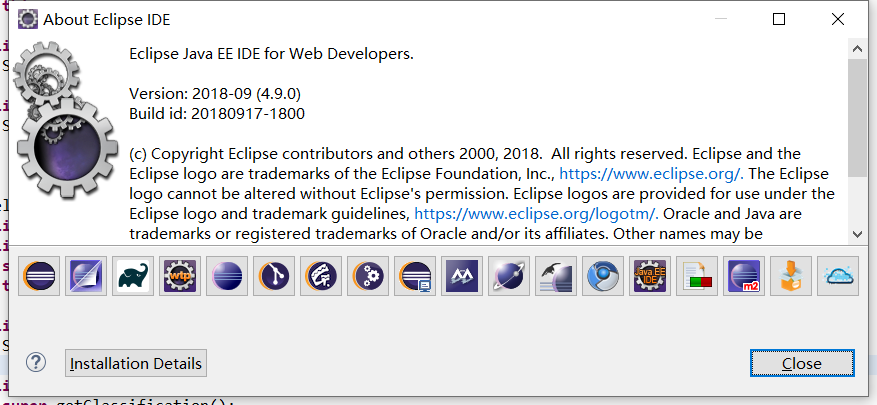
**1. 环境配置**

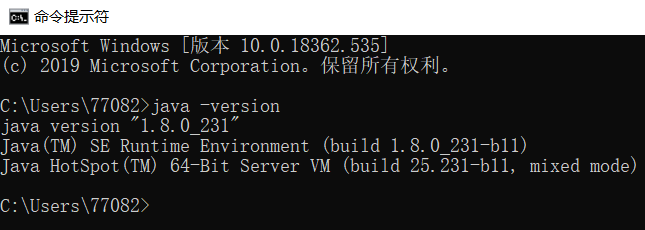
**1.1目前所用环境**

Tomcat7.0

eclipse:



Jdk:

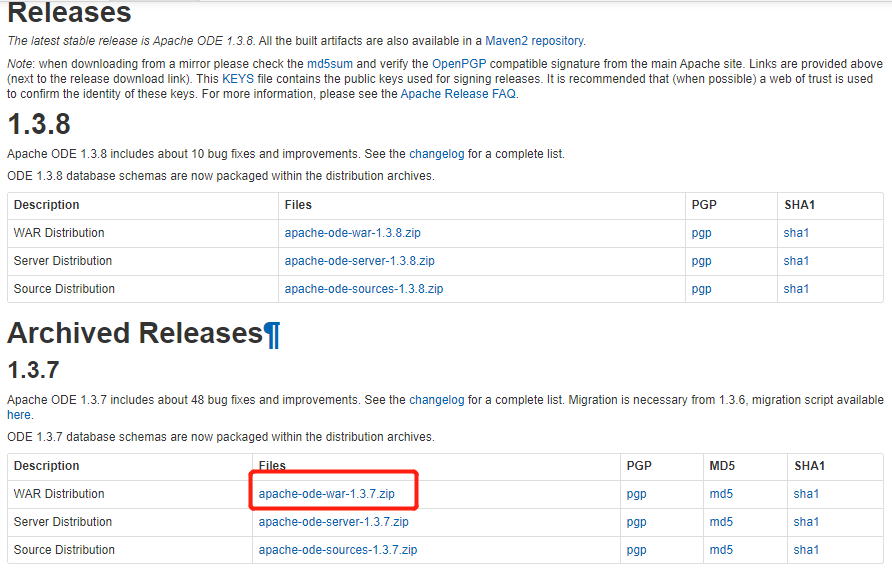


Tomcat与eclipse环境配置：

<https://blog.csdn.net/weixin_30955341/article/details/99508090>

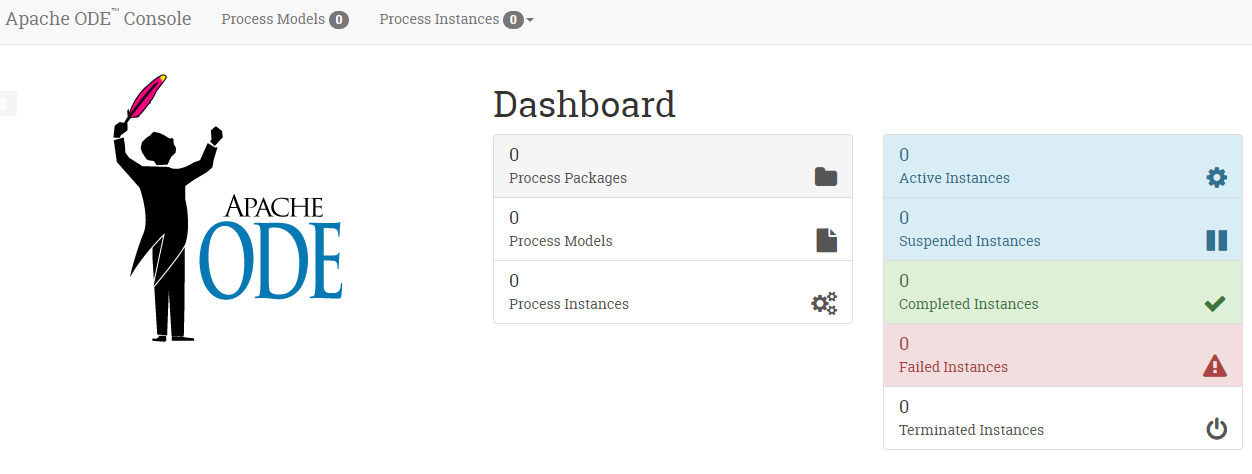
**1．2 配置 Apache ODE(BPEL 解析器)**

1) ODE 下载（本人所用版本为 1.3.7）

http://ode.apache.org/getting-ode.html

2) 解压文件，将 ode.war 拷贝到apache-tomcat-8.5.47\webapps目录下

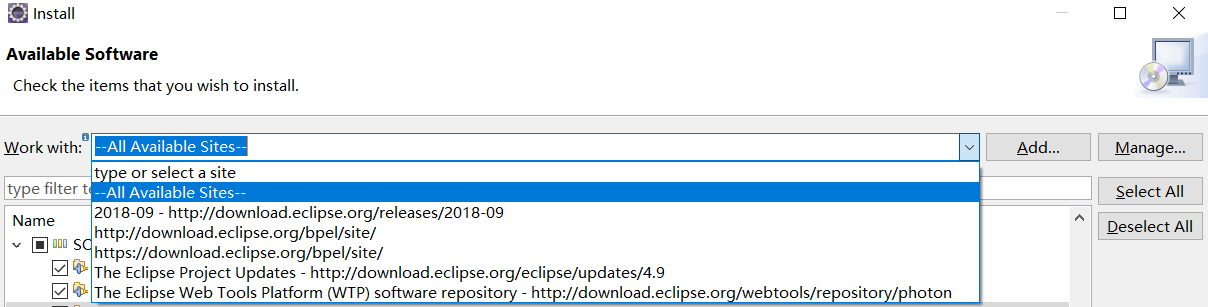
3) 重新启动 tomcat，输入 http://localhost:8080/ode 可以看到 ODE 的启动界面，如下所示：(如果启动不了tomcat则运行shutdown.bat清理即可)

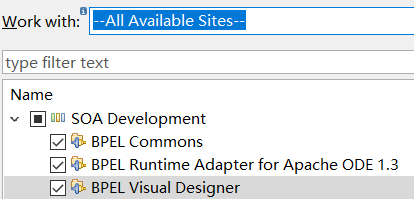


**1．3 安装 BPEL Designer 插件**

1) 启动 Eclipse，选择 Help -> Install New Software，进入 Install wizard，在输入栏输入 下载地址 <http://download.eclipse.org/bpel/site/>

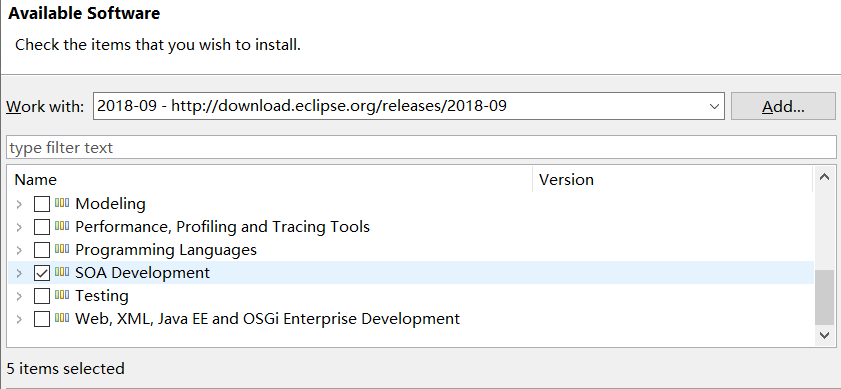
**(上面方法已经失效)**

**新方法：选择--ALL Available Sites-- ->选择SOA Development->勾选BPEL三项。**

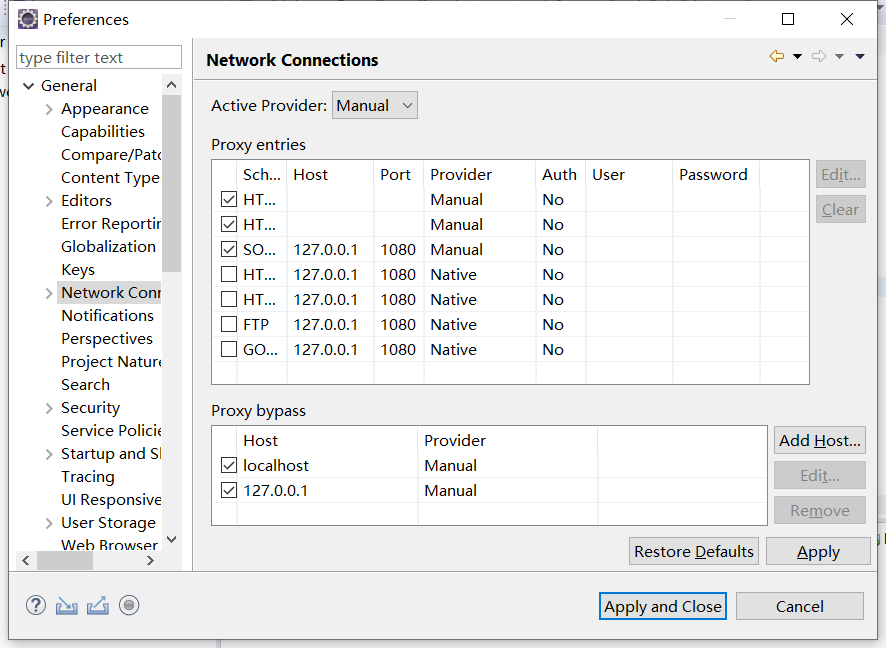


**假如不行的话：**

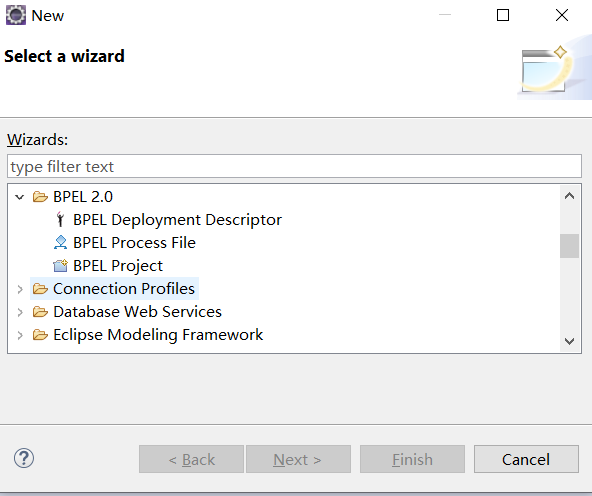
**选择2018-09如下所示的选项也是可以的。**



**PS：安装下载插件的过程会特别特别慢，你需要准备科学上网的工具，在window->preferences->General->Network Connections->设置socks**

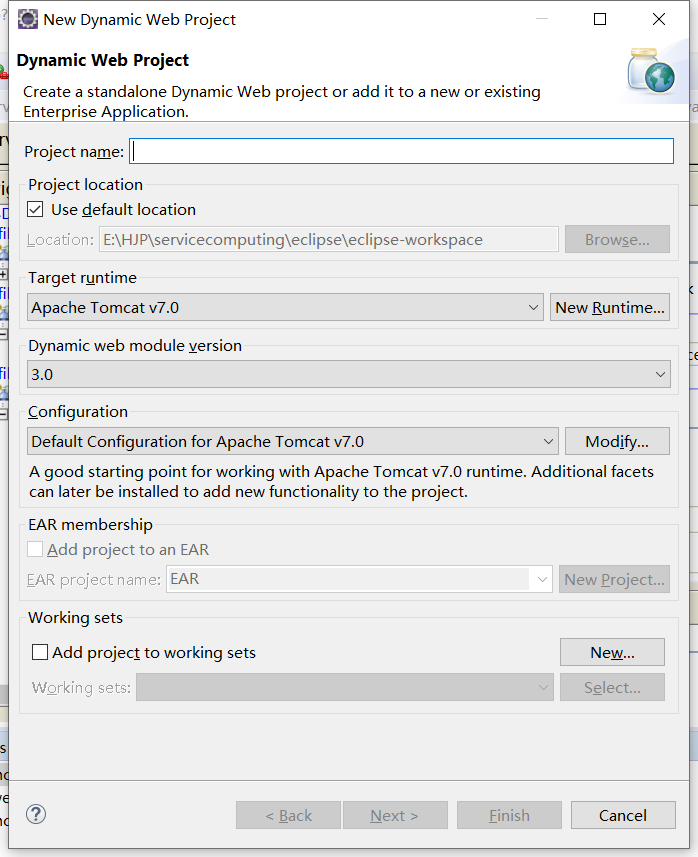


2) 安装成功后，在 Eclipse 中选择 File->new->Other,可以看到 BPEL 选项，如下图所 示，则证明 BPEL 开发环境配置完成。

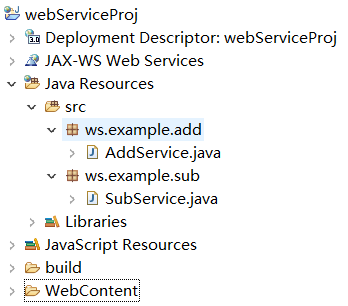


**2. 加减法WebService的开发部署**

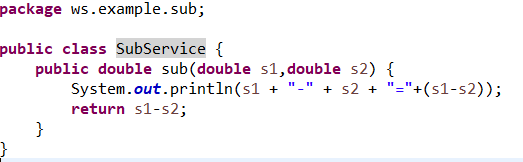
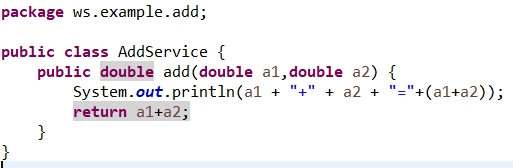
2．1 新建 Web 工程 启动 Eclipse，选择 File->New->Other->Web->Dynamic Web Project，新建一个 web 工 程。命 名为webServiceProj，如下图所示



2．2 新建加减法类 AddService 和 SubService

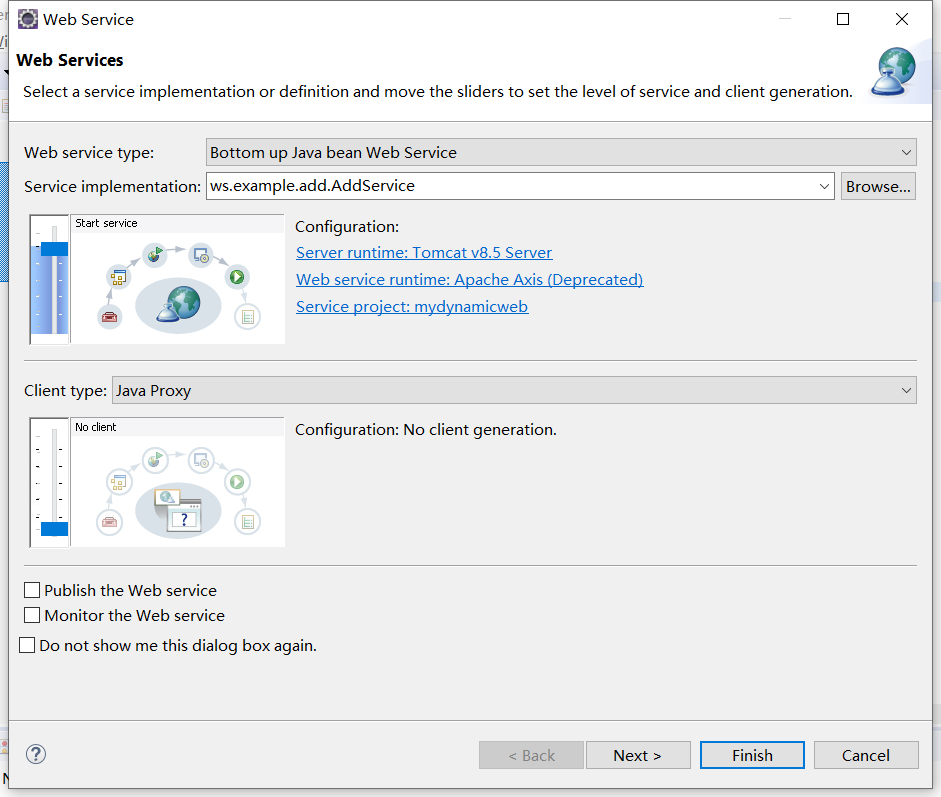


代码如下：

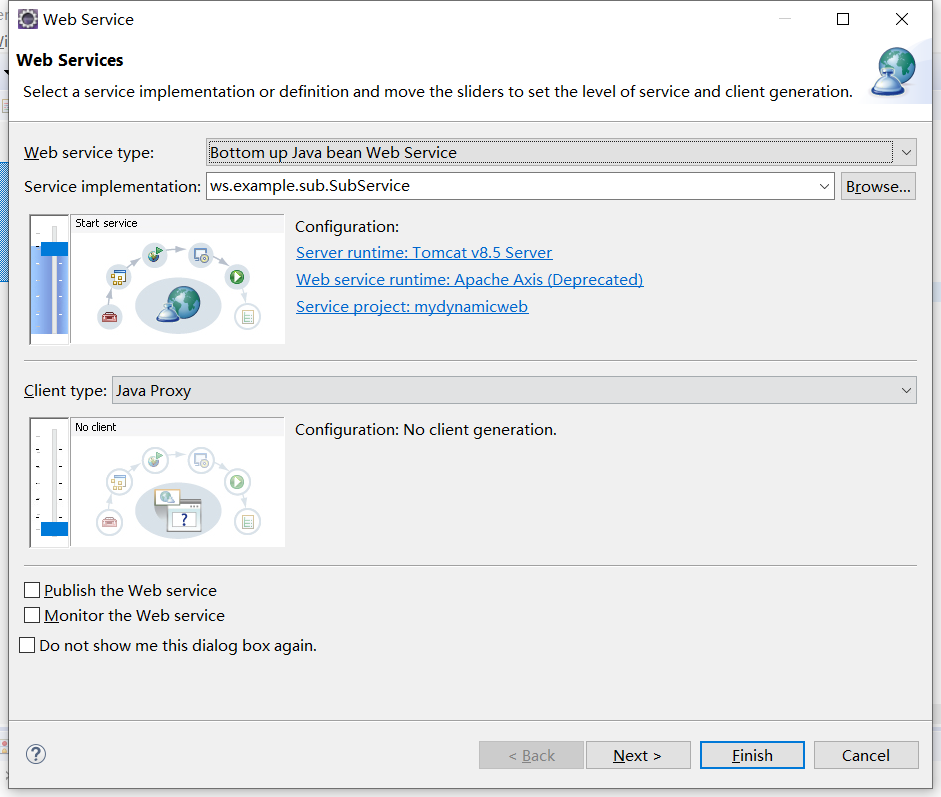


2．3 生成加减法的 WebService WSDL

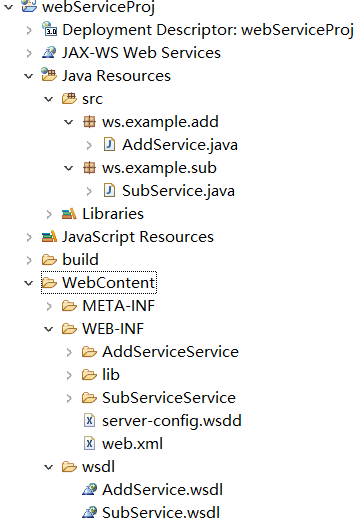
(1)java resource里面右键选择AddService.java，选择Web Services->Create Web Service，按照如下界面选择功能点，直接点 finish 完成：



(2)右键选择 SubService.java，选择 Web Services->Create Web Service，按照如下界面 选择功能点，直接点 finish 完成：



(3)点 Finish 完成之后，在webServiceProj的 WebContent 目录下可以发现生成了 AddService.wsdl 与 AddService.wsdl 文件,如下图所示



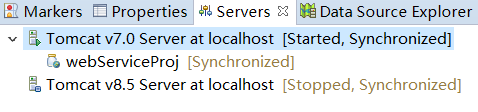
(4)启动 tomcat 服务器

**坑：eclipse中启动tomcat后, 无法访问localhost:8080**

<https://blog.csdn.net/wuting3680278/article/details/70196232>

**假如使用的是Tomcar8.5建议换成干净的Tomcat7.0版本，高版本有不可预知的错误。**

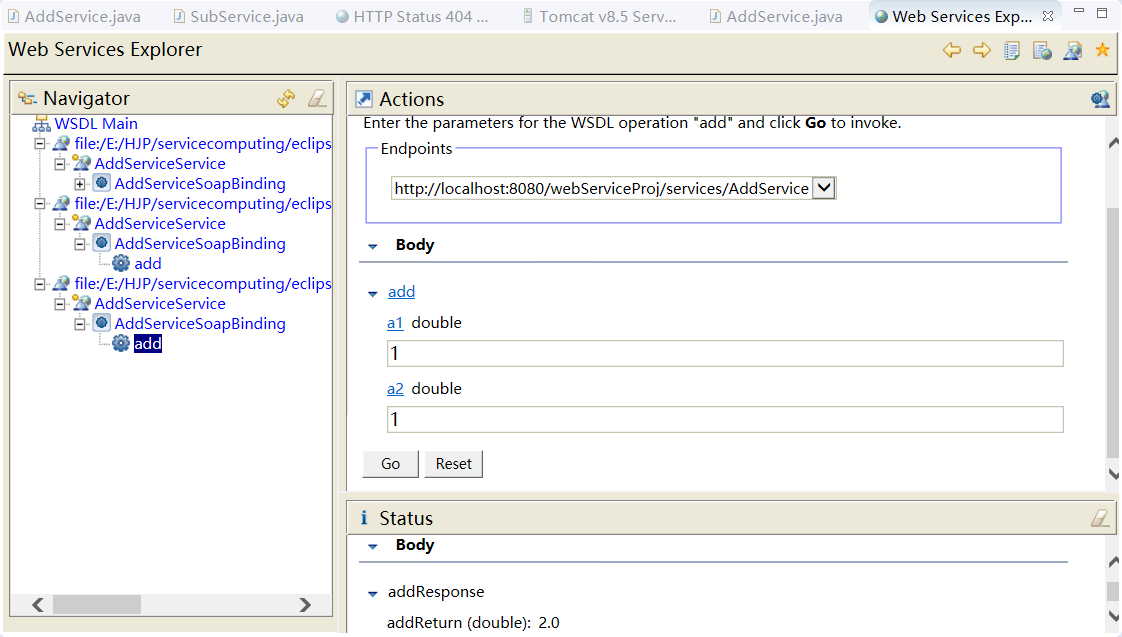
在下方选择Tomcat 7.0 右键start



对着webServiceProj右键run as->run on server即可。（404不要紧第5步正常输出即可，主要在于把文件映射到tomcat）

(5)启动 tomcat 服务器后，右键单击 AddService.wsdl 文件Web servicesTest with Web

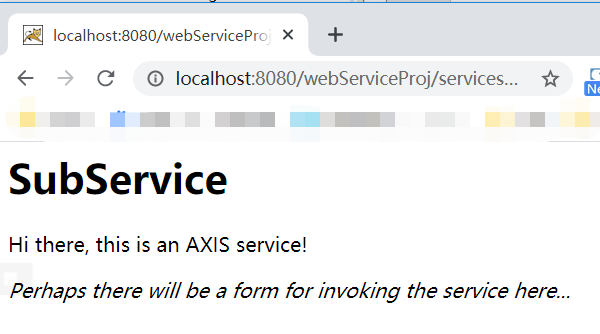
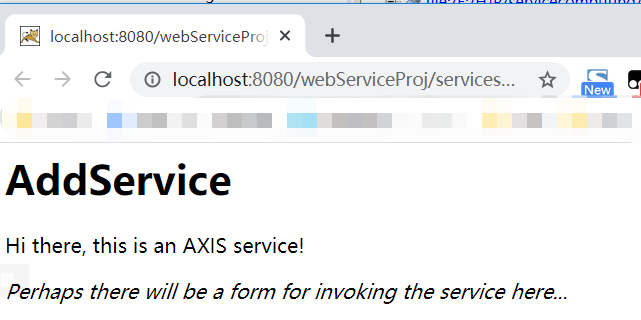
Services Explorer，对 SubService.wsdl 文件同样的测试方法，测试结果如下图所示：（调用服务成功，并且在Tomcat上面输出2.0）



2．4 加减法 WebService 的部署(1)右键 webServiceProj 工程，Export->Web->WAR file，导出 War 包，将导出的 War 包放到tomcat 的 webapps 目录下，重启 tomcat,即完成部署。

(2) 在浏览器地址栏输入

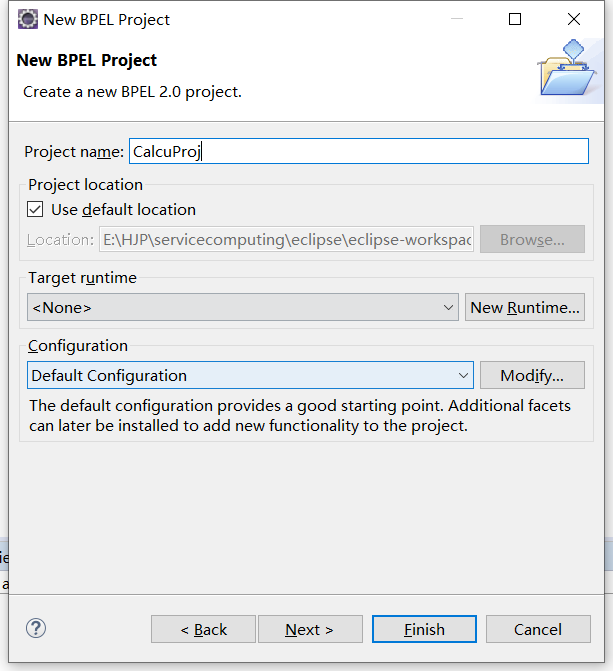
http://localhost:8080/webServiceProj/services/AddService?wsd，若能

打开服务对应的 wsdl 则意味着服务发布成功。

**3.BPEL实例开发与部署**

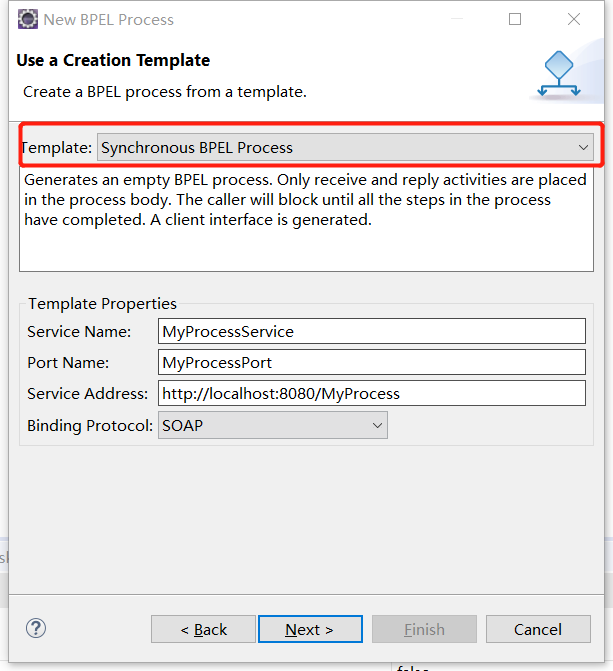
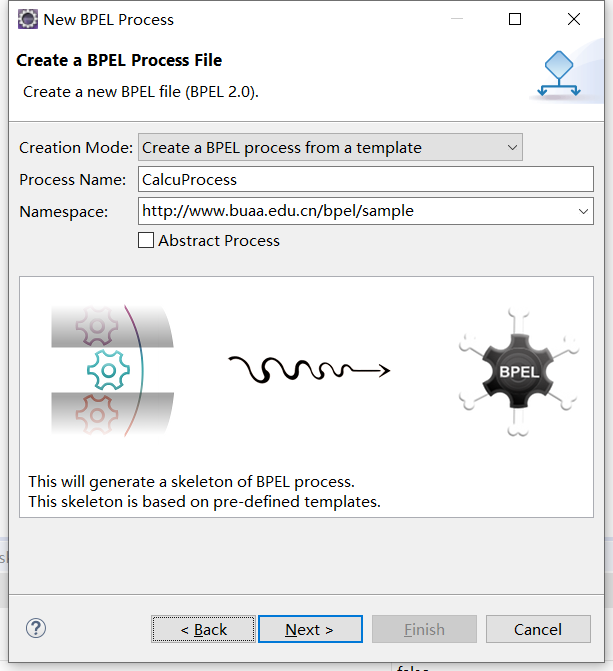
**3.1 创建 BPEL 工程**

启动 Eclipse，点击 File->New->Other，然后选择 BPEL 2.0 ->BEPL Project，在弹 出的对话框中 输入项目名 CalcuProj。



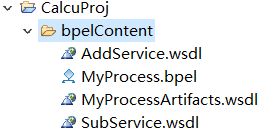
**3.2 创建 BPEL 流程**

右键 CalcuProj 工程下的 bpelContent，点击 New->Other，选择 BPEL 2.0- > New BPEL Process File ，出现 New BPEL Process 对 话 框 ， 输 入 流 程 名 CalcuProcess ，名字空间设为：http://www.buaa.edu.cn/bpel/sample ，选择同步流程。点击 Finish 完成。如下图所示：



3.3 导入加减法服务的wsdl文件

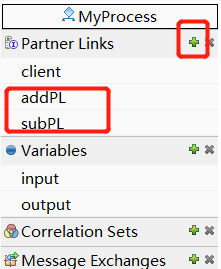
将 AddService.wsdl 与 SubService.wsdl 文件复制到 BPEL 工程目录里。



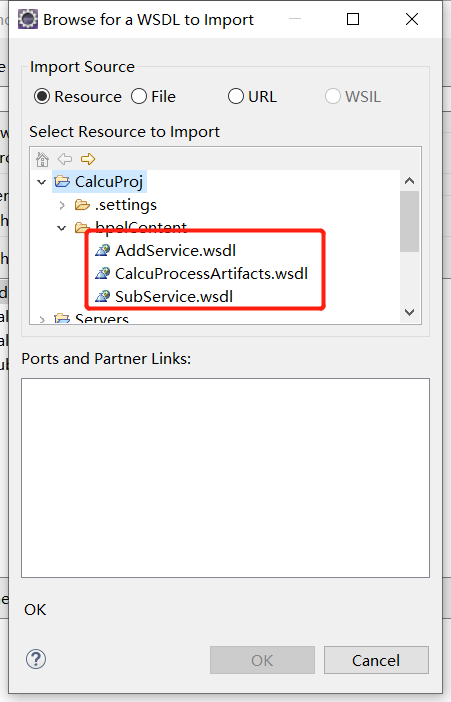
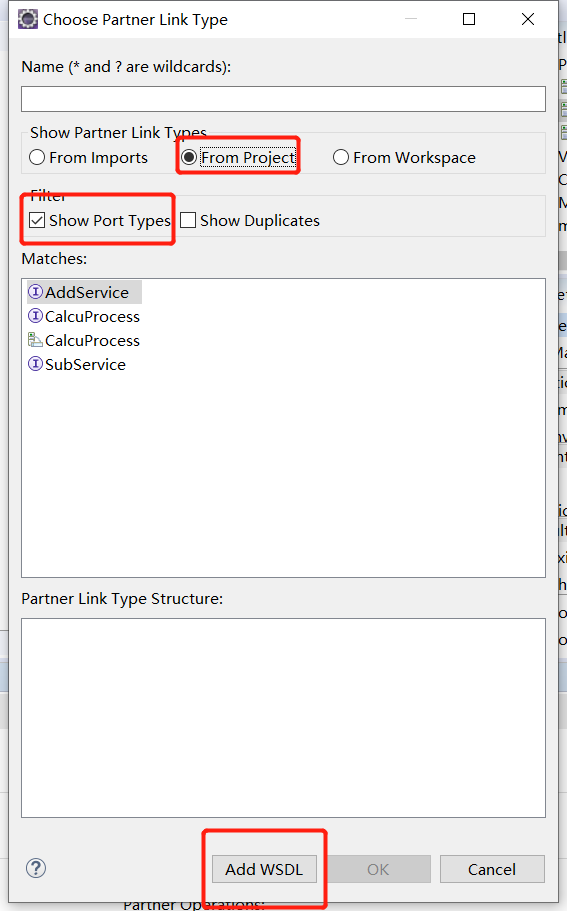
3.4 设置伙伴链接 Bpel

把涉及其中的所有服务都称之为伙伴链接。Bpel 对应的 CalcuProcess.bpel 本身也是一个 伙伴链接。系统已经自动生成了。对于流程中要用到的加法服务，减法服务，应该生成对应 的伙伴链接。

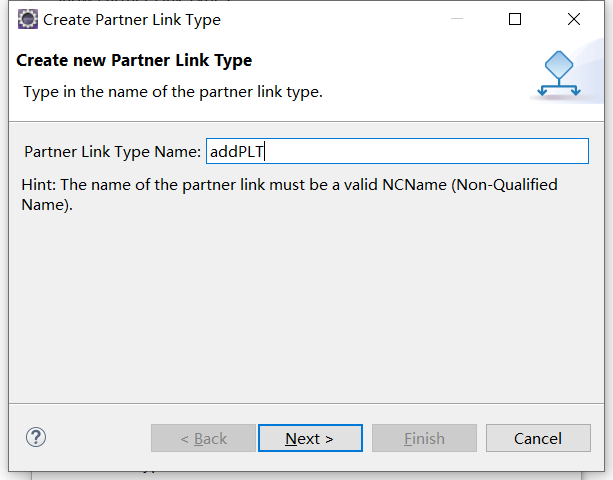
1. 在右上角的 Partner Links 中，增加两个链接，分别是 addPL，subPL。如下图所示：

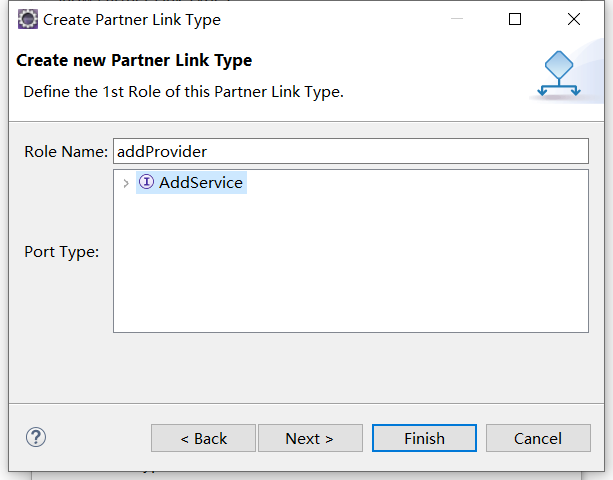


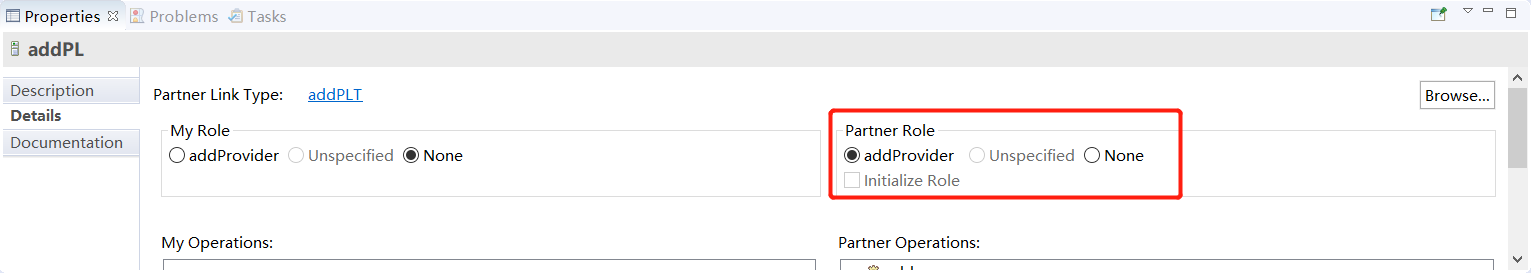
(2)设置伙伴链接. 右击 addPL，选择属性打开属性视图，在属性视图中 Details 页面 中点 Browser，打开对话框。点 Add WSDL 把用到的两个 WSDL 即 AddService.wsdl 和 SubService 加进来。



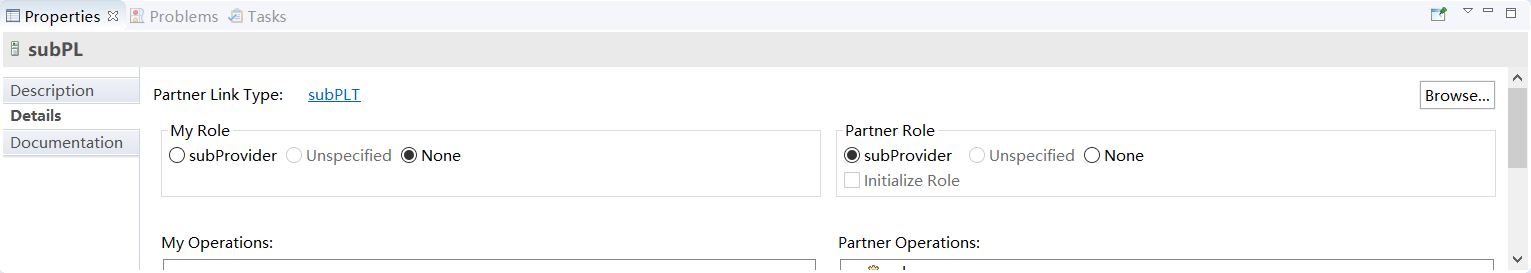
(3)选上 AddService 接口，点 Ok,弹出伙伴链接类型定义，把名字取为 addPLT，点 Next。要 求输入 Role Name，这里输入“addProvider”,点 Finish 完成。





(4) 同时在属性视图中把 ParterRole 选为 addProvider。如下图所示：

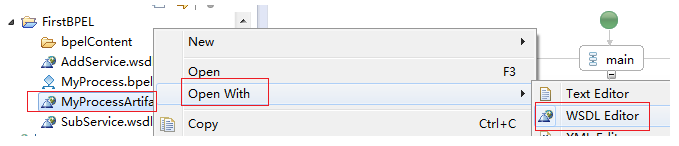
(5) 以同样的方式建立 subPL 设置，subPLT，subProvider。

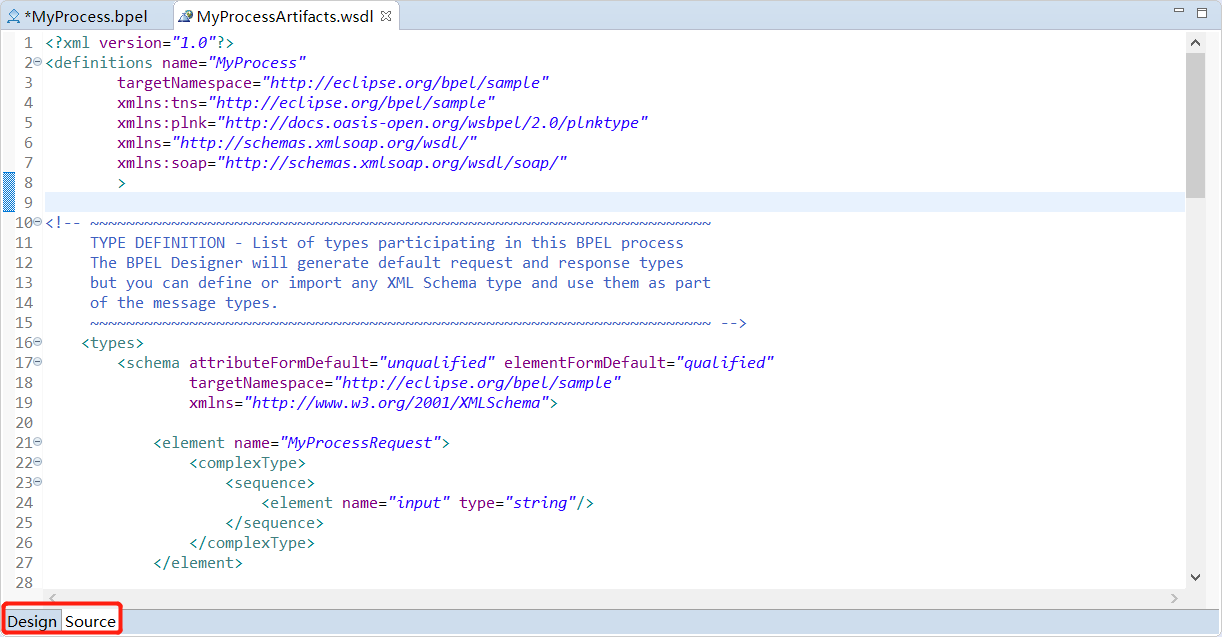


**3.5 修改 MyProcessArtifacts.wsdl 文件**

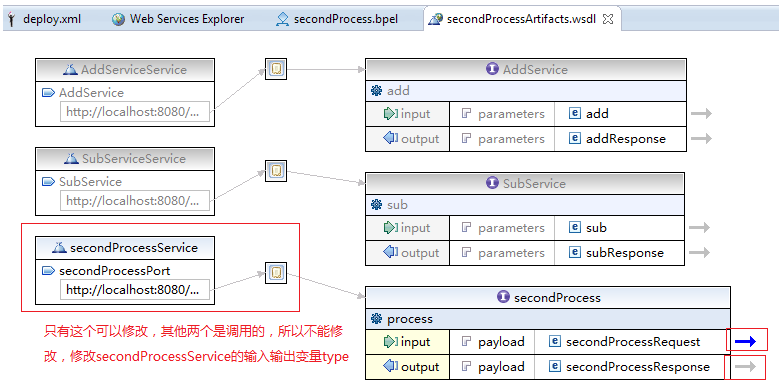
因为其默认的输入输出不符合我们项目的要求，右键 CalcuProcessArtifacts.wsdl 文件，用 open

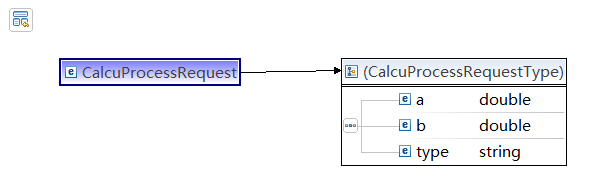
with WSDL Editor 打开该文件，转到设计视图里。如下图所示：

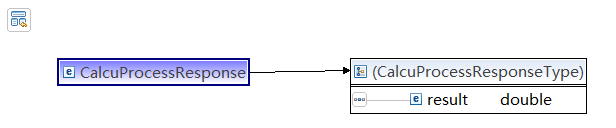




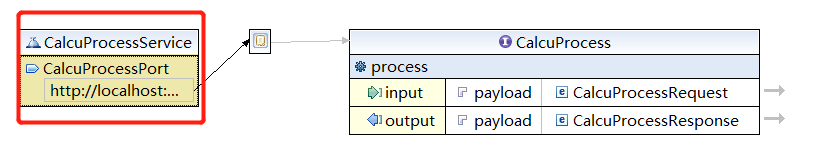
单击 input 后边对应的箭头，打开一个新的 Editor 对输入消息进行配置。进去后选中类型，右键，就可以增加元素，并设置元素的类型，结果如下，其中 a，b 分别表示需要加减的2个值，type 用来区别加减法，当type=’add’，时，调用加法服务，当type=’sub’时调用减法服务。





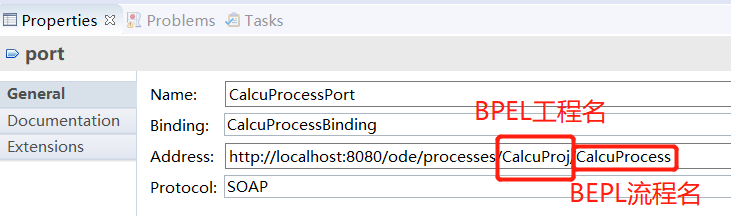
同样，单击 output 后边对应的箭头，修改后结果如下： 

**最后修改 CalcuprocessPort（非常重要，直接关乎 BPEL 工程部署的成功与否，以往教程上基本没有，会导致一次又一次的失败）**



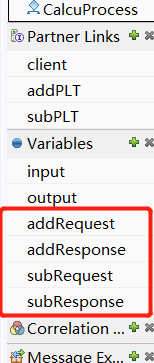
**用鼠标右击 CalcuprocessPort，选择 Show properties，选 择 general 选 项 卡，修改其中addresss，localhost 接口号根据个人设置而异。**

**Address：<http://localhost:8080/ode/processes/CalcuProj/CalcuProcess>**



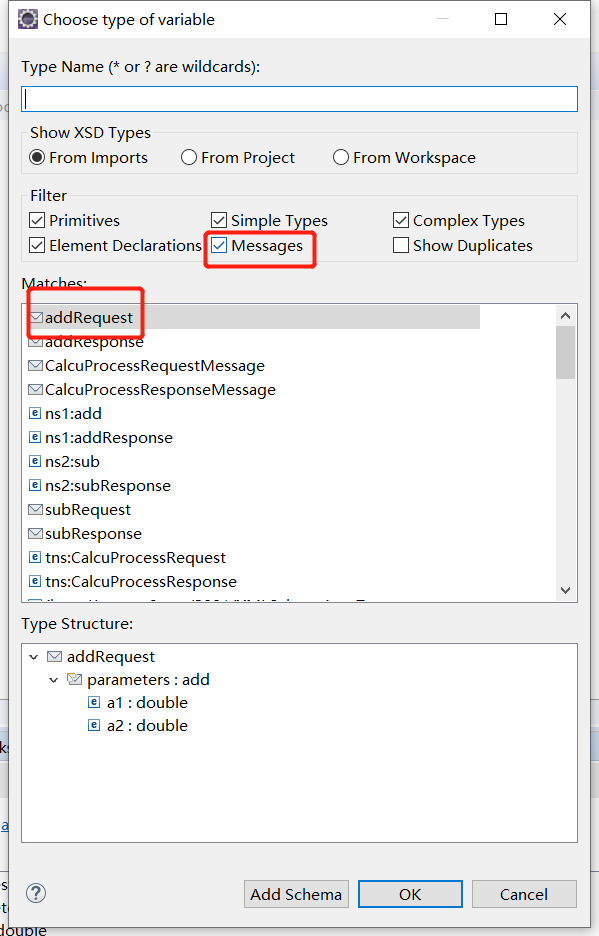
3.6 创建调用服务时需要的变量

每次调用一个服务时都要有对应的变量，同时要进行赋值。我们这里要调用加法服务和减法服务。因此需要额外加入addRequest,addResponse,subRequest,subResponse加完后的效果如下：



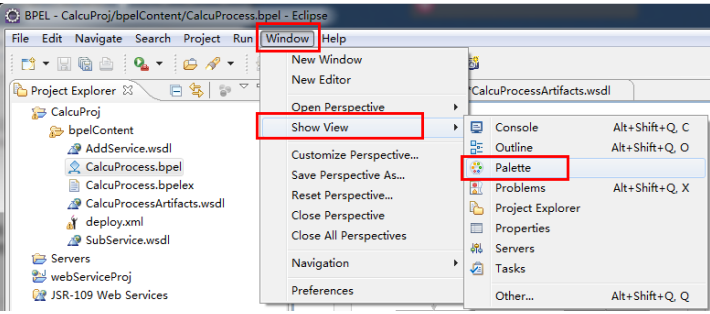
**3.7 配置变量**

点上 addRequest 变量，属性视图中切换到 Details 页，点右上角的 Browse..按钮，打开类型选择对话框。addRequest 是加法服务的输入，因此把它的类型设为加法服务里的输入定义就可以了。一个变量的类型可以是简单类型，也可以是复杂类型，还可以是 wsdl 中的Message 定义。简单起见，我们这里就把addRequest 的类型设为 AddService 服务中的addRequest Message。 同理，其他 3 个变量也做相应的处理。

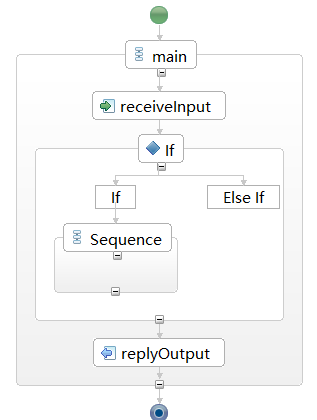


**3.8 BPEL 流程设计**

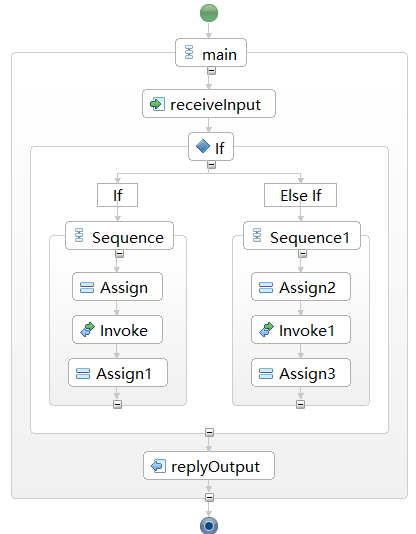
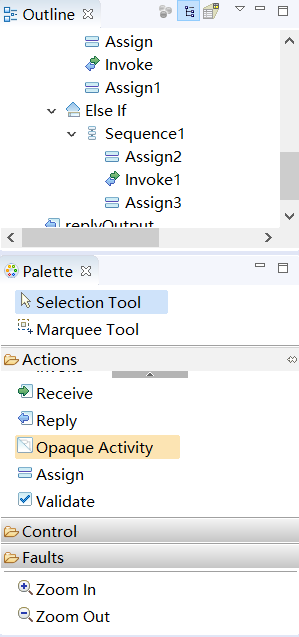
打开控件视图,如下图:



首先点击选中删除FIX\_ME-Add\_Business\_Logic\_Here，然后选择IF控件，插入到receiveInput后面,添加 if 控件,然后右击 Add Elself，同样加入其它控件:



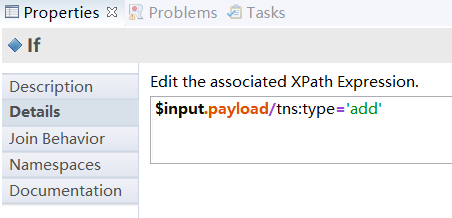
然后选中新添加的IF控件，右键Add Elself，然后便依次拖入invoke，assign控件，最终效果如下:

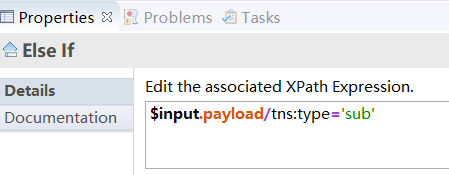
**3.9 BPEL 流程控件配置**

(1)If 语句的配置

选择 If,在属性视图中点开 Details 页，输入条件语句：$input.payload/tns:type='add'，如下图所示：

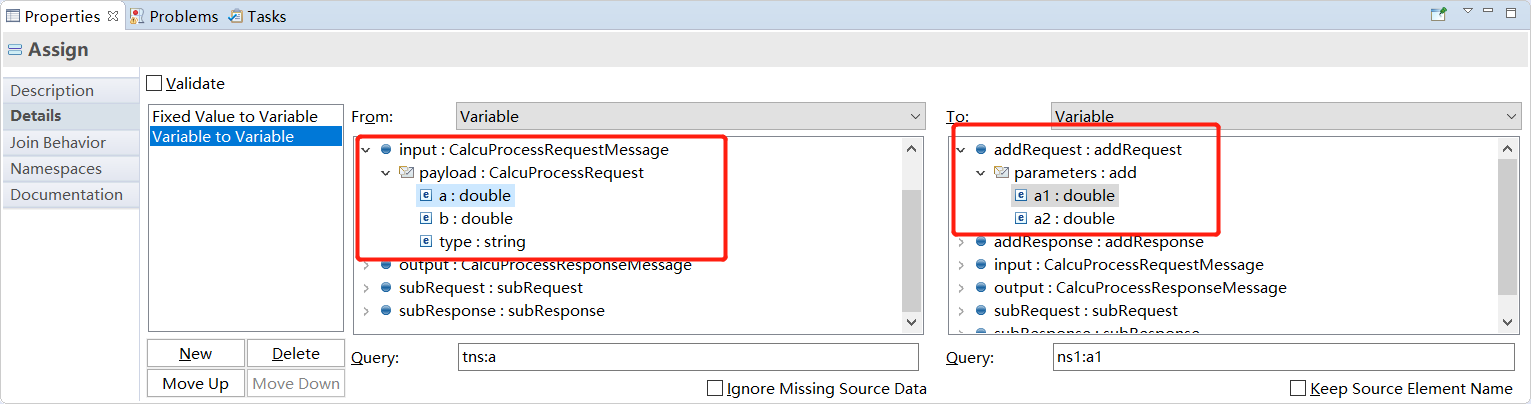
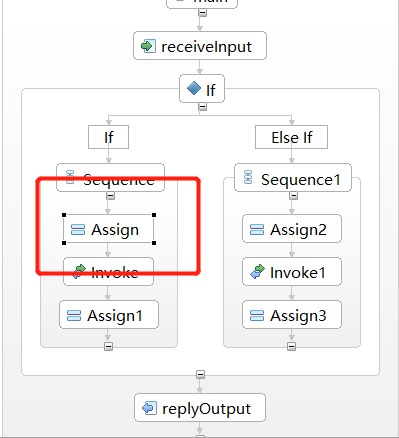


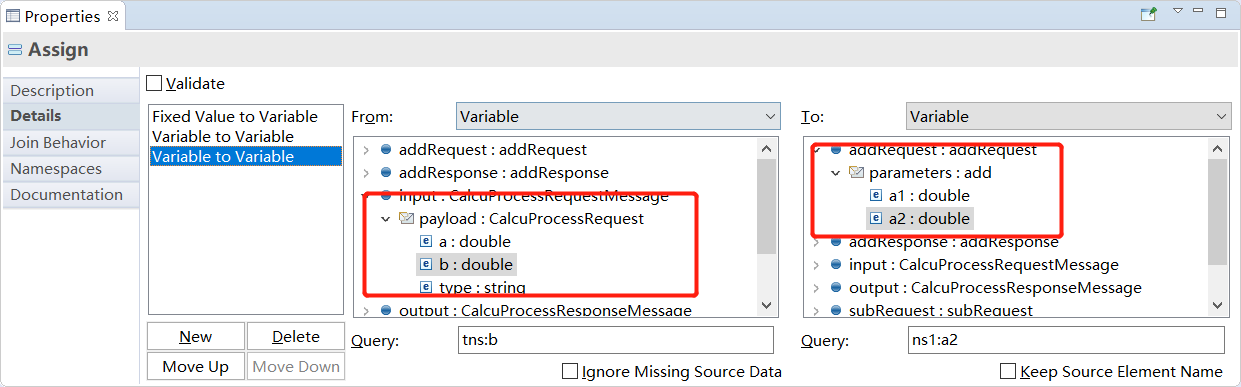
同样，选择 Elself：



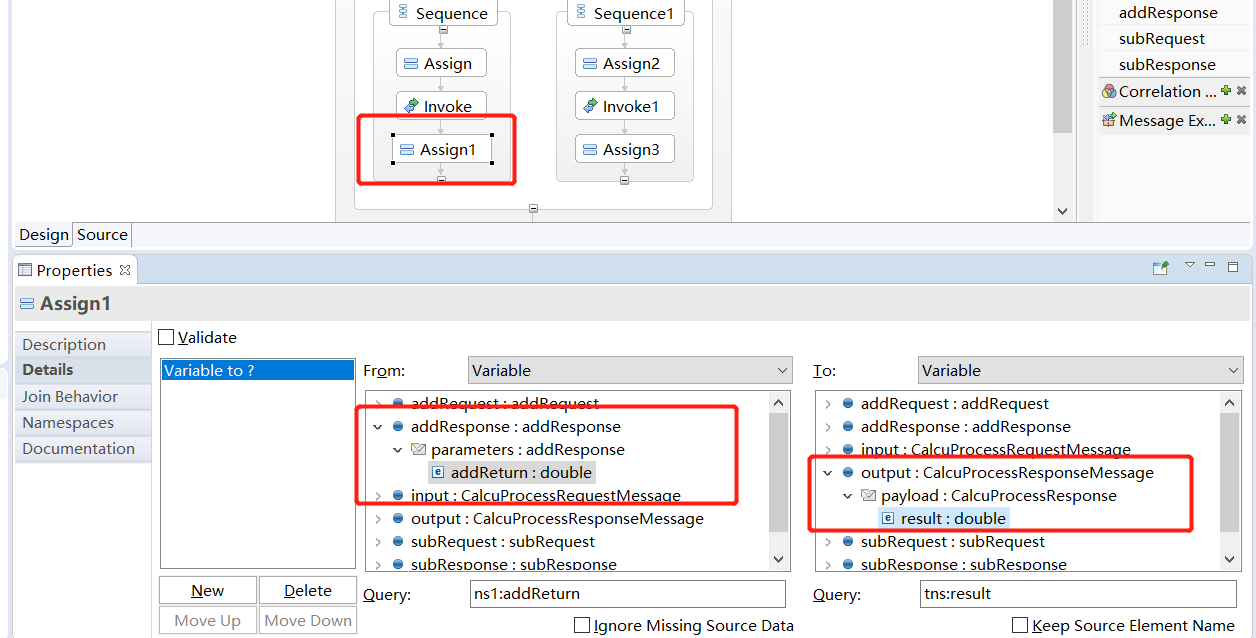
(2)Assign 语句的配置

第一个 Assign 的作用是把输入变量 input 中的前两个参数传给 addRequest 变量。需要注意的是，BEPL 中的变量赋值只能是一个一个的赋，不能直接把 input 赋给addRequest（当然这里也不匹配，即使匹配也不行），而只能把 input 下的参数挨个赋给addRequest。如下图所示，将 input 的a赋给 addRequest 的值 a1，此时会弹出一个对话框，询问是否要初始化addRequest，选择 Yes。同理，将 input 的值b赋给 addRequest的值a2。

的值 a2。 



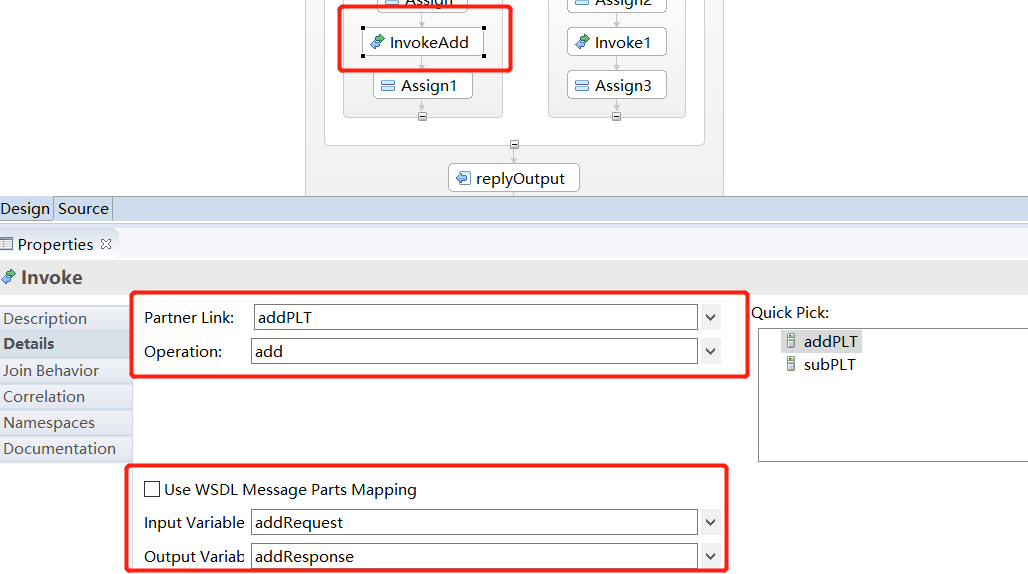
Assign1 的配置：



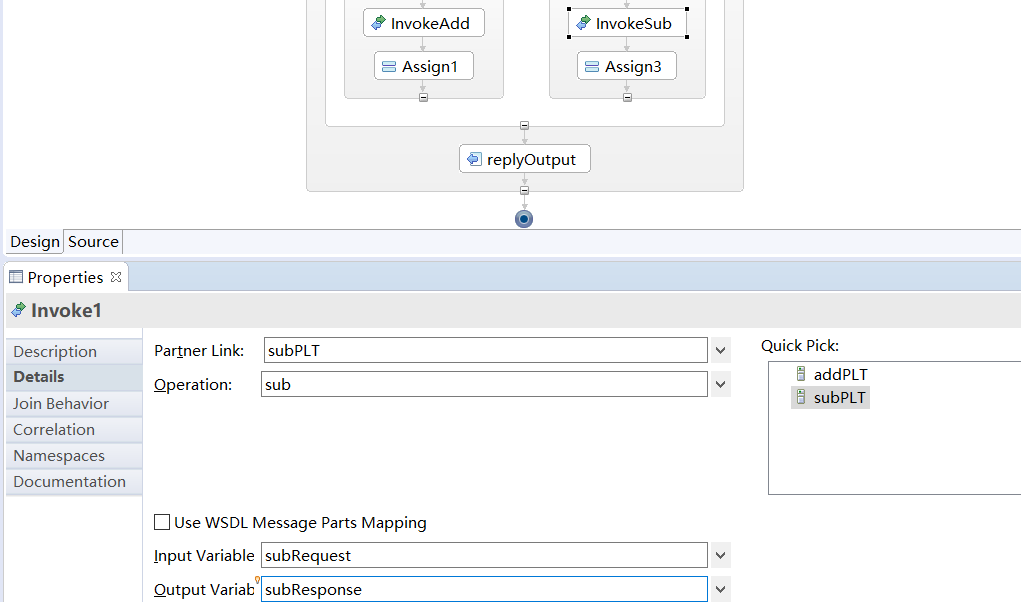
同理配置减法的 Assign2, Assign3。(3)Invoke 语句的配置

Invoke 的作用是调用伙伴链接对应接口下的对应操作，还需要指定输入输出变量。左键选择 InvokeAdd（Invoke），在属性视图的 Details 页面中进行如下配置：

InvokeAdd:

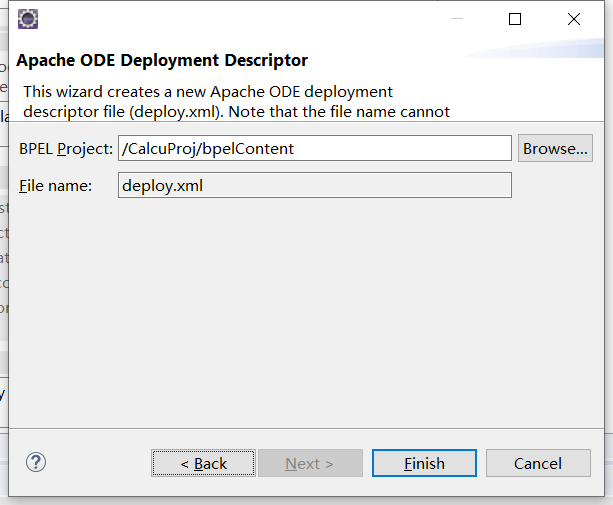


InvokeSub:

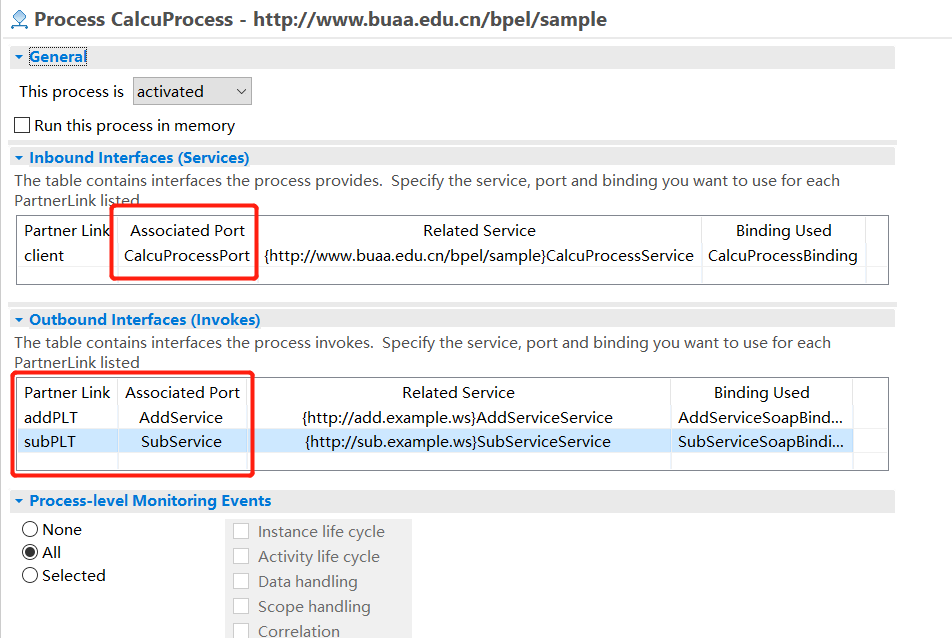


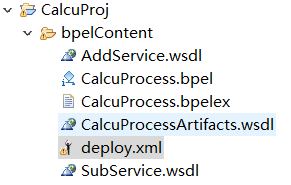
**3.10 BPEL 流程发布**

BPEL 流程最终的目的也是发布成一个 web-service，只不过这个 web 服务自己基本上不实现功能，主要是通过调用别人的功能来完成任务。因此，最后也要把流程对应的 WSDL 发布出去。默认生成的 WSDL 只有类型定义，消息定义和操作定义，缺少绑定信息和服务定义。选择CalcuProj→bpelContent右键→New→Other→BPEL 2.0→BPEL Deployment→Descriptor，如下所示，直接 Finish 即会生成一个 deploy.xml 文件。



打开 deploy.xml 进行端口配置：

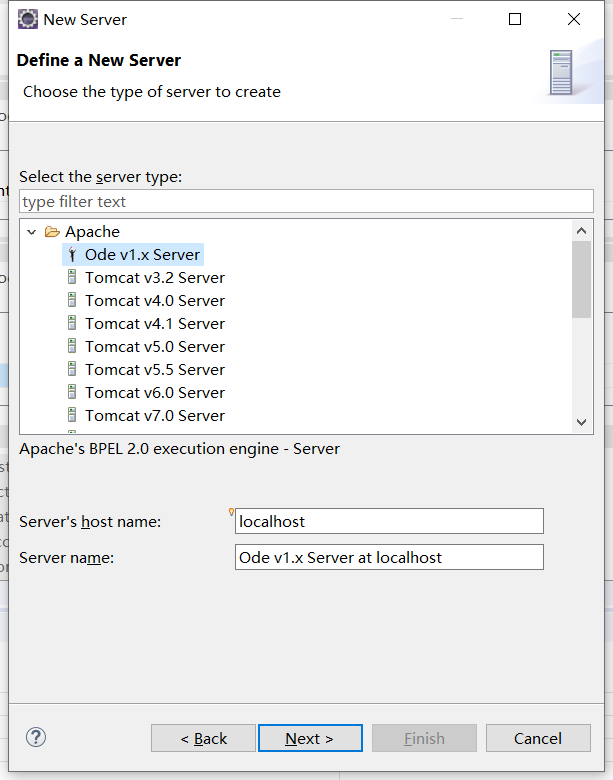


最终 BPEL 工程目录如下： 

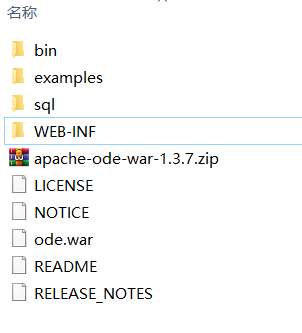
**4. BPEL 运行与测试**

**4．1 创建 ODE 服务器**

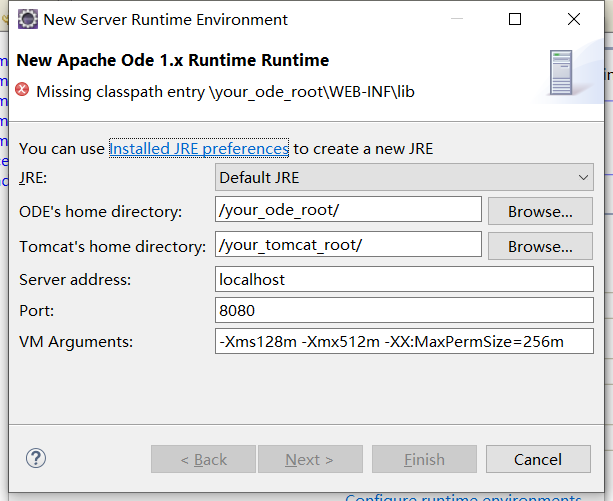
Eclipse中File→new→other→Server→Ode v1.x Server→Finish



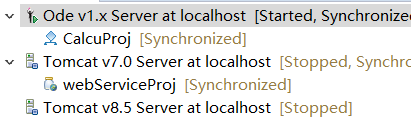
首先在下载的apache-ode-war-1.3.7里面新建WEB-INF然后把lib文件夹移动进去：

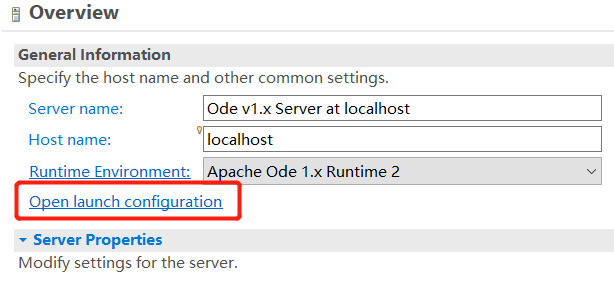


然后这里的配置选择你的ode的根目录和tomcat的根目录。

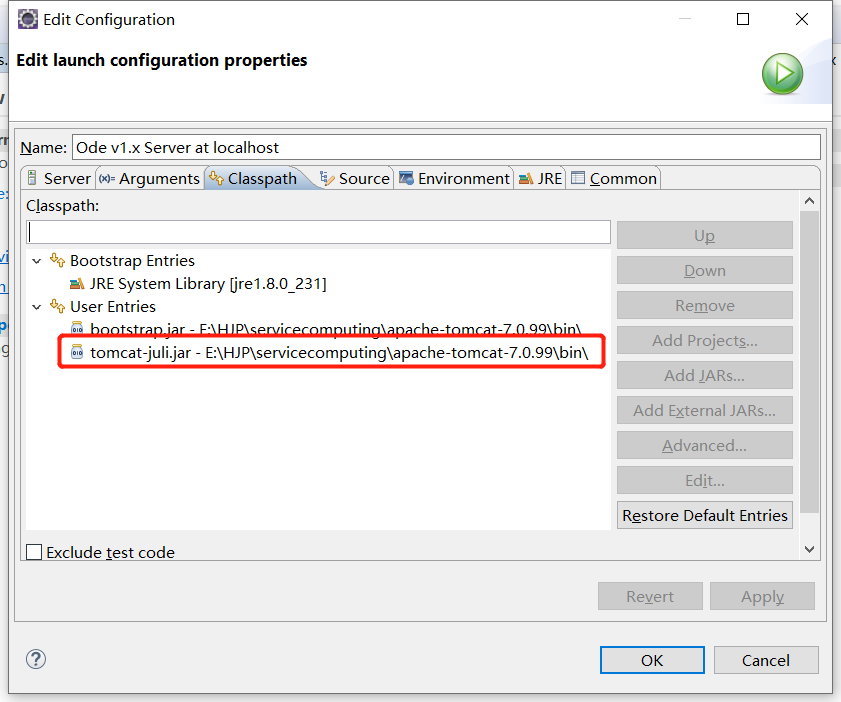


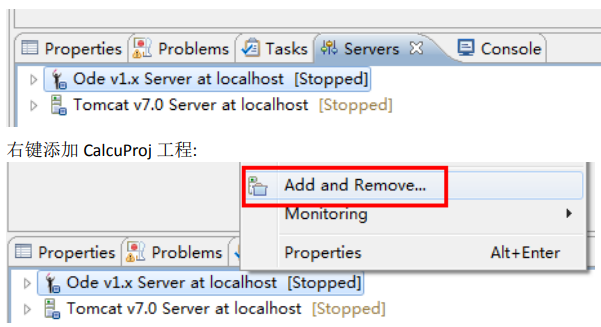
**然后右键Ode v1.x Server，点击Open，点击open launch configuration**

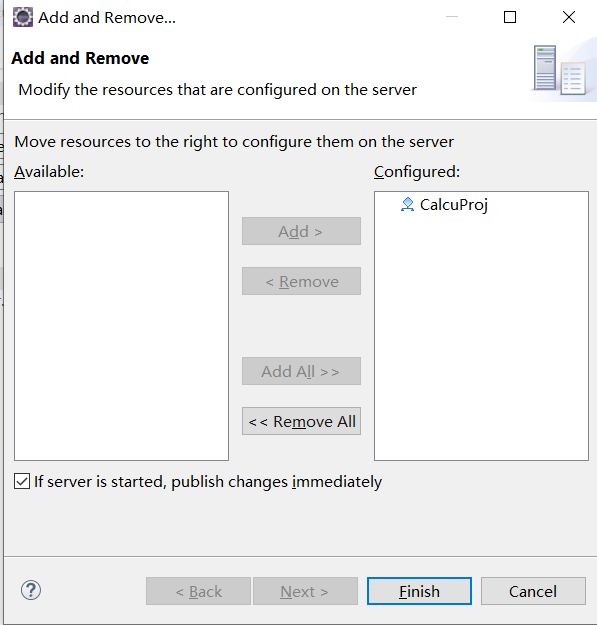




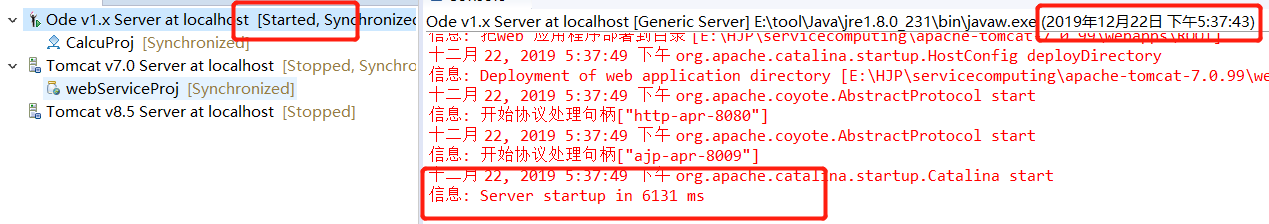
**选择classpath,点击add external JARS,到tomcat\bin目录下添加tomcat-juli.jar文件。**



**4．2 部署 BPEL 工程 CalcuProj**

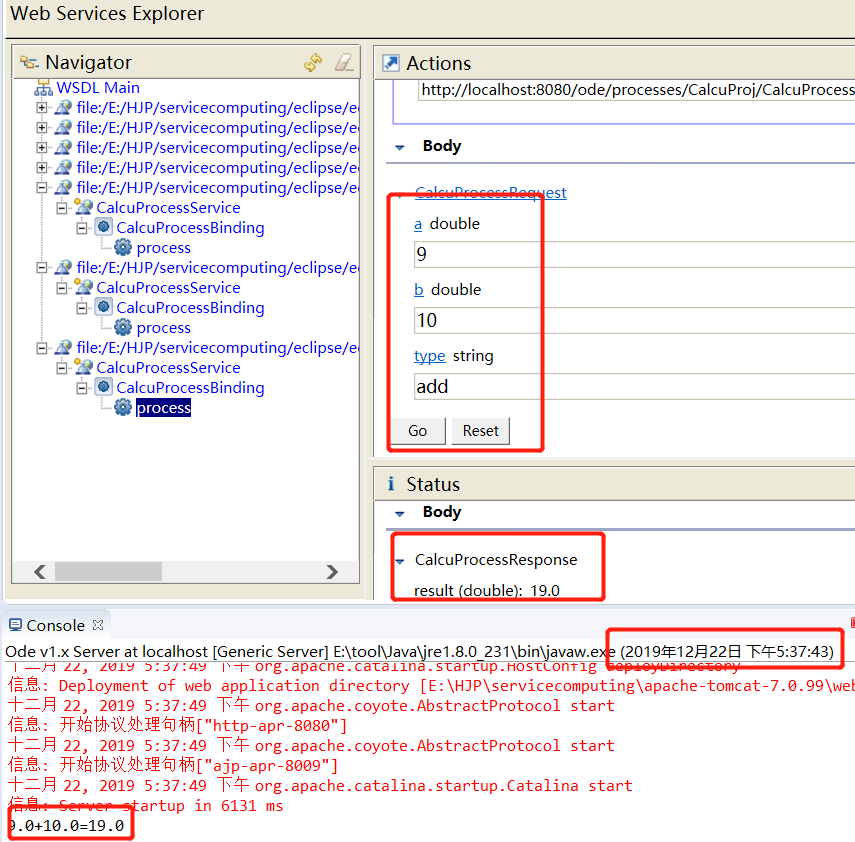


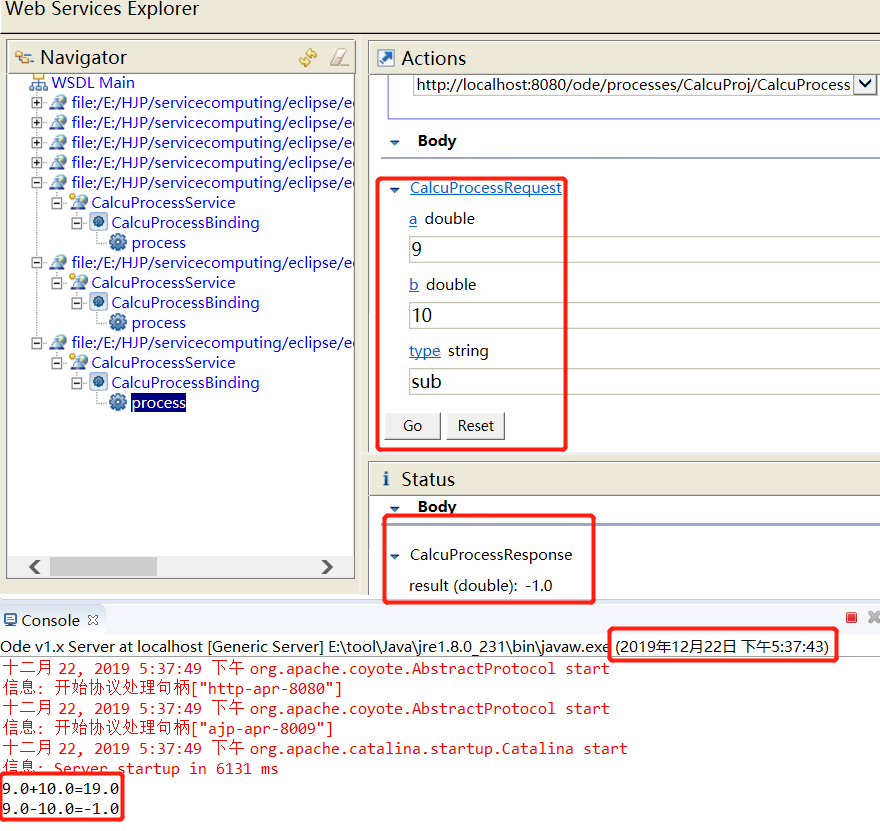
**4．3 BPEL 工程 CalcuProj 测试**启动ode服务器

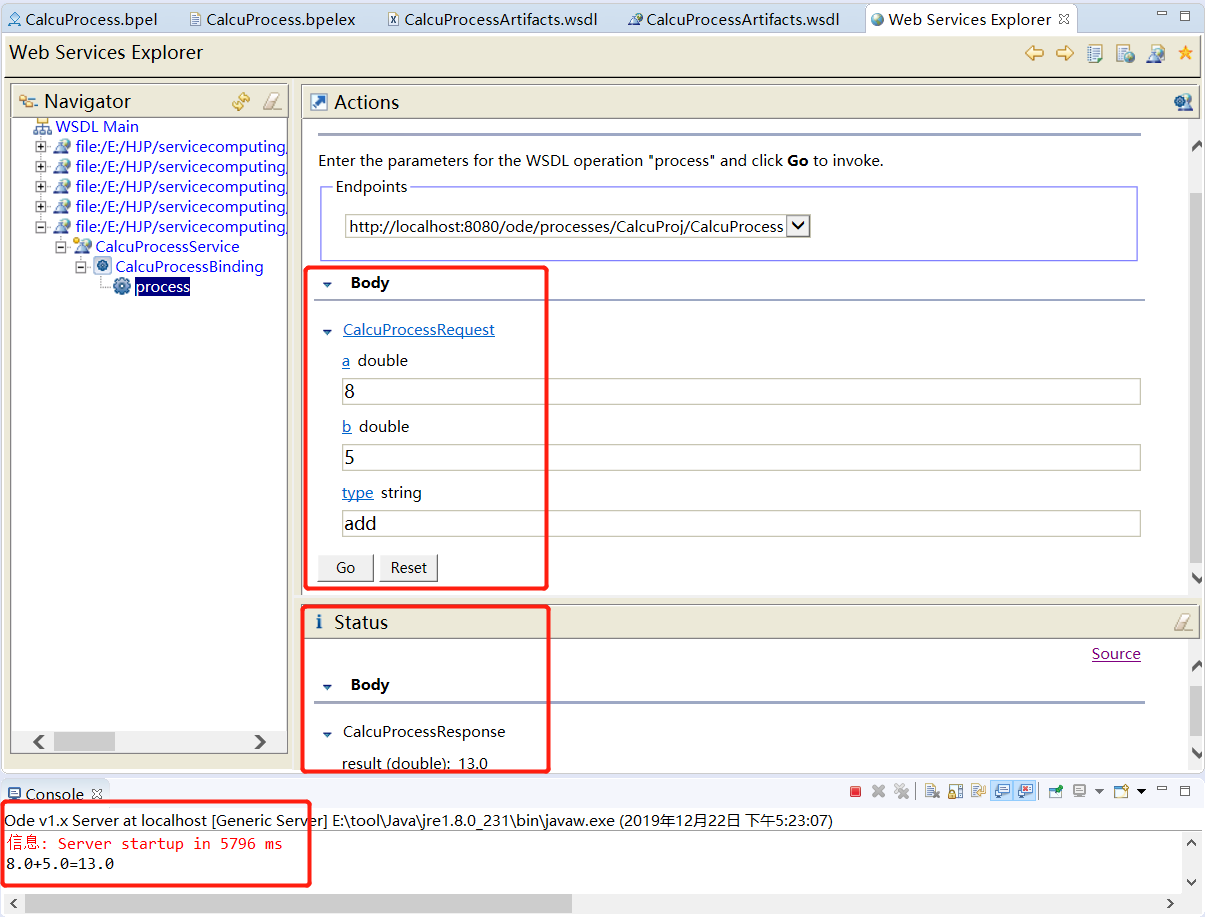


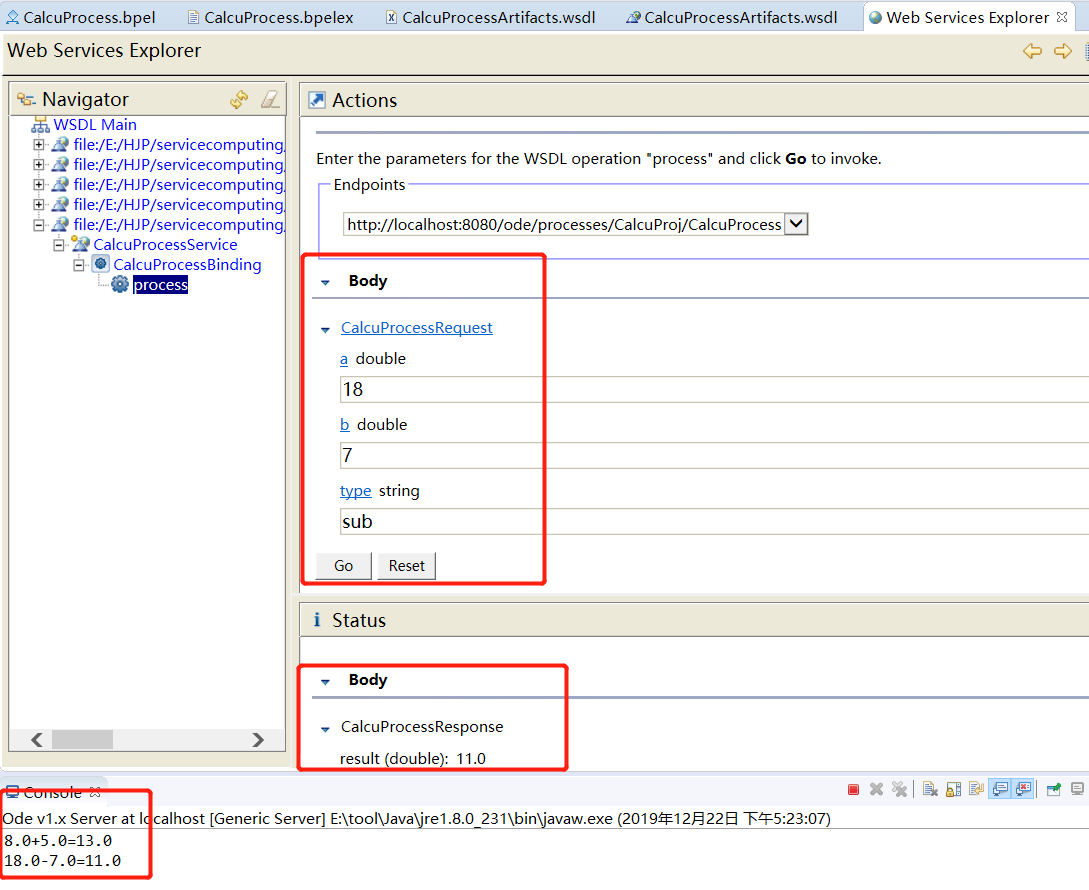
右键 CalcuProcessArtifacts.wsdl→Web Services→Test with Web Services

Explorer



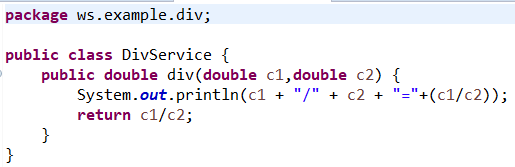


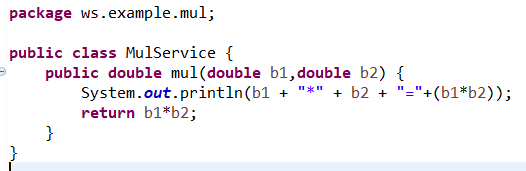


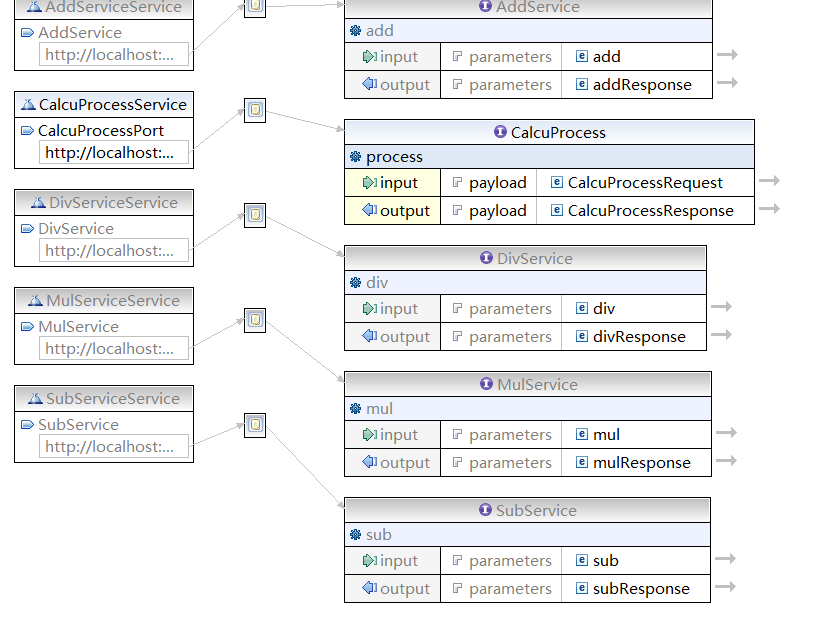


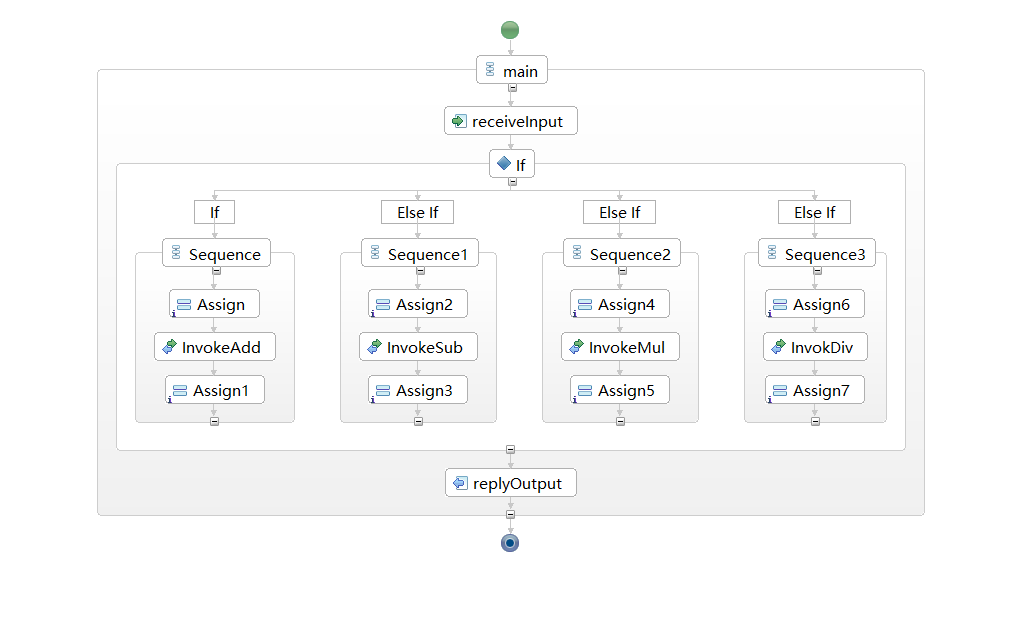
测试成功！！！

最后我们增加两个web服务完成作业要求：乘法和除法。

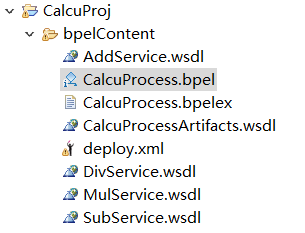






*  

最后的程序结构：



测试乘法和除法：

