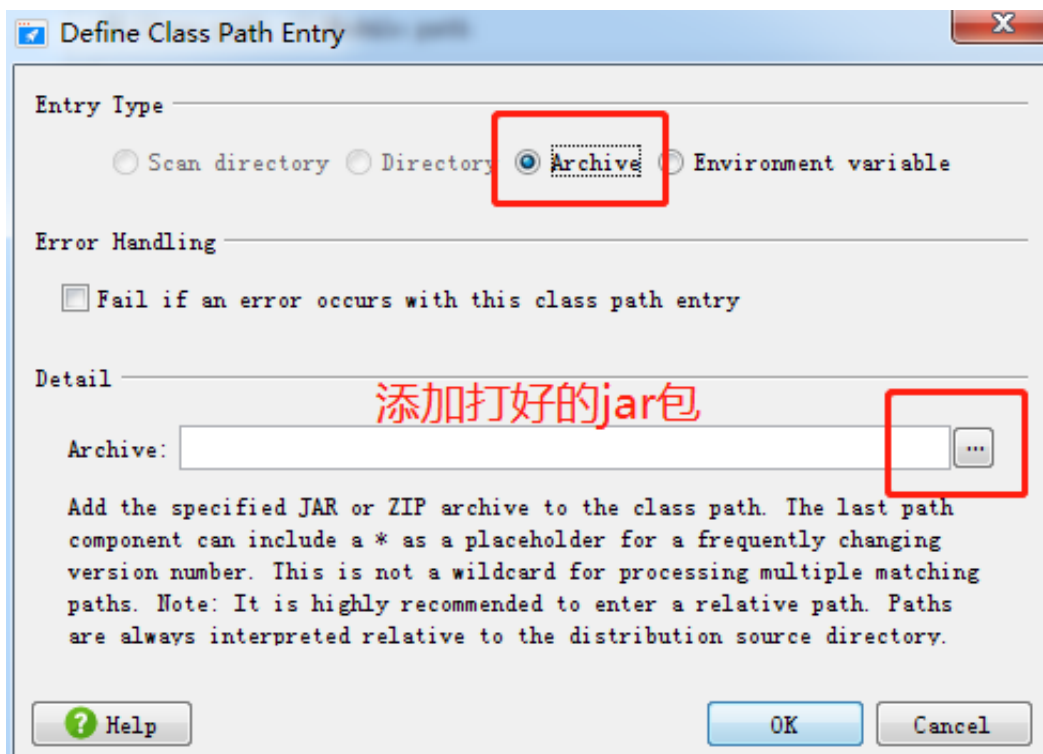


生成的EXE默认是在32位的JDK环境先运行。如果是目标计算机的运行环境为64位，则还应勾选如下选项：

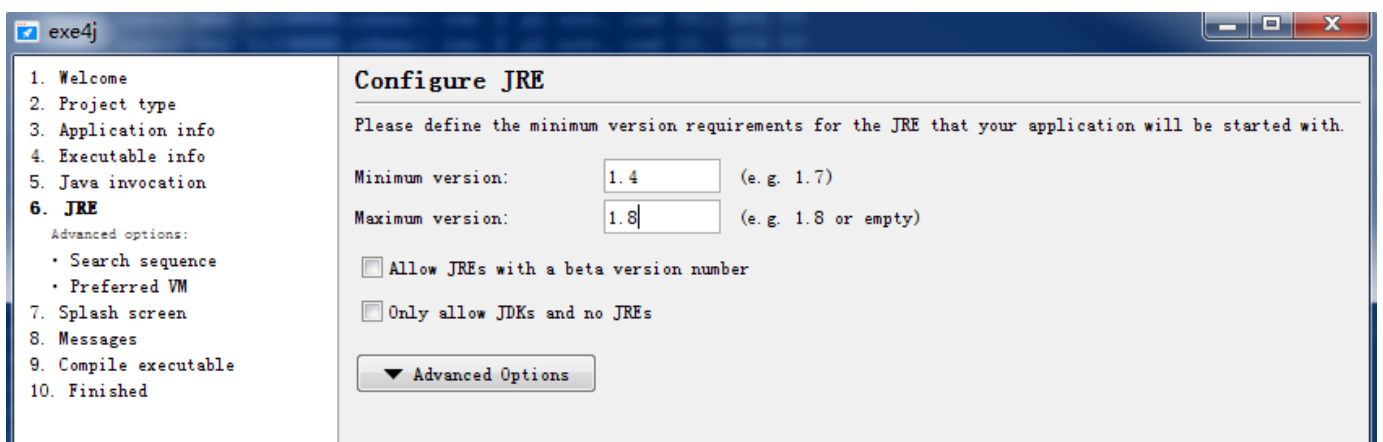


5. 点击next，添加需要生成EXE的jar包并设置好它的mainclass。





6. 根据自己应用对Java API的使用情况，设定运行时jre的最小版本和最大版本限定。



7. 接着设定生成EXE文件时所依赖的JDK。如果已将Java正确配置到系统环境变量中，该步骤一般可跳过。

8. 接着一路next，看到如下界面时说明已经将jar包转换为对应的exe文件。这时便可以打开

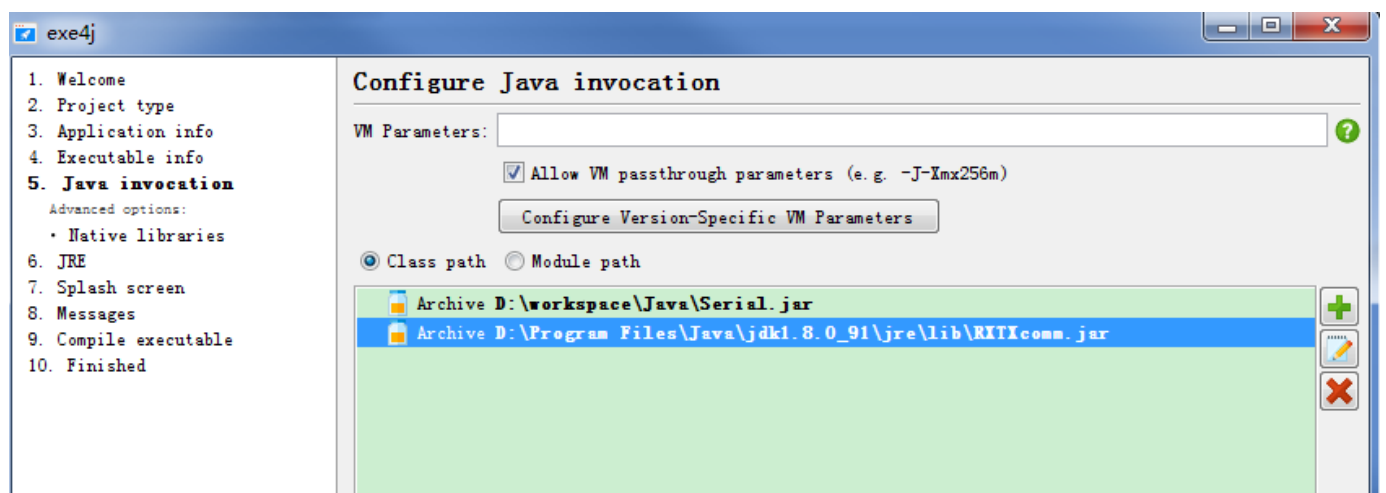
对应的exe文件看是否正常运行。

四、可能遇到的问题

1. 执行EXE文件时提示java.lang.NoClassDefFoundError

```
java.lang.NoClassDefFoundError: gnu/io/PortInUseException
    at serial.SerialUtility.sendCmdFromFile<SerialUtility.java:24>
    at serial.SerialUtility.main<SerialUtility.java:91>
    at sun.reflect.NativeMethodAccessorImpl.invoke0<Native Method>
    at sun.reflect.NativeMethodAccessorImpl.invoke<Unknown Source>
    at sun.reflect.DelegatingMethodAccessorImpl.invoke<Unknown Source>
    at java.lang.reflect.Method.invoke<Unknown Source>
    at com.exe4j.runtime.LauncherEngine.launch<LauncherEngine.java:81>
    at com.exe4j.runtime.WinLauncher.main<WinLauncher.java:94>
Caused by: java.lang.ClassNotFoundException: gnu.io.PortInUseException
    at java.net.URLClassLoader.findClass<Unknown Source>
    at java.lang.ClassLoader.loadClass<Unknown Source>
    at sun.misc.Launcher$AppClassLoader.loadClass<Unknown Source>
    at java.lang.ClassLoader.loadClass<Unknown Source>
    ... 8 more
```

这是由于你程序中有引用到第三方的jar包，而生成EXE文件时没有将其关联起来。在ConfigureJava invocation时，需要将引用到的第三方库也包含进来。



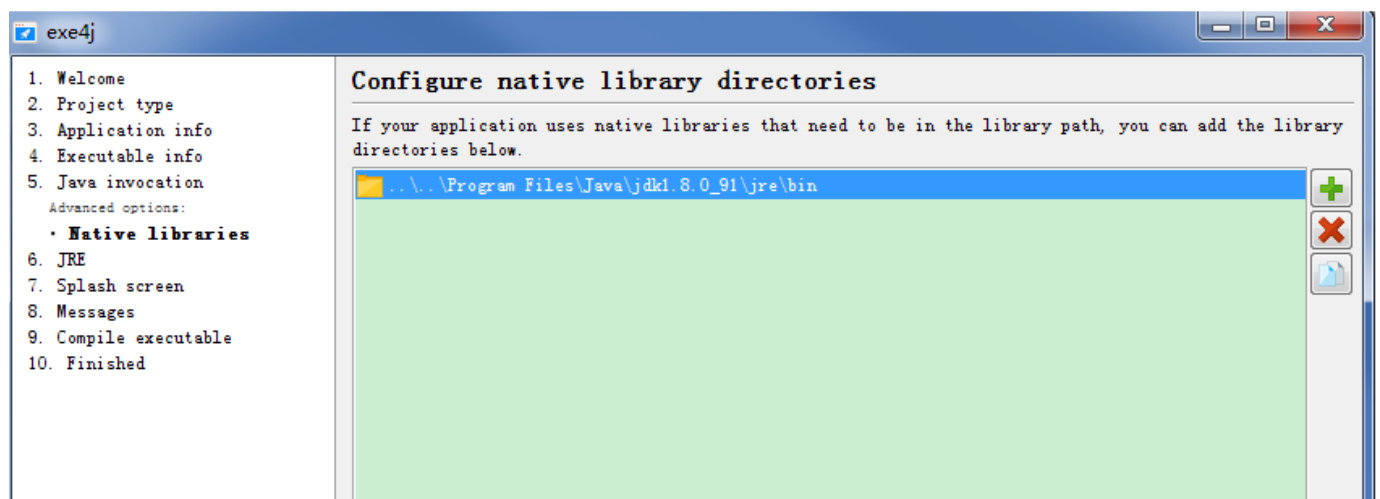
2. 执行EXE文件时提示java.lang.UnsatisfiedLinkError


```

java.lang.UnsatisfiedLinkError: no rxtxSerial in java.library.path thrown while
loading gnu.io.RXTXCommDriver
java.lang.UnsatisfiedLinkError: no rxtxSerial in java.library.path
.....
at java.lang.ClassLoader.loadLibrary(Unknown Source)
at java.lang.Runtime.loadLibrary0(Unknown Source)
at java.lang.System.loadLibrary(Unknown Source)
at gnu.io.CommPortIdentifier.<clinit>(CommPortIdentifier.java:123)
at serial.SerialBean.Initialize(SerialBean.java:44)
at serial.SerialUtility.sendCmdFromFile(SerialUtility.java:26)
at serial.SerialUtility.main(SerialUtility.java:91)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(Unknown Source)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(Unknown Source)
at java.lang.reflect.Method.invoke(Unknown Source)
at com.exe4j.runtime.LauncherEngine.launch(LauncherEngine.java:81)
at com.exe4j.runtime.WinLauncher.main(WinLauncher.java:94)

```

这是由于程序中引用到的jar包中，有通过JNI机制去调用到一些dll文件中的方法，而在配置生成EXE的过程中没有指定。在Configure Java invocation时，需要在native libraries中添加dll所在的目录。



3. 执行EXE文件时提示java.lang.UnsupportedClassVersionError

```

java.lang.UnsupportedClassVersionError: serial/SerialUtility : Unsupported major.minor version 52.0
.....
at java.lang.ClassLoader.defineClass1(Native Method)
at java.lang.ClassLoader.defineClass(Unknown Source)
at java.security.SecureClassLoader.defineClass(Unknown Source)
at java.net.URLClassLoader.defineClass(Unknown Source)
at java.net.URLClassLoader.access$100(Unknown Source)
at java.net.URLClassLoader$1.run(Unknown Source)
at java.net.URLClassLoader$1.run(Unknown Source)
at java.security.AccessController.doPrivileged(Native Method)
at java.net.URLClassLoader.findClass(Unknown Source)
at java.lang.ClassLoader.loadClass(Unknown Source)
at sun.misc.Launcher$AppClassLoader.loadClass(Unknown Source)
at java.lang.ClassLoader.loadClass(Unknown Source)
at com.exe4j.runtime.LauncherEngine.launch(LauncherEngine.java:77)
at com.exe4j.runtime.WinLauncher.main(WinLauncher.java:94)

```

这个一般是由于运行时JDK版本与开发的JDK版本不一致导致（比如你开发时用的是64位的JDK，而在其它电脑上运行时，它的JDK缺失32位的），两边的JDK位数一致就行。

