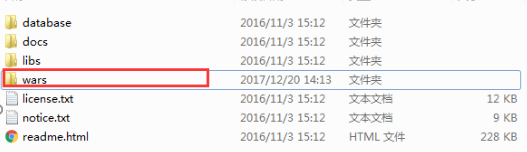
**Activiti 使用教程**

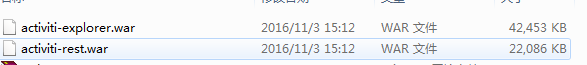
# 一、下载

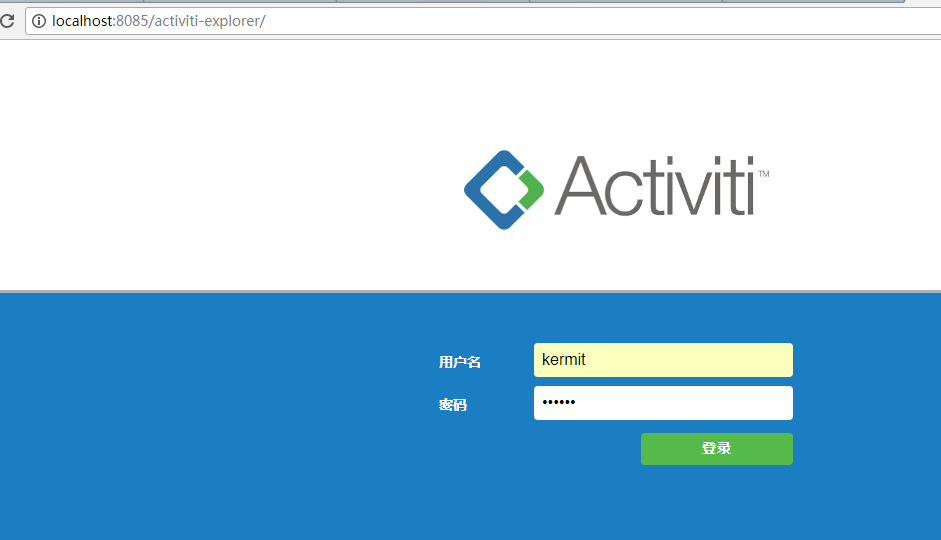
<https://www.activiti.org/download-bpm>

国内网站可能进入比较慢。

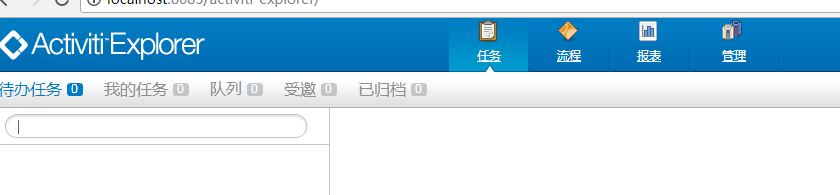
官网上可以下演示版本，war。直接部署到tomcat 可以查看。







用户名和密码都是kermit



可以上github 下载

<https://github.com/Activiti/Activiti-Designer/releases>

# 二、集成activiti

eclispe 和 idea 都可以集成。Myeclipse 集成比较麻烦。这里采用了idea 。

## Eclipse集成

### 在线安装

打开Eclipse windows 🡪 install new software

**http://activiti.org/designer/update/**

等待安装完成即可。

如果出现错误，网上下载一下几个jar 放到Eclipse的安装目录的plugins目录下即可。离线安装报错使用同样的方法。

可以在 <http://mvnrepository.com/> 下载这几个jar

org.eclipse.emf.transaction-1.4.0.201306111400.jar

org.eclipse.emf.validation-1.7.0.201306111341.jar

org.eclipse.emf.workspace-1.5.1.201405281451.jar

### 离线安装

首先将

org.eclipse.emf.transaction-1.4.0.201306111400.jar

org.eclipse.emf.validation-1.7.0.201306111341.jar

org.eclipse.emf.workspace-1.5.1.201405281451.jar

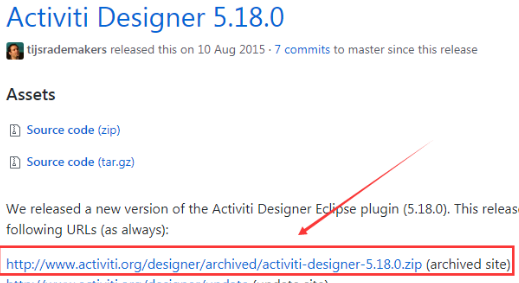
这三个jar 放到 plugins目录下。

网上搜一个 离线包。

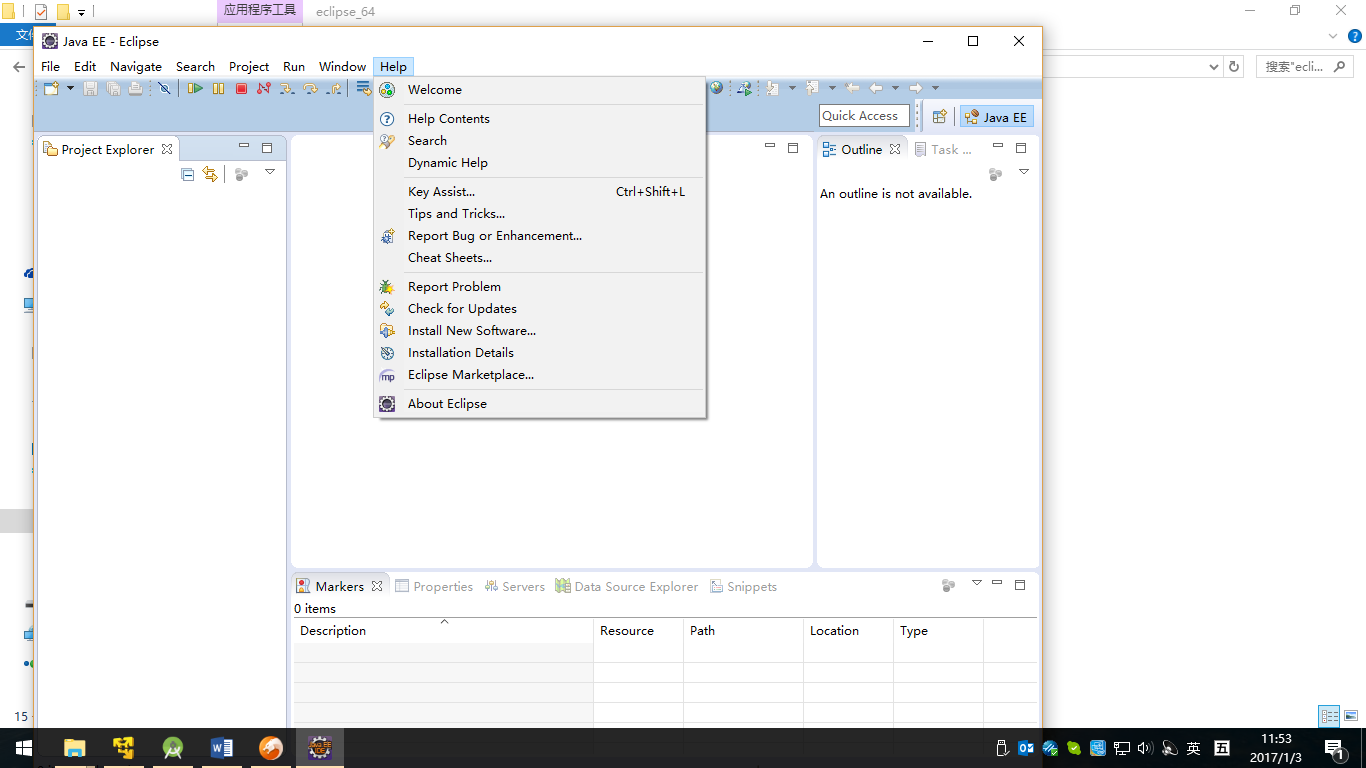
可以在git上下。

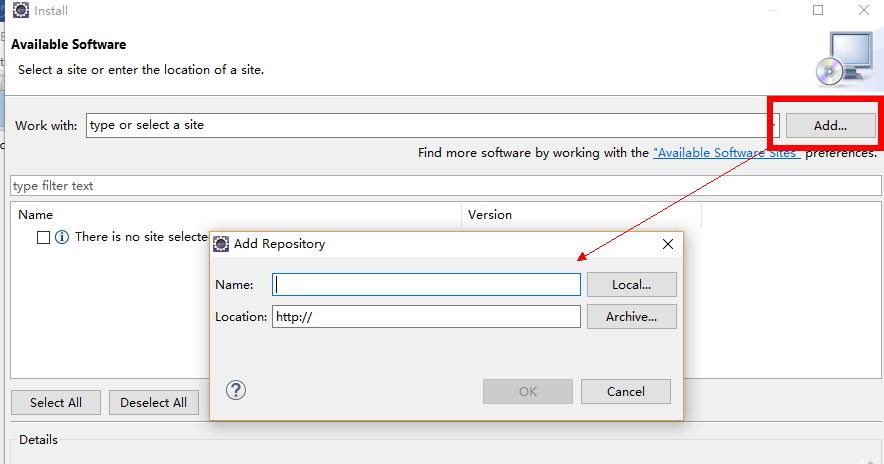
<https://github.com/Activiti/Activiti-Designer/releases>

<http://www.activiti.org/designer/archived/activiti-designer-5.18.0.zip>

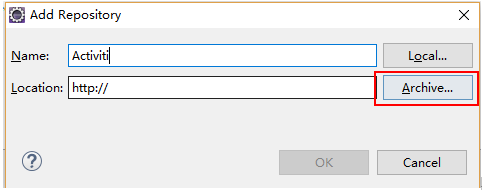


打开Eclipse windows 🡪 install new software

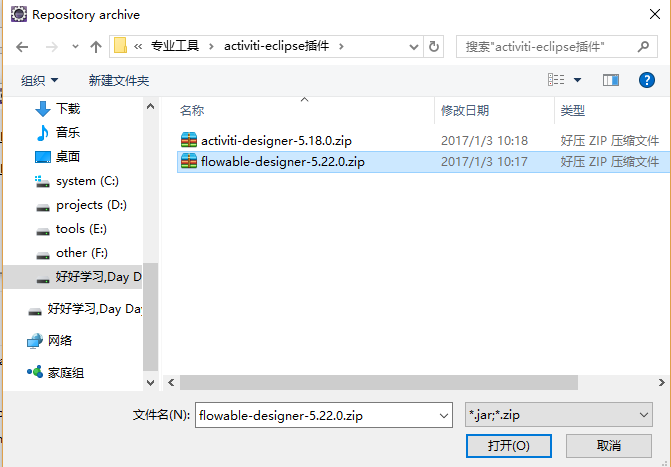


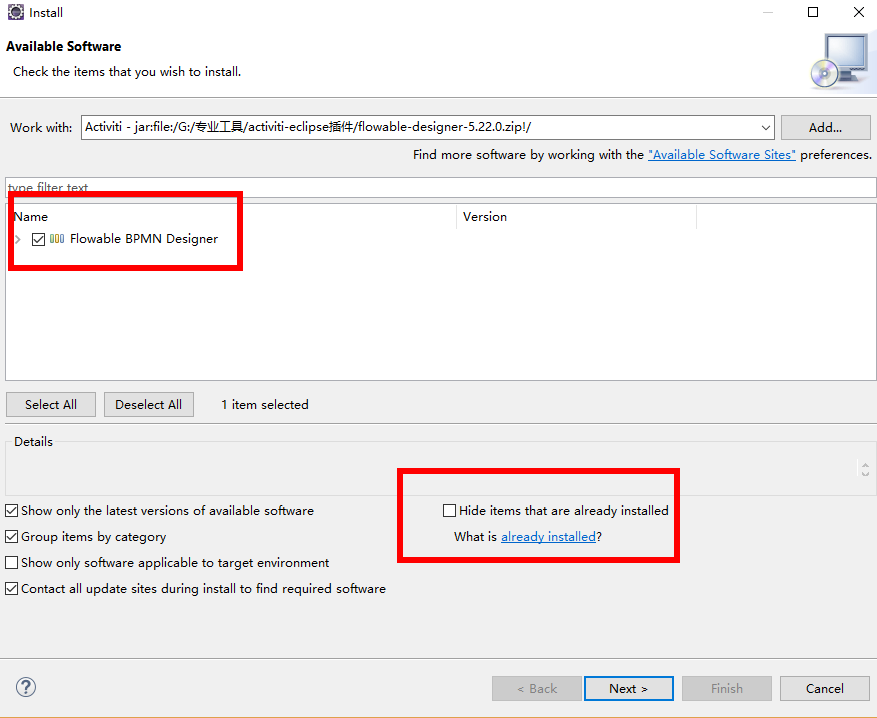


选择安装文件



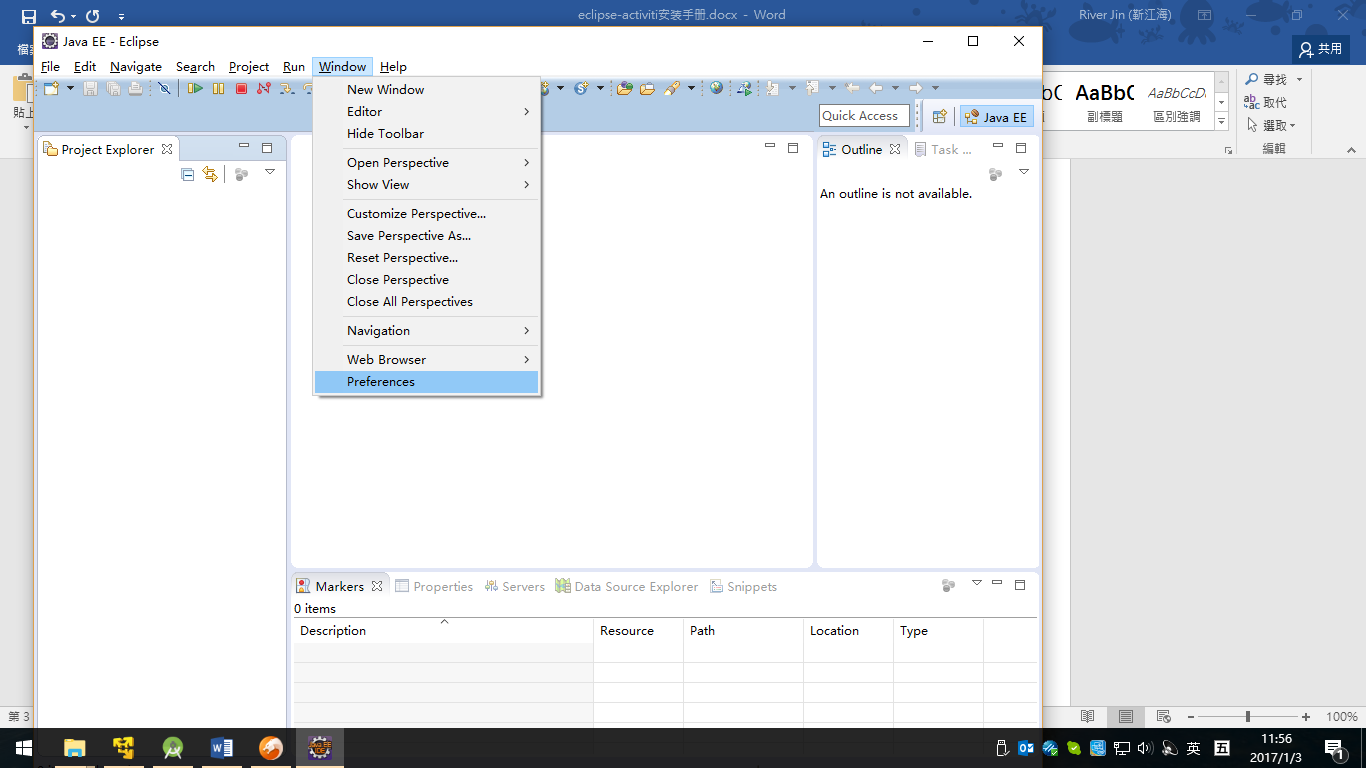
需要安装的版本





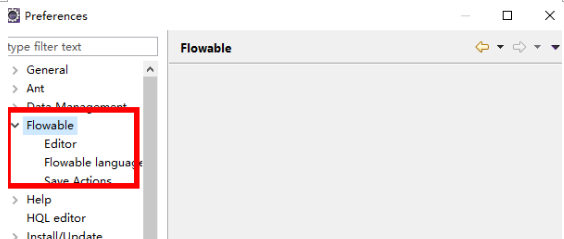
接下来一路next，finish;

安装完成重启eclipse打开

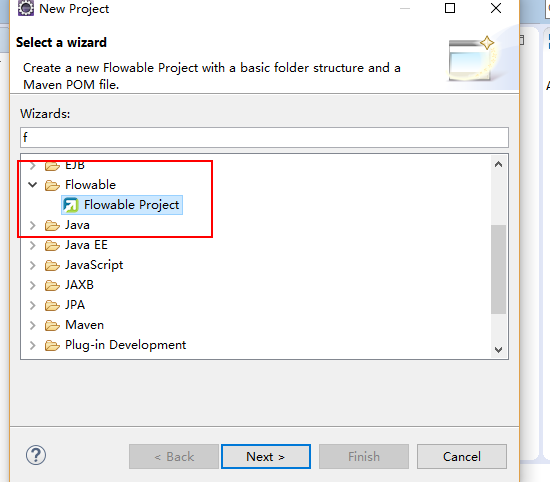


如果选择的是5.22以前的版本会出现一个Activiti…

如果是5.22及以后的版本会出现如下图



在新建文件的时候找到other 如图:



表示安装成功

## Myeclispe 集成

下载 <https://github.com/Activiti/Activiti-Designer/releases>

<http://www.activiti.org/designer/archived/activiti-designer-5.18.0.zip>

2. 下载后是一个zip包，当前最新的版本为5.18.0 名称为 activiti-designer-5.18.0.zip，为了方便管理，在MyEclipse的安装目录的dropins目录下创建一个目录，名为 “activiti-designer-5.18.0”(仅仅是为了区别版本和方便管理，你可以可以改为其他名称) ，然后将压缩包中的东西解压在此。比如我解压的目录就是C:\Soft\MyEclipse2014\dropins\activiti-designer-5.18.0

3. 然后在MyEclipse的安装目录下找到dropins目录，再创建一个 “activiti.link”文件，在其中输入：

path=C:\Soft\MyEclipse2014\dropins\activiti-designer-5.18.0\site\

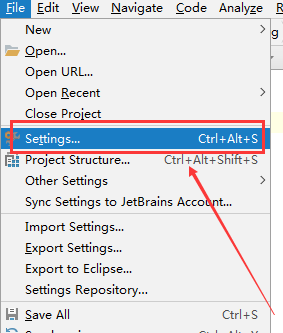
path指向你的解压后的那几个文件存放目录。

4. 然后重启你的MyEclipse，如果Windows→Preference→Activiti 存在，表示安装成功。

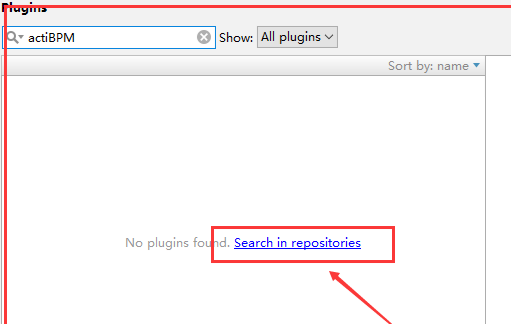
安装完成重启eclipse打开

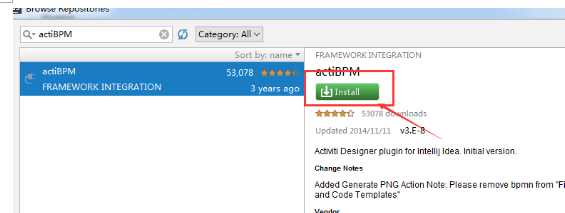
## idea集成

File 🡪 Settings

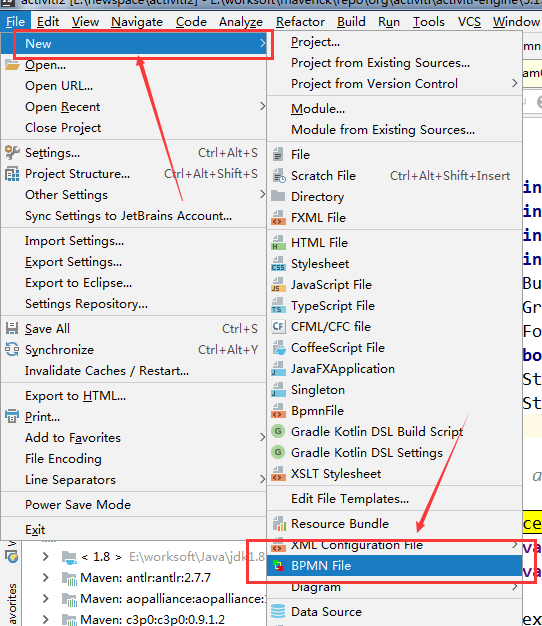


Plugins 下搜索。actiBPM 安装即可





重新启动idea、出现如图说明安装成功



# 三、简单的使用activiti

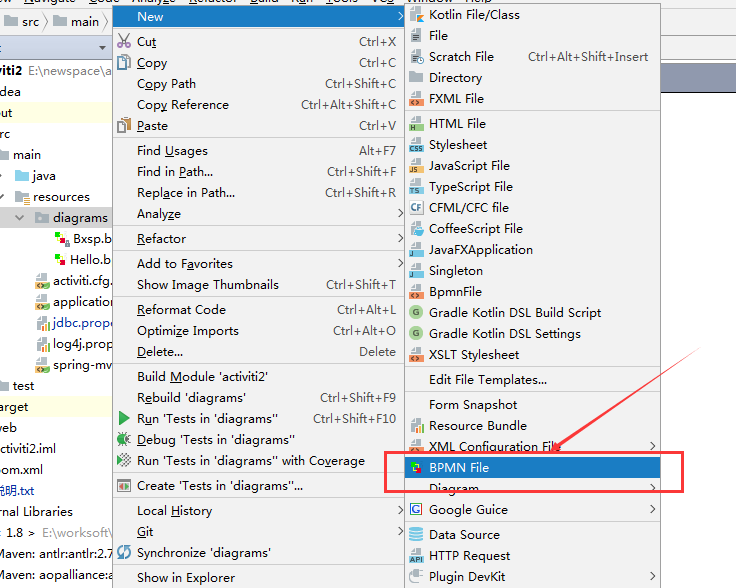
## 1、前置准备

项目放到git 上：<https://github.com/zhang925/activiti2>

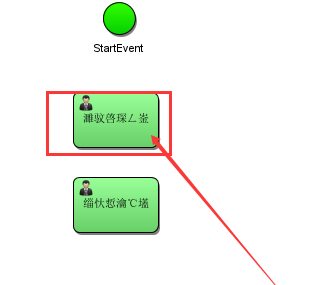
这里假设，已经配置好了数据库和项目。从画流程开始。

这里使用的是Idea

建议使用MyEclipse 或者 eclipse 比较方便，根据跟人习惯使用。



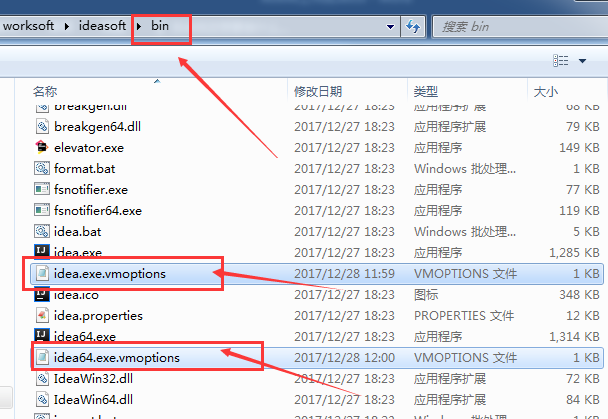
解决乱码方案：

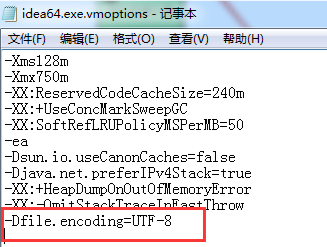


在idea安装目录下的bin目录下，找：

idea.exe.vmoptions和idea64.exe.vmoptions两个文件,打开后追加：

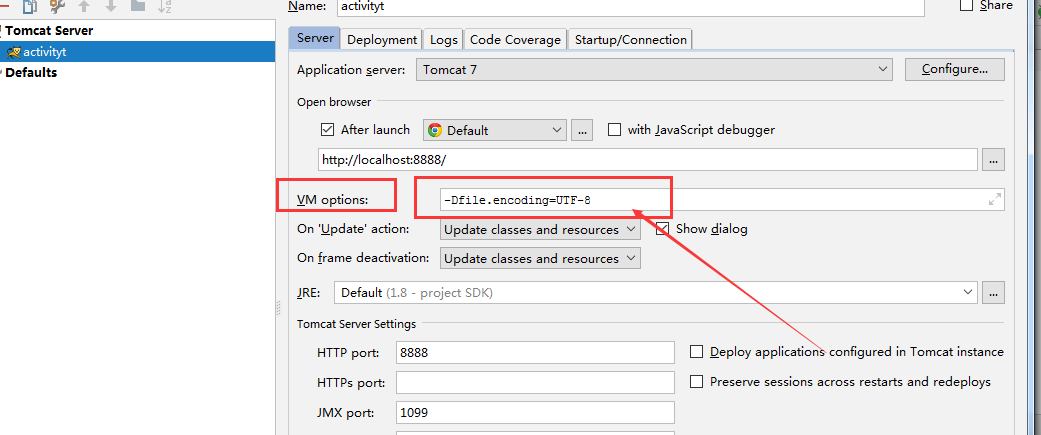
-Dfile.encoding=UTF-8





如果控制台输入乱码，vm options加入这个即可解决。

-Dfile.encoding=UTF-8



如果部署流程报错：

3 字节的 UTF-8 序列的字节 3 无效

Pom 文件添加：

<build>

<resources>

<resource>

<directory>src/main/resources</directory>

<filtering>true</filtering>

</resource>

</resources>

<plugins>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-resources-plugin</artifactId>

<configuration>

<encoding>UTF-8</encoding>

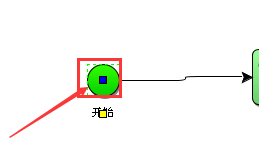
</configuration>

</plugin>

</plugins>

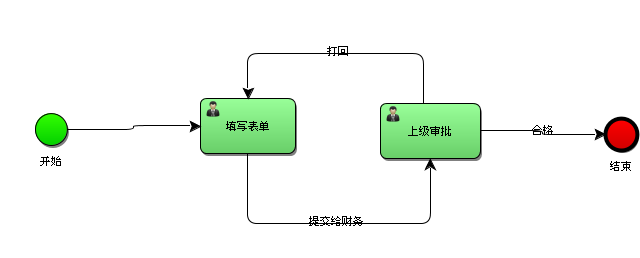
</build>

## 2、画流程



拖着中间的那个点，向外拉，就能画直线了。

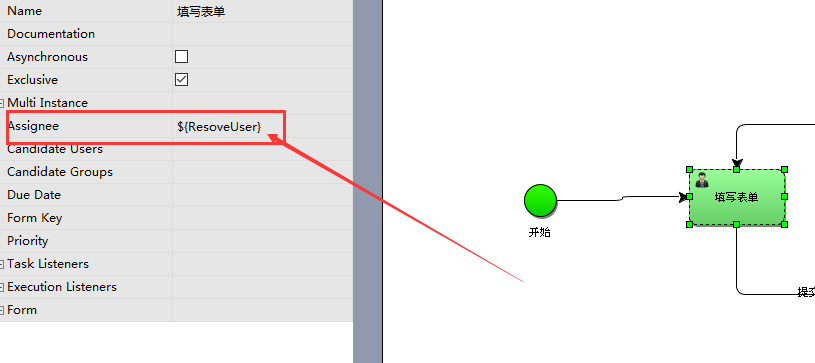
这里画了一个简单的报销流程。



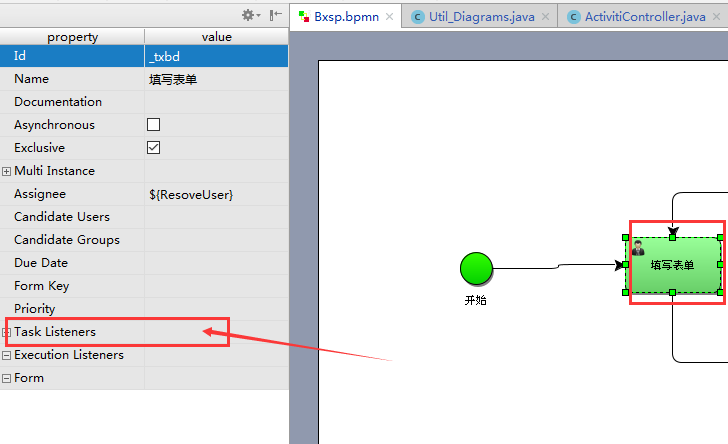
## 3. 绑定人员

### 3.1、方法一

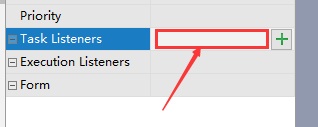
点击某个节点。 Assignee 绑定 处理人 。如果是${参数名}的方式，怎表示不固定处理人。如果说具体的名字则是固定处理人。这里我们命名${TXBD}



### 3.2、方法二

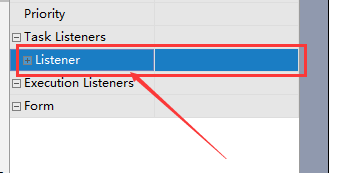


先点击 一下图中红色框所标注的地方，看到出现一个 + 在点击一次加号。

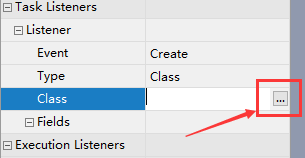


展开 Task listener 左侧的加号，注意是左侧。不是右边的那个。

下方出现一个listener



展开Listener



点击右侧的 省略号 绑定 java 文件即可

如果设置了监听类就不 需要在流程xml中设置Assignee的值

public class Util\_TaskListener implements TaskListener {

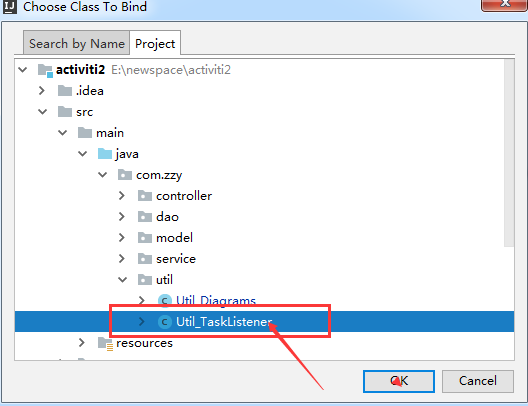
public void notify(DelegateTask delegateTask) {

//这里可以根据登录人 设置下一步 审批人 是谁

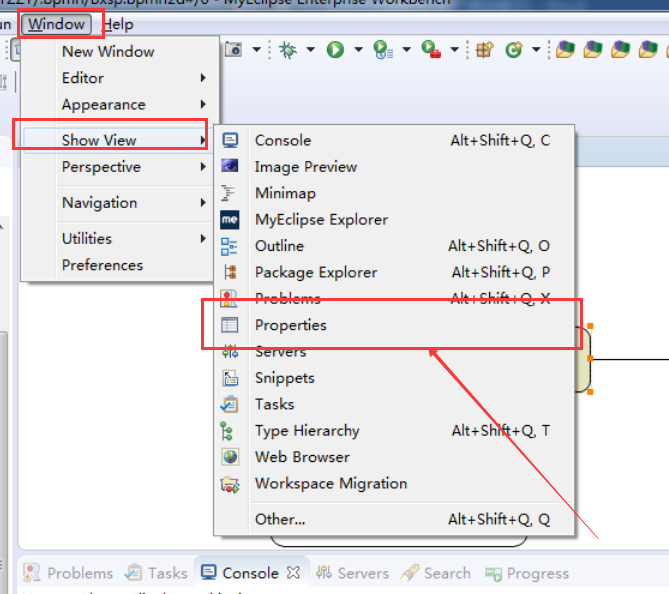
delegateTask.setAssignee("审批人");

}

}



### 3.3、myEclipse 方式：



首先去看数据库表：act\_ru\_identitylink

task\_id 对应act\_ru\_task中的主键id\_

type\_:对应类别有starter流程实例的发起者、candidate：节点的候选人

group\_id:候选组的id主键，

user\_id:对应人的id\_主键

rev\_:版本号

其中group\_id、user\_id不一定是activiti工作流引擎中人员组织关系，有可能是我们自定义的人员架构关系，这一点重点说明一下，因为后面的文章，我们会讲解如何扩展工作流人员架构

## 清除所有数据

### 清除流程数据

DELETE FROM act\_ge\_bytearray;

DELETE FROM act\_hi\_actinst;

DELETE FROM act\_hi\_identitylink;

DELETE FROM act\_hi\_procinst;

DELETE FROM act\_hi\_taskinst;

DELETE FROM act\_re\_deployment;

DELETE FROM act\_ru\_identitylink;

DELETE FROM act\_ru\_job;

DELETE FROM act\_ru\_task;

DELETE FROM act\_ru\_execution;

DELETE FROM act\_re\_procdef;

### 清除附件

DELETE FROM act\_hi\_attachment;

# 四、其他

Window下 ，其他系统自行搜索；font-family 中文格式

微软雅黑 对应Microsoft YaHei

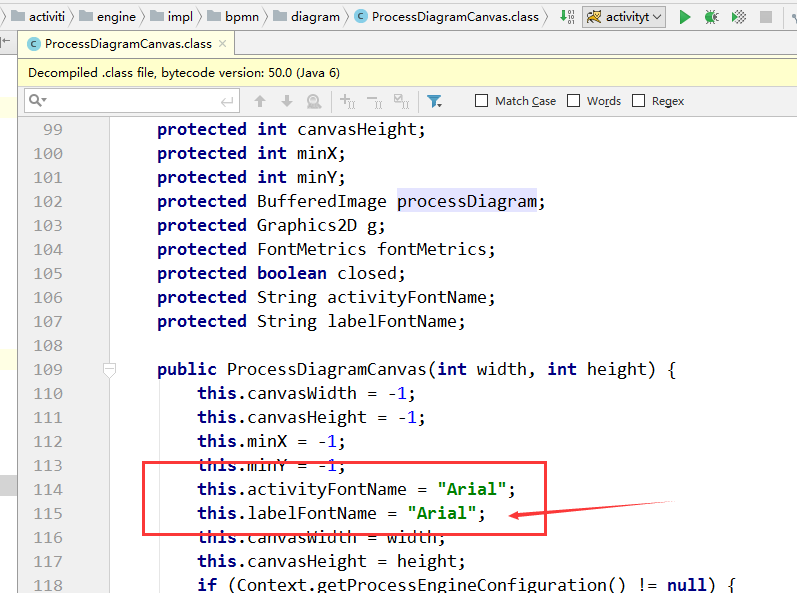
宋体 对应 SimSun

activiti-engine-5.15.1-sources.jar

ProcessDiagramCanvas

**public** ProcessDiagramCanvas(**int** width, **int** height)

**this**.activityFontName = **"Arial"**;  
**this**.labelFontName = **"Arial"**;



终极解决方法：

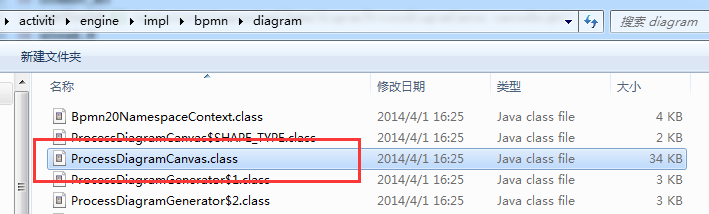
下载: jclasslib

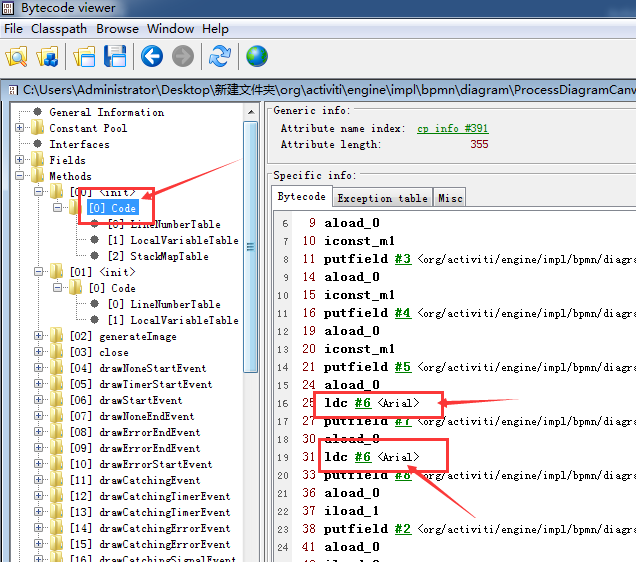
把自己的maven仓库中的

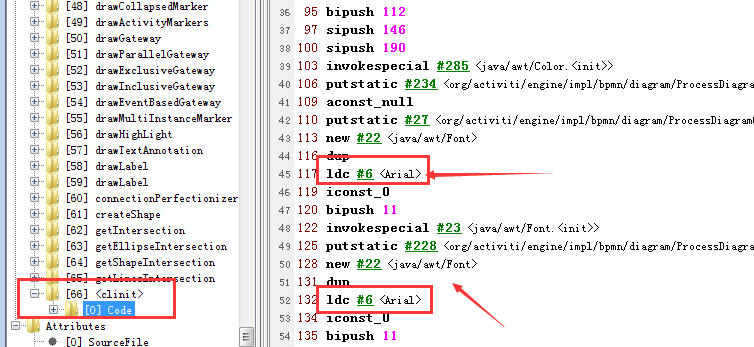
activiti-engine-5.15.1.jar

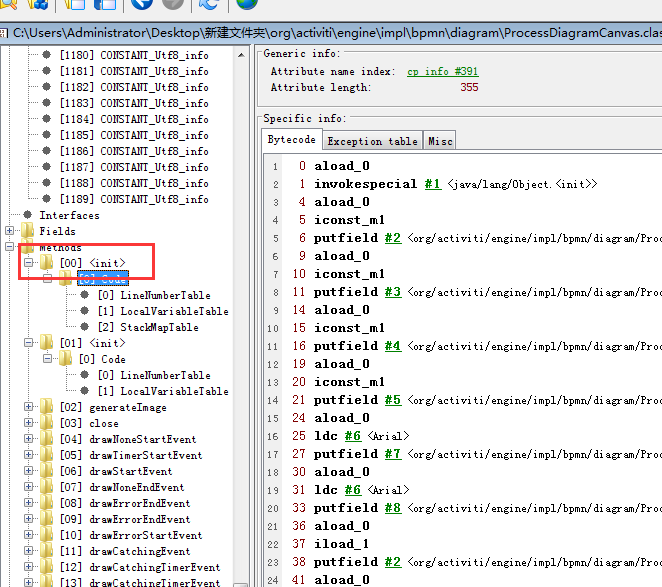
解压。

用 jclasslib 打开 找到

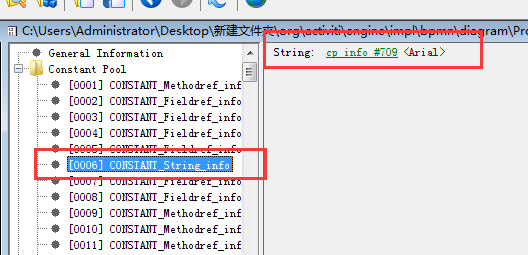








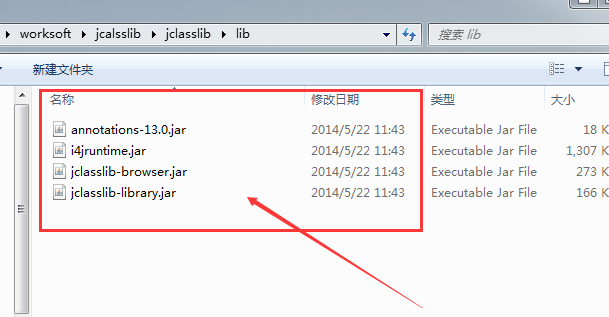
点击第 25行后面的 #6 会跳转到Constant Pool常量池的第0006个常量

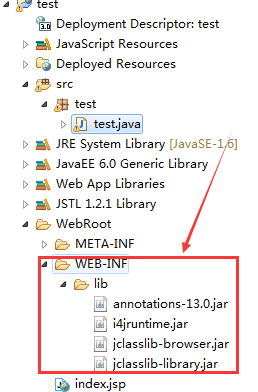


点击 右侧的#709 会跳转到 Constant Pool 的第 709 个常量

这里我们要做的据是修改第 709 的值

建立一个java类





import java.io.\*;

import org.gjt.jclasslib.io.ClassFileWriter;

import org.gjt.jclasslib.structures.CPInfo;

import org.gjt.jclasslib.structures.ClassFile;

import org.gjt.jclasslib.structures.constants.ConstantUtf8Info;

public class test {

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

String filePath = "C:\\ProcessDiagramCanvas.class";

FileInputStream fis = new FileInputStream(filePath);

DataInput di = new DataInputStream(fis);

ClassFile cf = new ClassFile();

cf.read(di);

CPInfo[] infos = cf.getConstantPool();

int count = infos.length;

for (int i = 0; i < count; i++) {

if (infos[i] != null) {

System.out.print(i);

System.out.print(" = ");

System.out.print(infos[i].getVerbose());

System.out.print(" = ");

System.out.println(infos[i].getTagVerbose());

if(i == 362){

ConstantUtf8Info uInfo = (ConstantUtf8Info)infos[i];

uInfo.setBytes("芝麻不开门!".getBytes());

infos[i]=uInfo;

}

}

}

cf.setConstantPool(infos);

fis.close();

File f = new File(filePath);

ClassFileWriter.writeToFile(f, cf);

}

}

