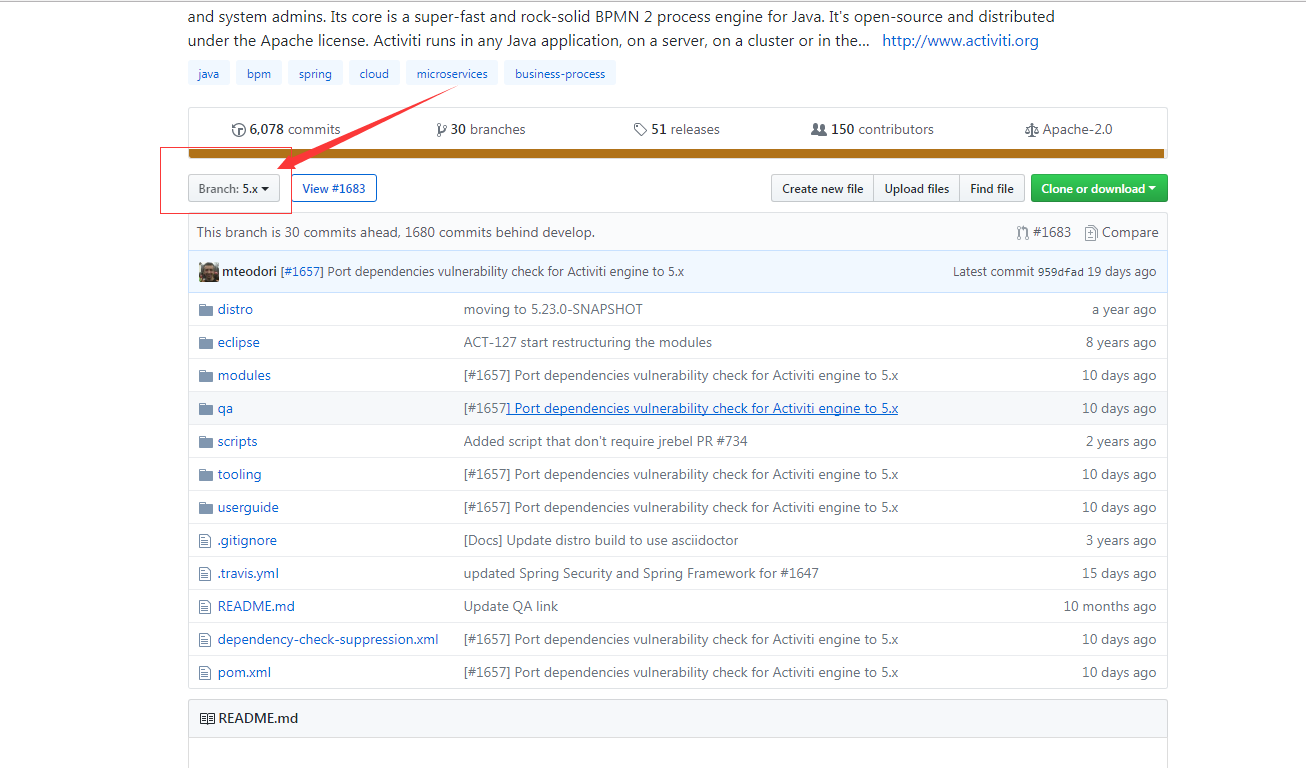
## 以下所有都是要建立在ssm+activiti已经能运行起来:spring至少要4要不然不支持@RestController

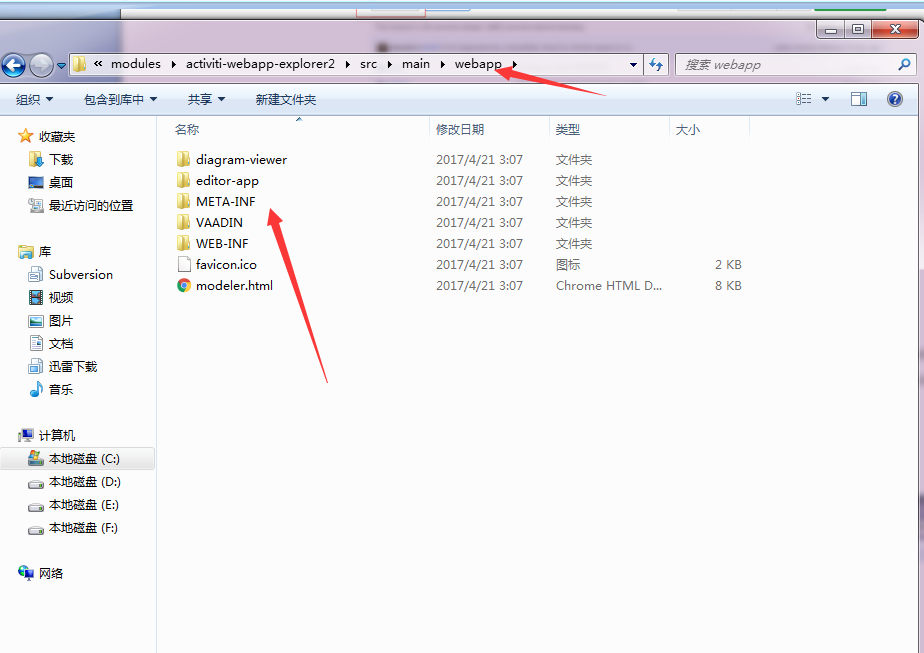
首先需要到github：<https://github.com/Activiti/Activiti>下载源码包：可以本地git或者下载zip包



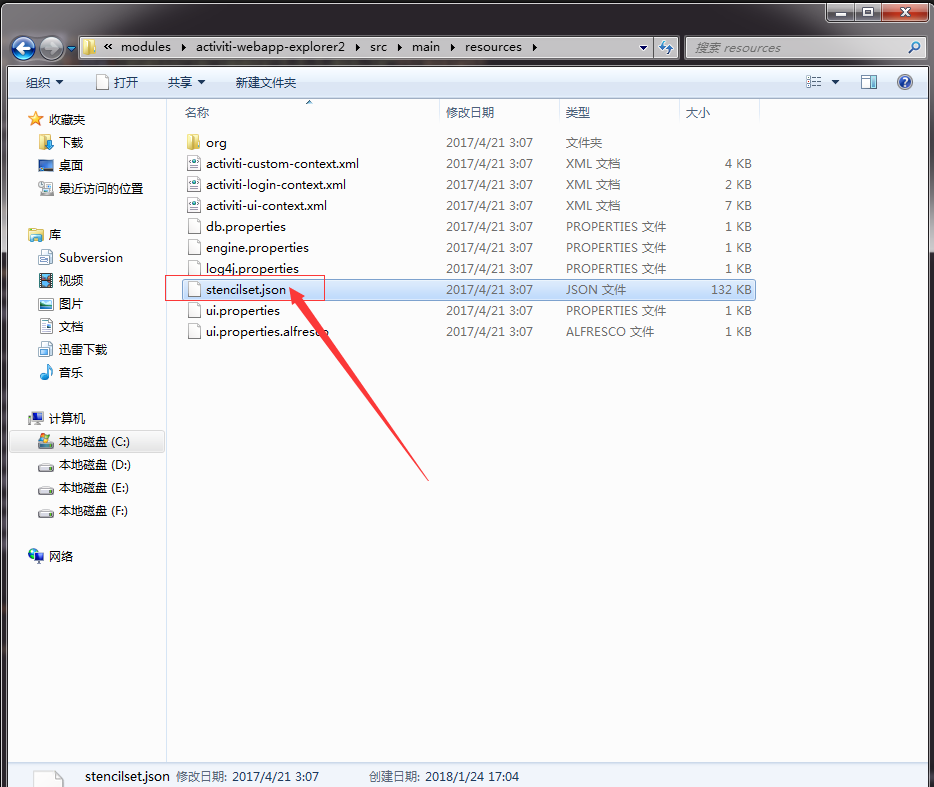
一、获得前端界面显示所需要的：

1、解压Activiti-master.zip

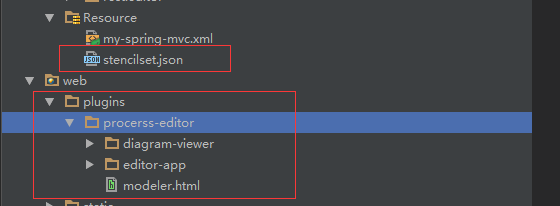
2、找到modules\activiti-webapp-explorer2\src\main\webapp

diagram-viewer、editor-app、modeler.html，将其加到项目的webapp中。

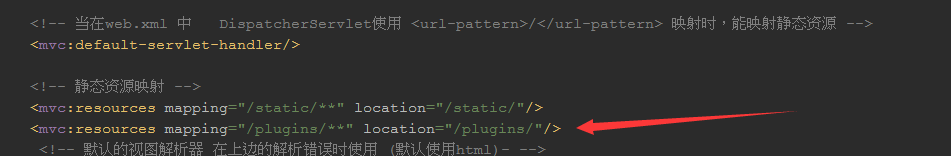
从activiti-webapp-explorer2 的 main/resource下找到stencilset.json放到项目的配置文件中



拷到项目web目录中：idea和eclipse名称也许不一样：我放到了plugins目录下

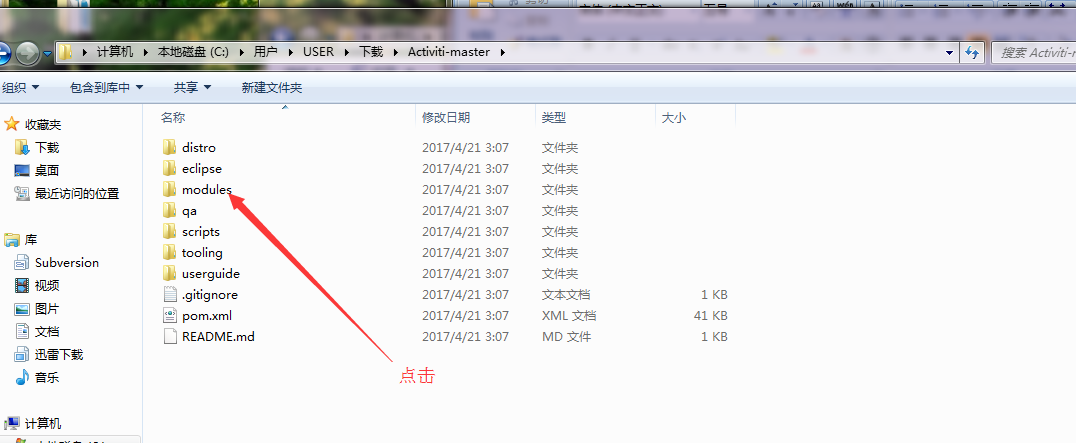


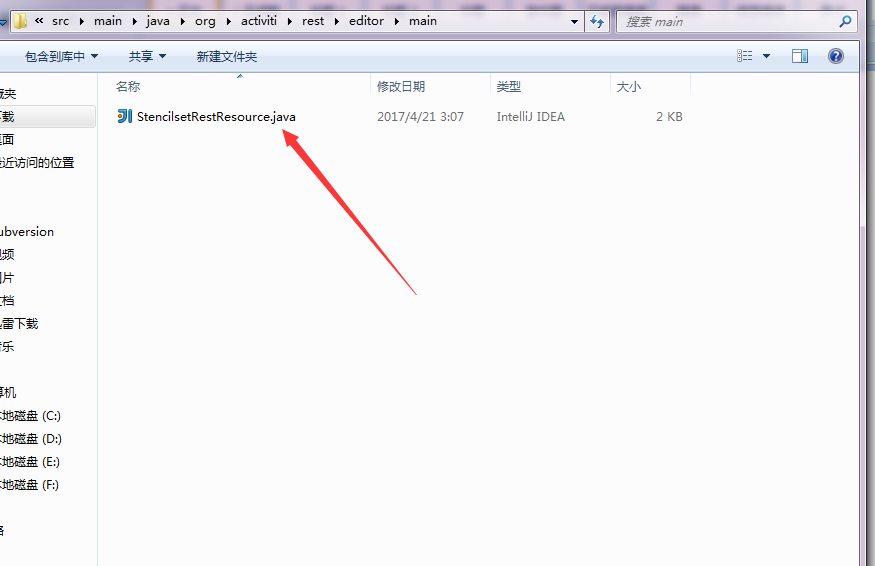
注意：如果springmvc配置的是/，那么需要开启不拦截该目录：



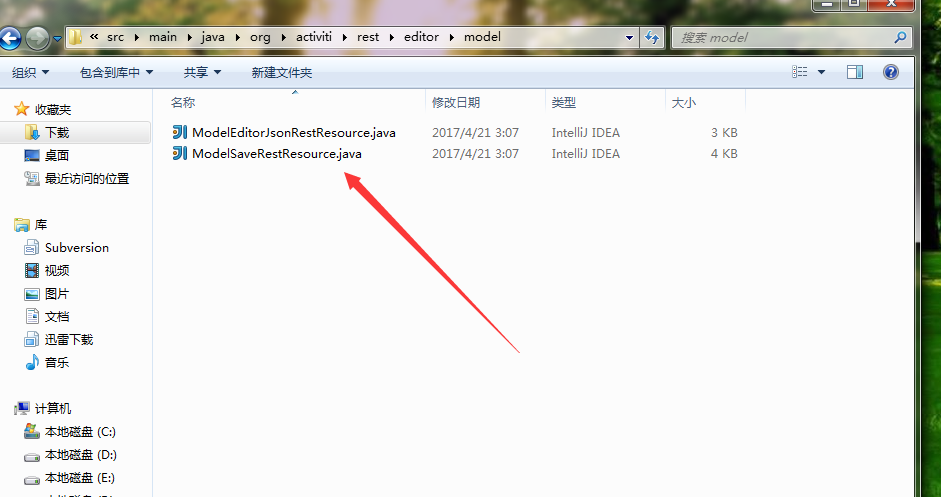
## 二、获取到运行所需要的controller

在Activiti-master下的modules文件夹下有一个activiti-modeler文件

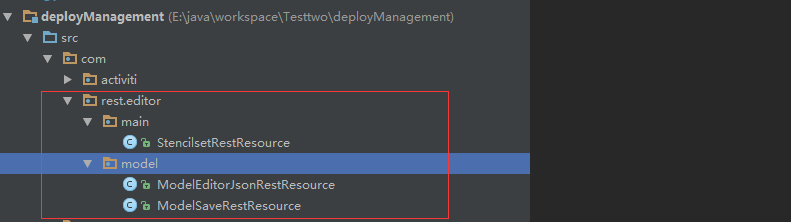


->activiti-modeler\src\main\java\org\activiti\rest\editor\main

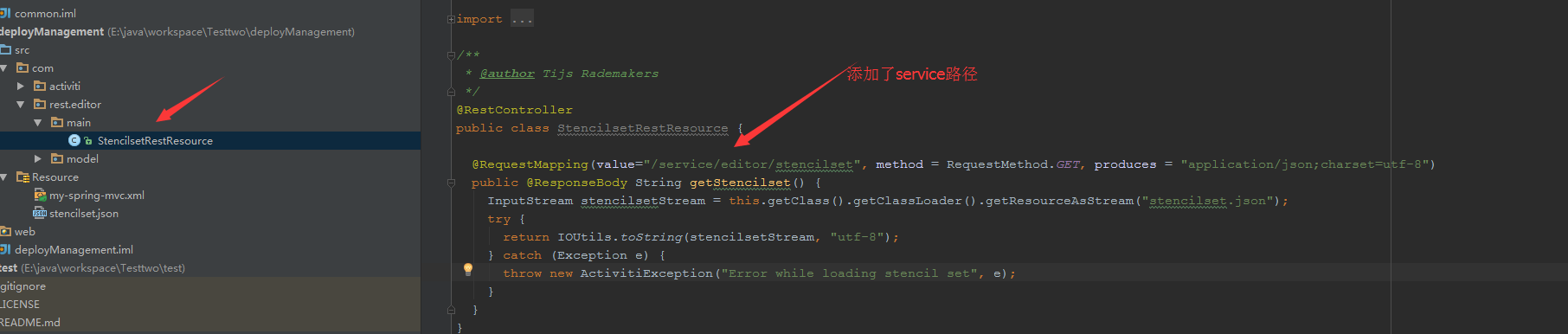
->activiti-modeler\src\main\java\org\activiti\rest\editor\model



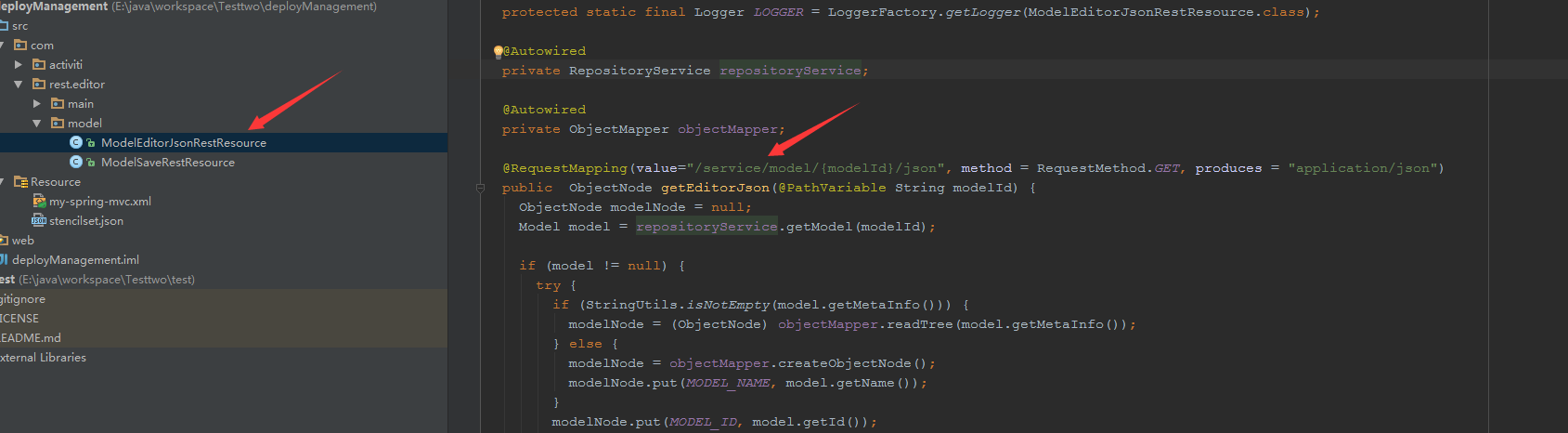
在我们项目下创建com.rest.editor.main来放StencilsetRestResource.java;com.rest.editor.model: ModelEditorJsonRestResource.java 和ModelSaveRestResource.java



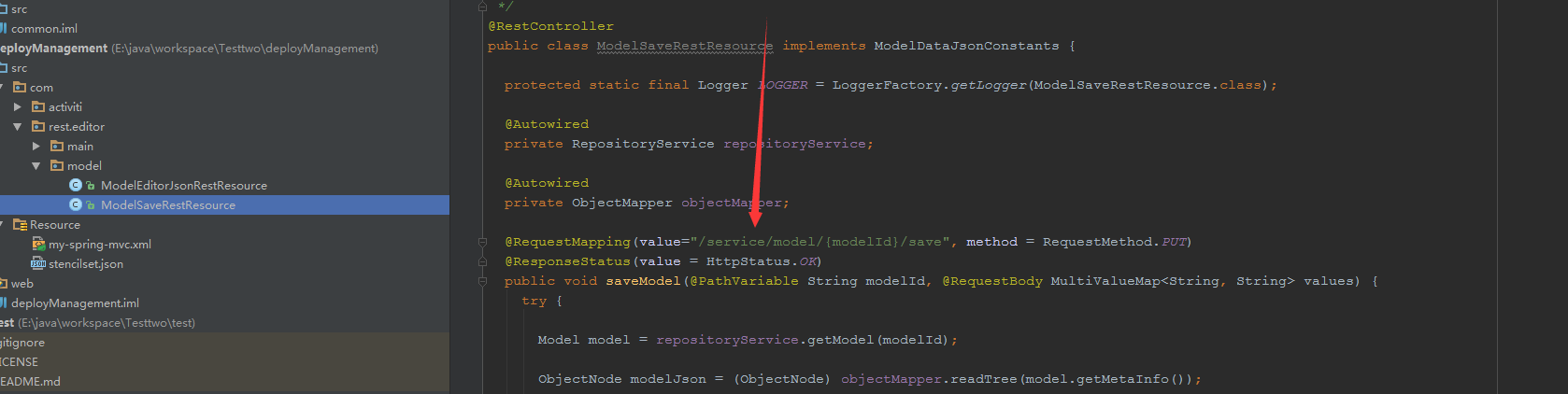
### 修改StencilsetRestResource的映射url:



### ModelEditorJsonRestResource.java url:



### ModelSaveRestResource.java url:



### 根据controller来配置：app-cfg.js

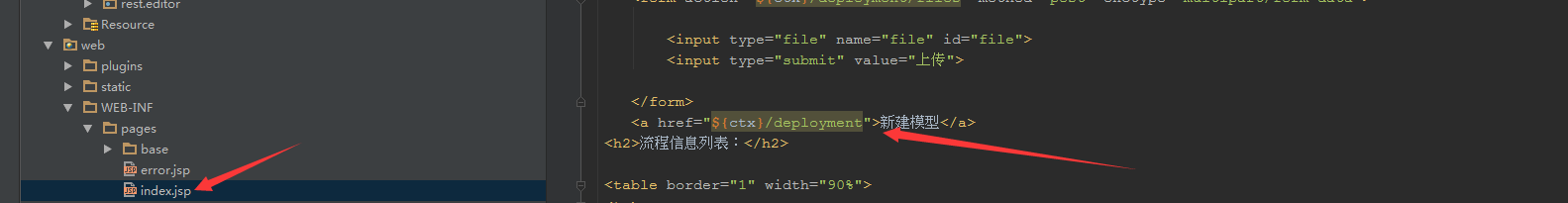


### 在自己的controller添加跳转操作：这里操作的是act\_re\_model这张表

添加如下代码：

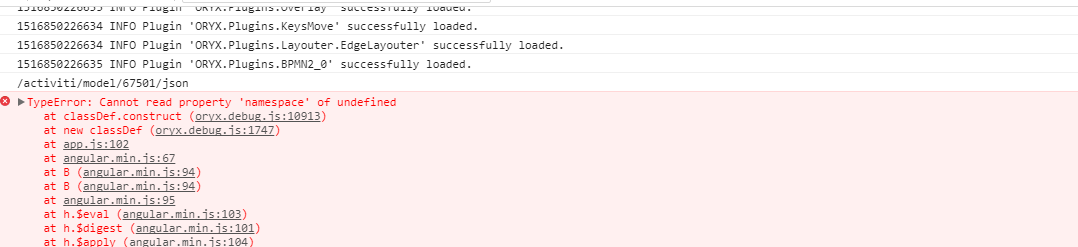
|  |
| --- |
| /\* \* \* 跳转到新建模型对象，这里需要创建一个modelId \* \*/ @RequestMapping(value = "/deployment",method = RequestMethod.*GET*) public void toCreate(HttpServletRequest request, HttpServletResponse response){  try {    ObjectMapper mapper = new ObjectMapper();//创建jackson对象  ObjectNode editorNode = mapper.createObjectNode();//创建节点  editorNode.put("id", "canvas"); //添加数据  editorNode.put("resourceId", "canvas");  ObjectNode stencilSetNode = mapper.createObjectNode();  stencilSetNode.put("namespace", "http://b3mn.org/stencilset/bpmn2.0#");  editorNode.put("stencilset", stencilSetNode);  org.activiti.engine.repository.Model modelData = repositoryService.newModel(); //使用repositoryService创建model   ObjectNode modelObjectNode = mapper.createObjectNode();  modelObjectNode.put(ModelDataJsonConstants.*MODEL\_NAME*, "hello1111");  modelObjectNode.put(ModelDataJsonConstants.*MODEL\_REVISION*, 1);  String description = "hello1111";  modelObjectNode.put(ModelDataJsonConstants.*MODEL\_DESCRIPTION*, description);  modelData.setMetaInfo(modelObjectNode.toString());  modelData.setName("hello1111");  modelData.setKey("12313123");   //保存模型  repositoryService.saveModel(modelData);  repositoryService.addModelEditorSource(modelData.getId(), editorNode.toString().getBytes("utf-8"));  //跳转到modeler.html并传递刚刚生成的modelId  response.sendRedirect(request.getContextPath() + "/plugins/procerss-editor/modeler.html?modelId=" + modelData.getId());  }catch (Exception e){  System.*err*.println("出现了异常");  } } |

### 在jsp界面中添加跳转的链接：

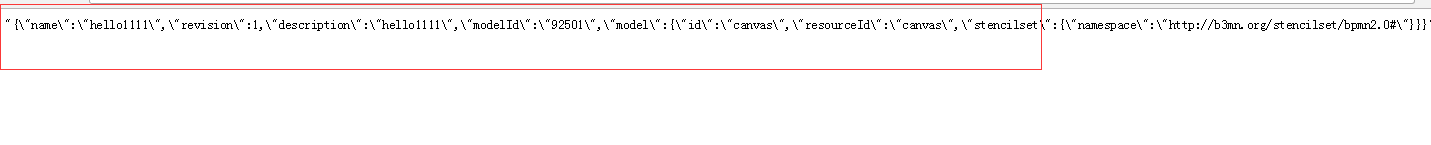


## 也许会出现的问题：

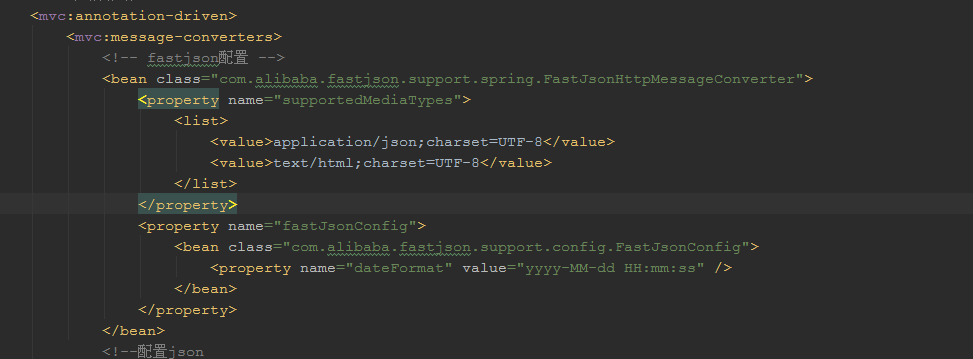
### 报下面这个错：



### 产生的原因：返回了带斜杠的json格式的数据：



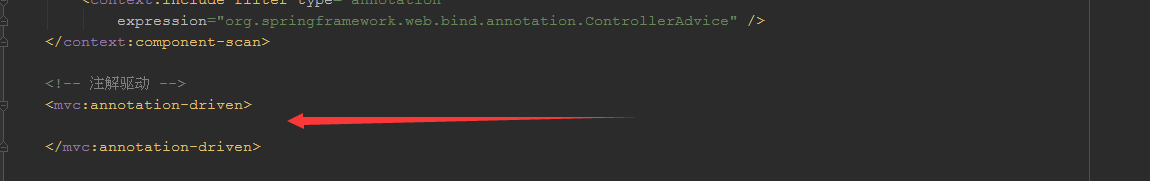
### 可是在spring中已经配置了json解析器啊？，why



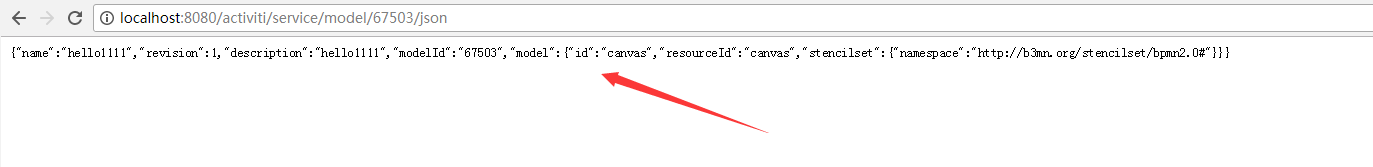
原因是json文件被二次装换了，controller返回过来的已经是json格式的文件了，在返回数据的时候又被转换了一次，导致了格式不对：controller已经返回了json数据：



## 解决办法一：删除spring mvc配置文件下的json解析：

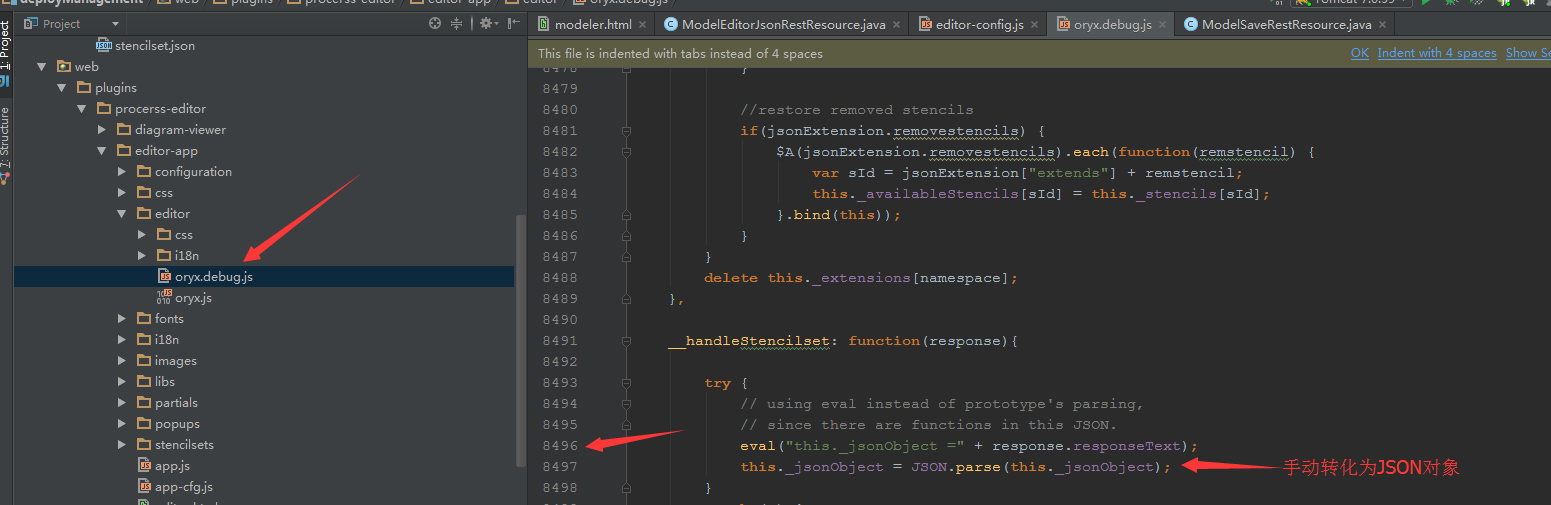


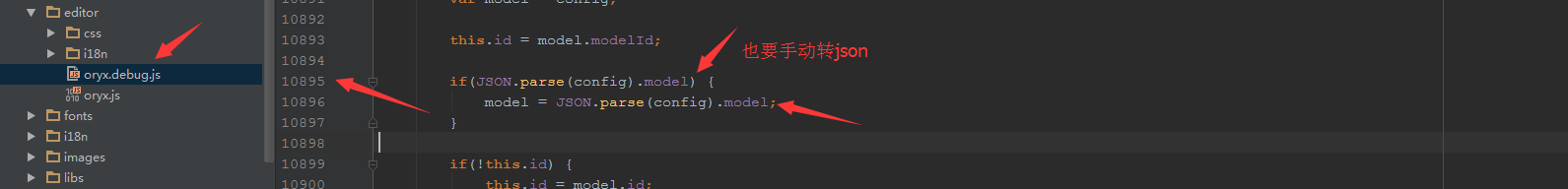
### 重启服务在访问：得到了正确的数据



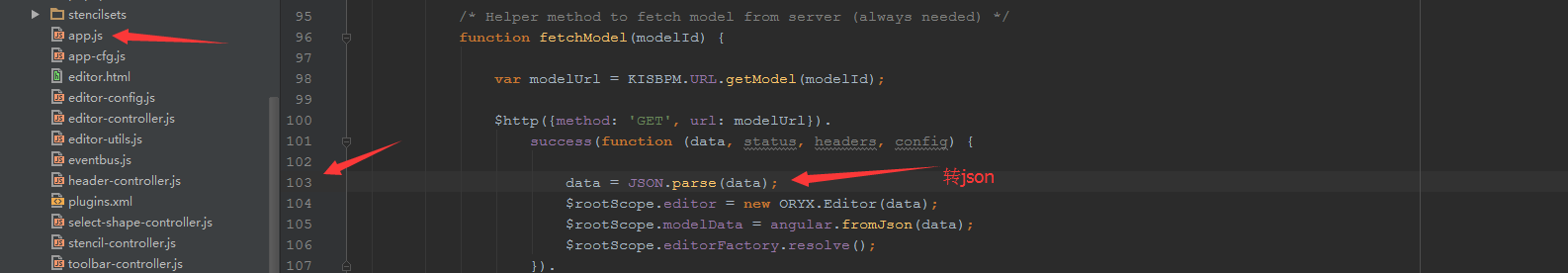
## 解决方法二：修改js文件

修改oryx.debug.js：





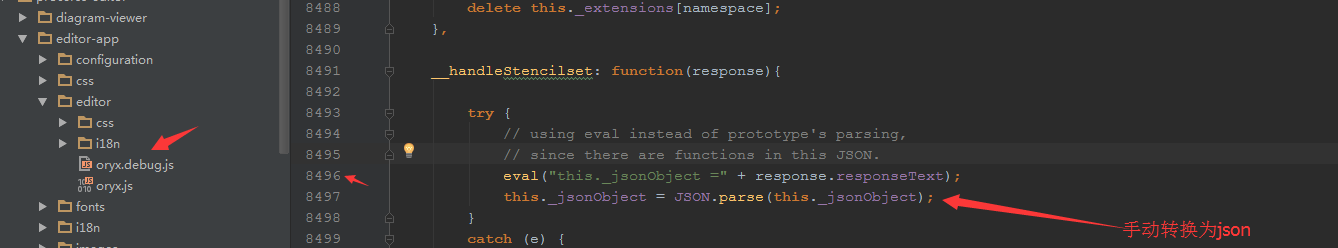
修改：app.js



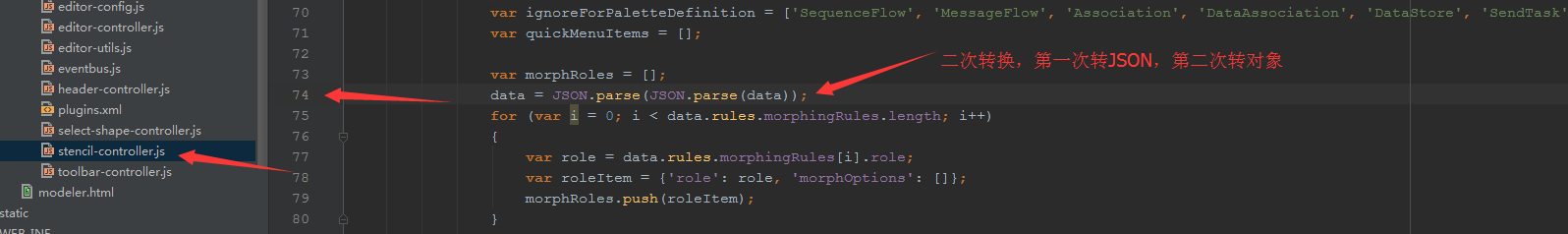
Js中如果报这个错：

|  |
| --- |
| TypeError: Cannot read property 'split' of undefined     at Object.ORYX.Core.StencilSet.stencilSet (oryx.debug.js:8640)     at oryx.debug.js:8619     at prototype-1.5.1.js:446     at Array.Object.extend.\_each (prototype-1.5.1.js:672)     at Array.Enumerable.each (prototype-1.5.1.js:445)     at Object.ORYX.Core.StencilSet.stencilSets (oryx.debug.js:8618)     at ORYX.Editor.getStencilSets (oryx.debug.js:11705)     at ORYX.Editor.\_createCanvas (oryx.debug.js:11192)     at ORYX.Editor.construct (oryx.debug.js:10921) |

oryx.debug.js：



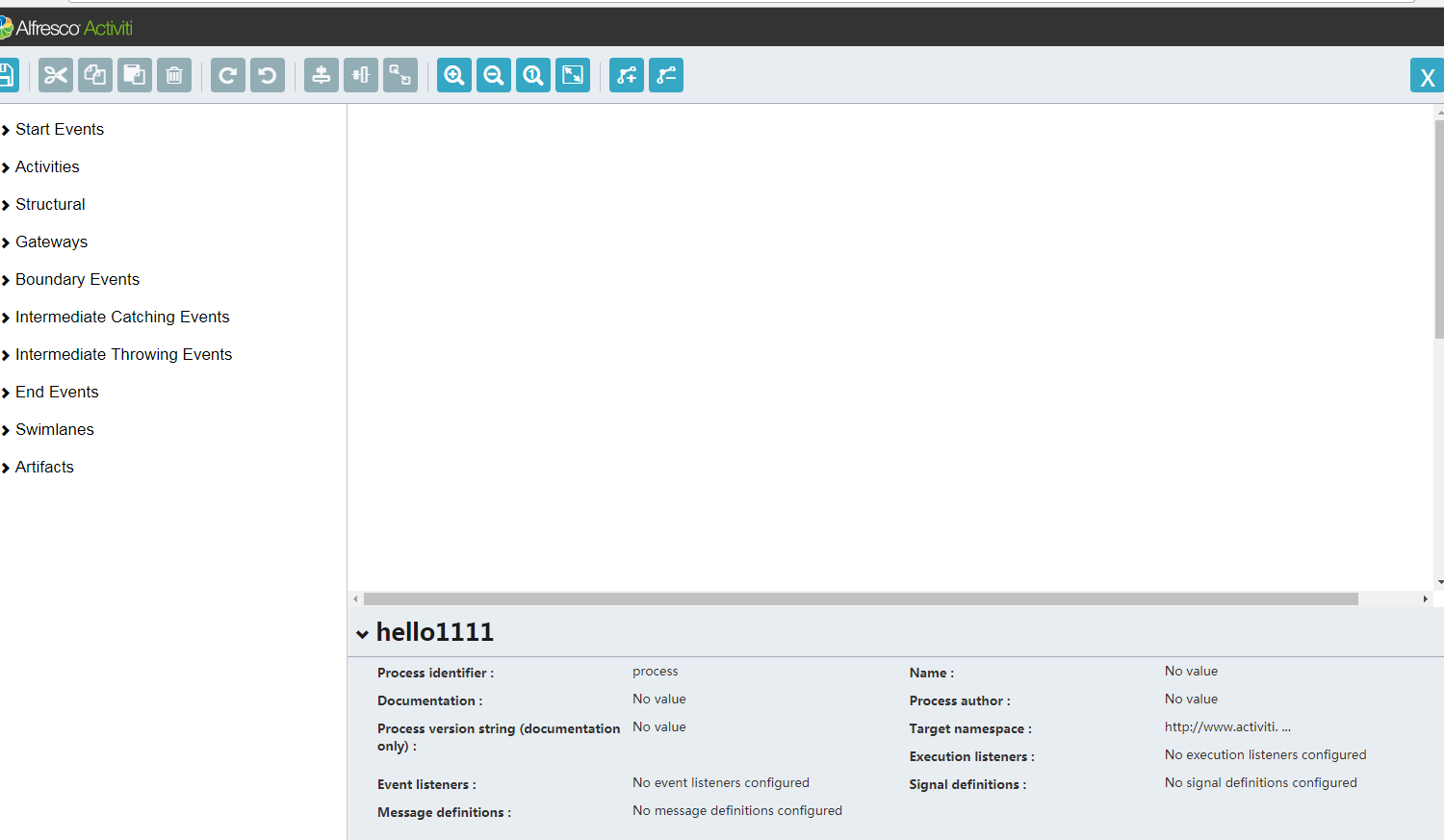
stencil-controller.js：



## 改了这些地方之后重新访问界面：



点击成功进入：



提醒：如果你使用导入jar的方式可以到activiti的开发包那里解压activiti-explorer.war找到对应的jar

