目录

[【1】SOAMANAGER激活（激活WebService的工作环境） 2](#_Toc420606763)

[修改SAP的参数文件 2](#_Toc420606764)

[调整SAP服务器主机的hosts文件 2](#_Toc420606765)

[在自己机器下修改hosts文件 2](#_Toc420606766)

[SICF激活下列的服务 3](#_Toc420606767)

[WSDL文件打开的默认用户设置 6](#_Toc420606768)

[测试SOAMANAGER 7](#_Toc420606769)

[WSDL语法检查 8](#_Toc420606770)

[【2】SE37创建函数 10](#_Toc420606771)

[【3】创建WebService 11](#_Toc420606772)

[【4】如何查看创建的webservice 14](#_Toc420606773)

[【5】函数的参数发生变化处理 15](#_Toc420606774)

[新增加参数 15](#_Toc420606775)

[删除参数 17](#_Toc420606776)

[改变参数的数据类型 18](#_Toc420606777)

[修改代码无须重新生成webservice 19](#_Toc420606778)

[【6】启动SOAMANAGER完成webservice的配置 19](#_Toc420606779)

[【7】SOAPUI与JDK安装与配置 22](#_Toc420606780)

[JDK的配置 22](#_Toc420606781)

[系统变量→新建 JAVA\_HOME 变量 23](#_Toc420606782)

[修改Path系统变量 23](#_Toc420606783)

[系统变量→新建 CLASSPATH 变量 23](#_Toc420606784)

[检验JDK是否配置成功 24](#_Toc420606785)

[SOAPUI的安装 24](#_Toc420606786)

[【8】通过SOAPUI调用WEBSERVICE 24](#_Toc420606787)

[【9】SAP调用WebService 26](#_Toc420606788)

[SAP创建消费类 26](#_Toc420606789)

[SE24查看消费类信息 31](#_Toc420606790)

[通过SOAMANAGER配置端口信息 32](#_Toc420606791)

[编写ABAP程序调用WebService 36](#_Toc420606792)

[【10】SAP调用外部系统的WebService(非SAP) 38](#_Toc420606793)

[SAP创建消费类 38](#_Toc420606794)

[通过SOAMANAGER配置端口信息 42](#_Toc420606795)

[编写ABAP程序调用外部WebService 46](#_Toc420606796)

[【11】调用SAP WebService的日志监控 46](#_Toc420606797)

# 【1】SOAMANAGER激活（激活WebService的工作环境）

## 修改SAP的参数文件

事务代码 RZ10进入下面界面:

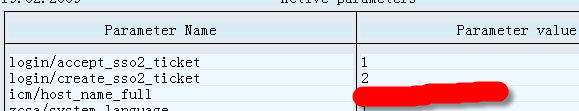


图2－1　修改参数

这里要注意一点的是icm/host\_name\_full必须是形如<computer name>.<domain>.<ext>，例如mysap.com.cn。

## 调整SAP服务器主机的hosts文件

添加如下一行

xxx.xxx.xxx.xxx mysap.com.cn mysap

如果你的服务器在域中，并且你的客户端也在域中，直接可以通过上面的全名访问到服务器的话，下面这步可以省略。

## 在自己机器下修改hosts文件

一般就位于c:\windows\system32\drivers\etc下添加

xxx.xxx.xxx.xxx mysap.com.cn mysap

## SICF激活下列的服务

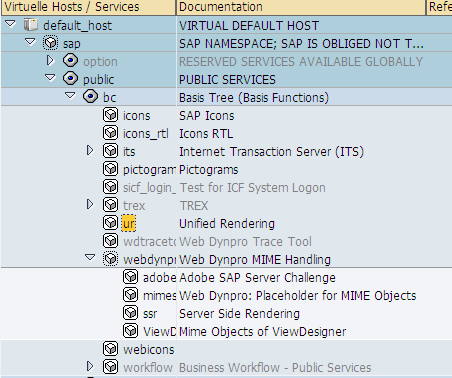


图2－2　default\_host/sap/public/bc

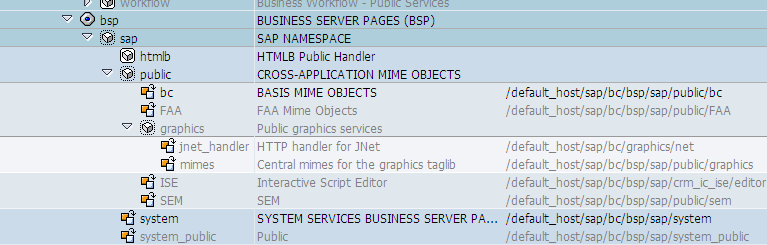


图2－3　default\_host/sap/public/bsp

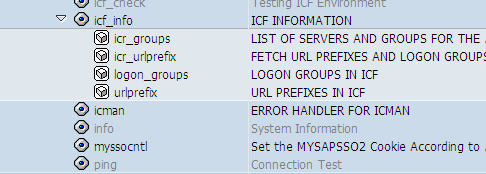


图2－4　default\_host/sap/public

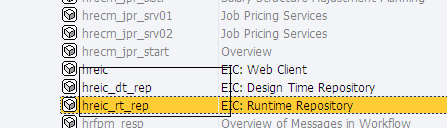


图2－6　default\_host/sap/bc/bsp/sap I

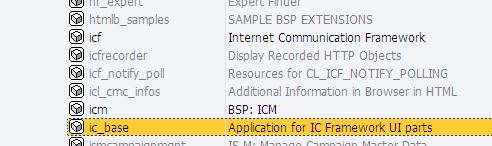


图2－5　default\_host/sap/bc/bsp/sap II

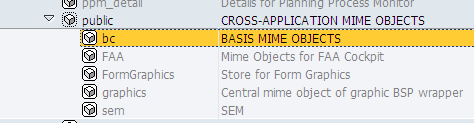


图2－6　default\_host/sap/bc/bsp/sap III

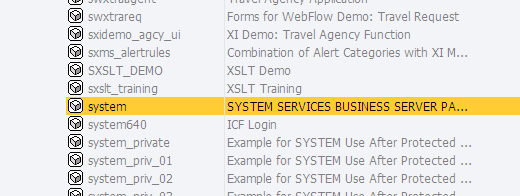


图2－7　default\_host/sap/bc/bsp/sap IV

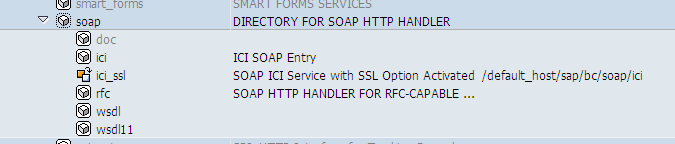


图2－8　default\_host/sap/bc/soap

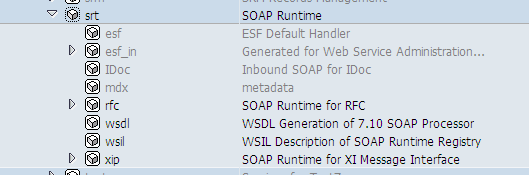


图2－9　default\_host/sap/bc/srt

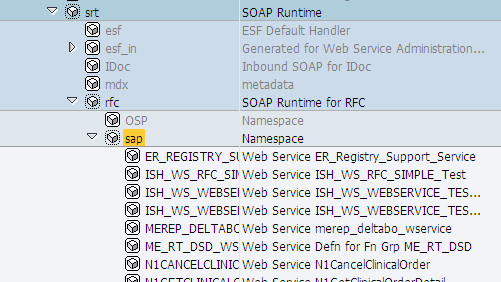


图2－10　default\_host/sap/bc/srt/rfc/sap

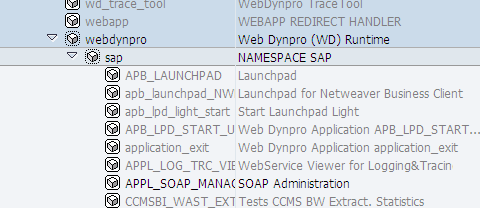


图2－11　default\_host/sap/bc/webdynpro/sap

## WSDL文件打开的默认用户设置

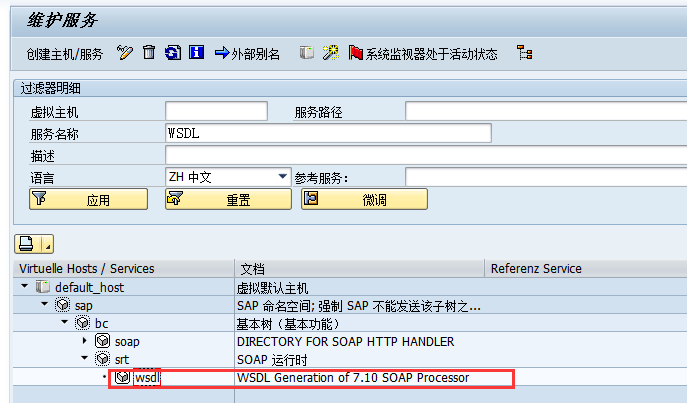


图2－12　default\_host/sap/bc/srt/wsdl

双击上图进入如下界面填入所对应的用户和密码



## 测试SOAMANAGER

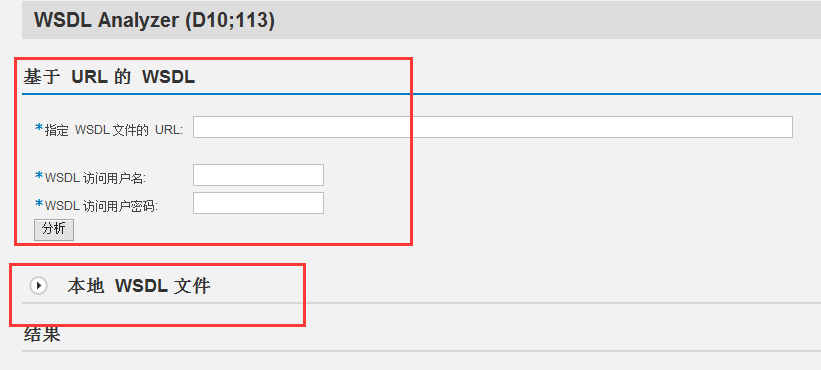
在对应的客户端运行TCODE:SOAMANAGER，SAPGUI会自动帮你找开一个浏览器（IE），这个界面会因版本的不同而不同，包括里面的操作界面也会因版本不同而不同

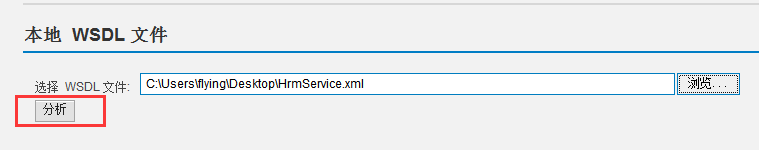


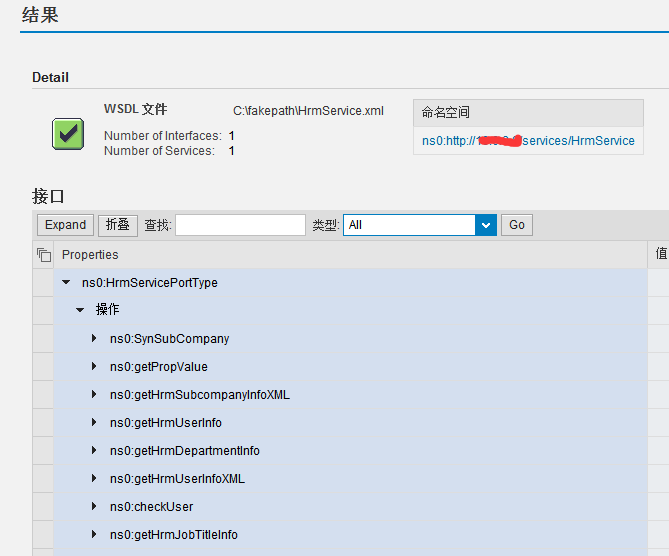
## WSDL语法检查

SAP对于外部WSDL的要求比较，对于调用外部WebServices的情况，可以调用这个功能检查WSDL文件是否满足要求，这个工具能检测到一些语法错误的情况



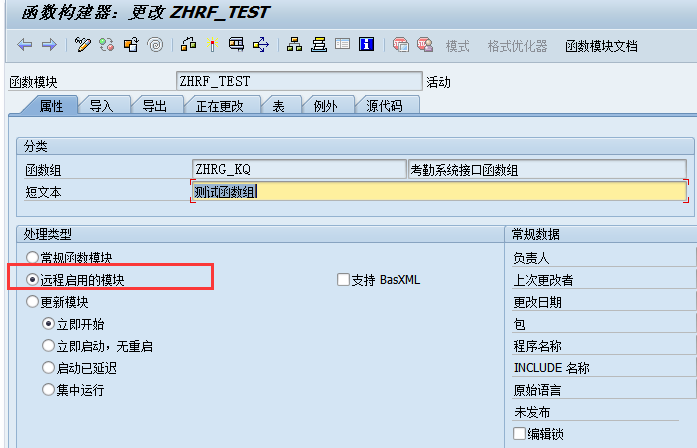




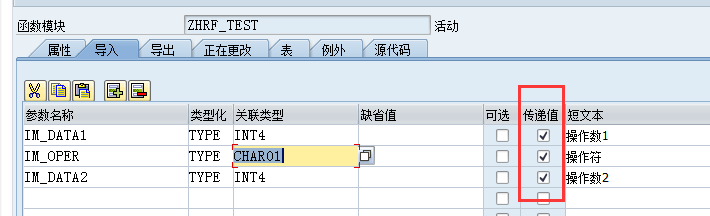


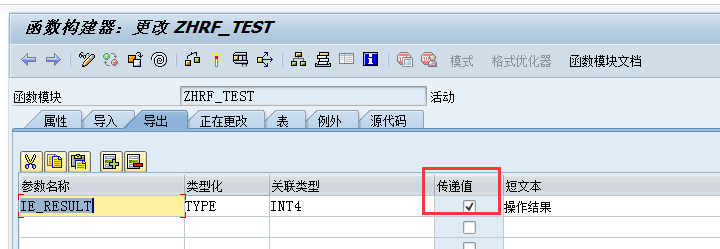
# 【2】SE37创建函数

注意必须是远程启用的模块

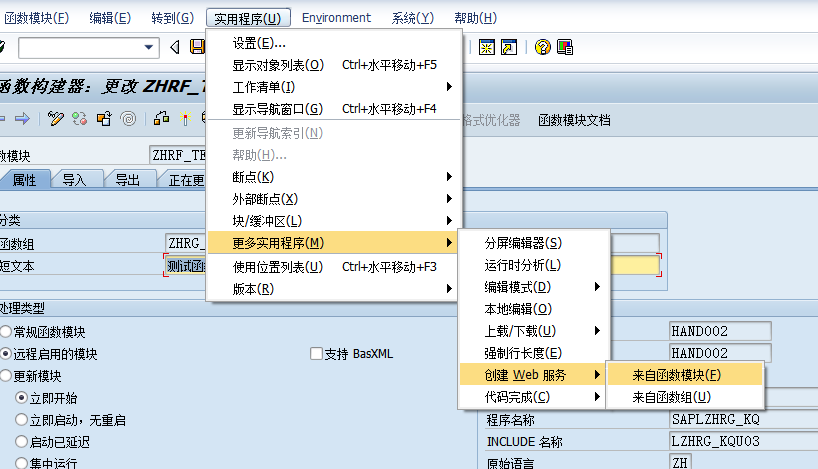


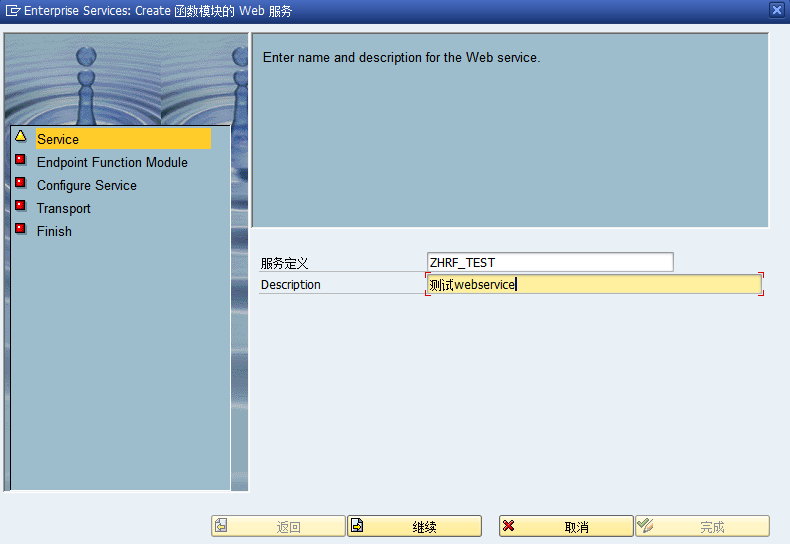
注意：对于RFC函数参数必须传递值



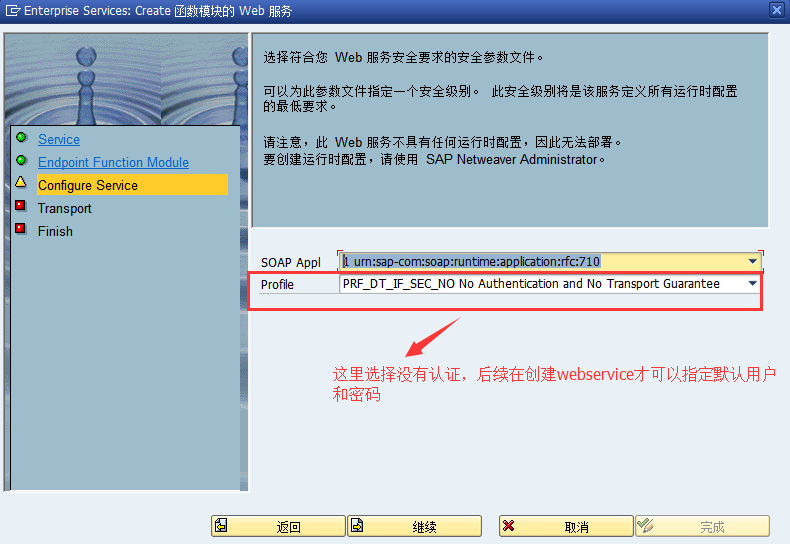


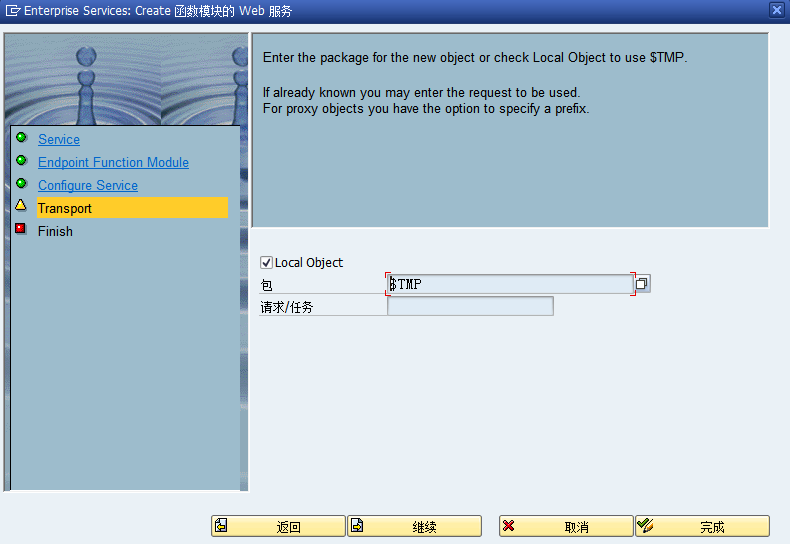
# 【3】创建WebService

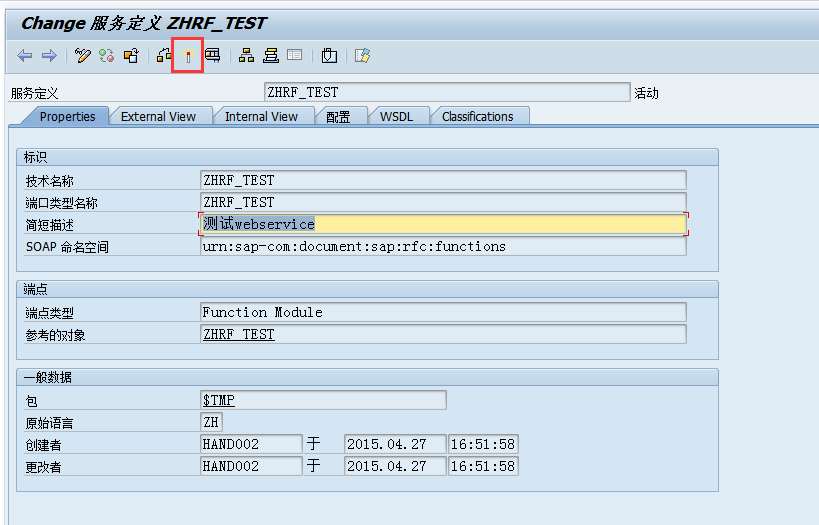


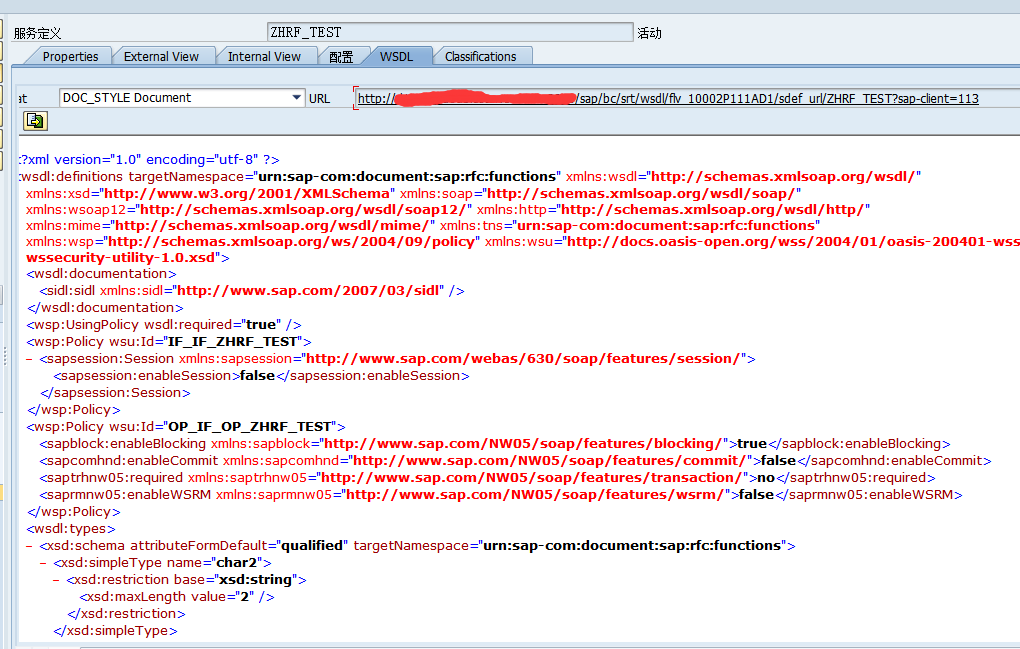




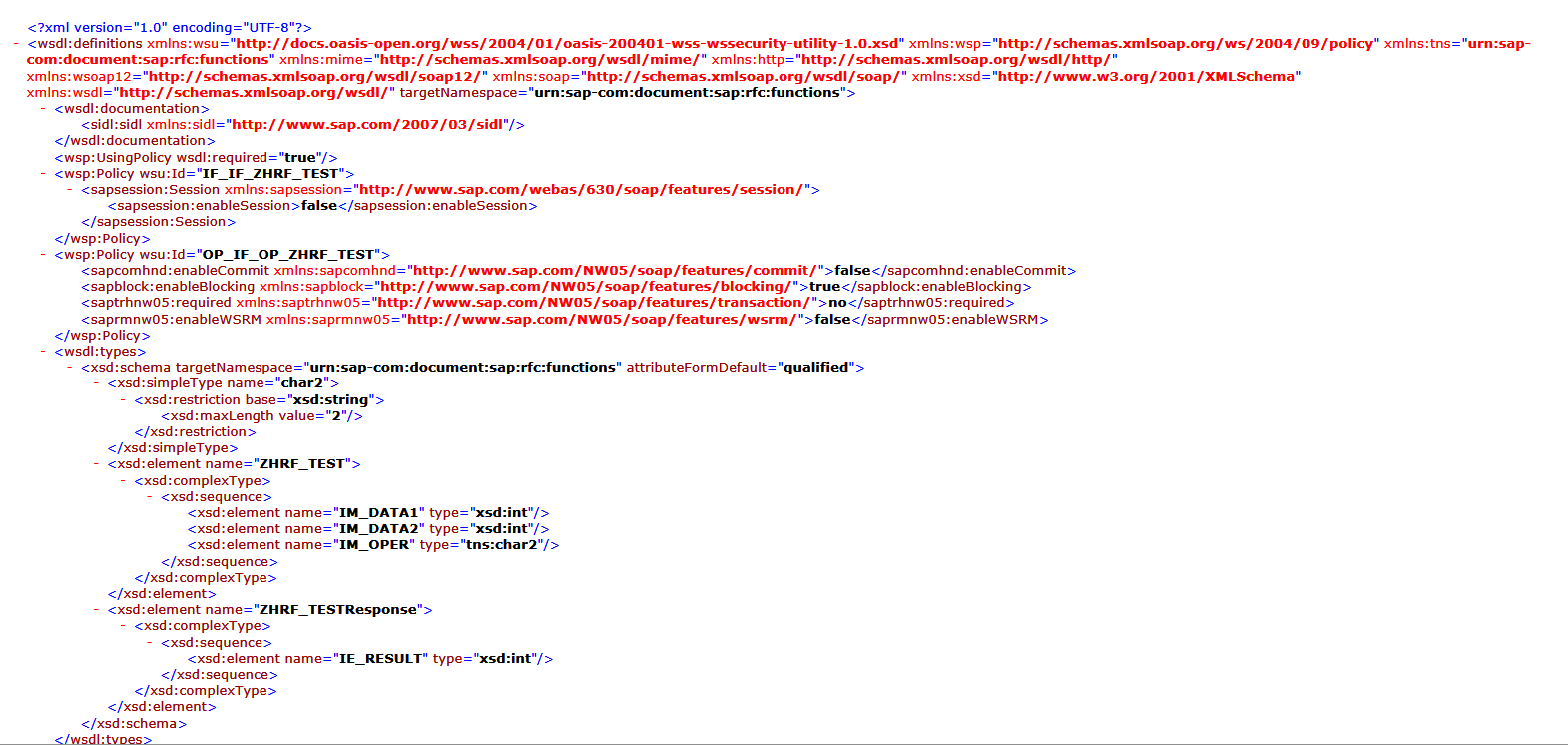






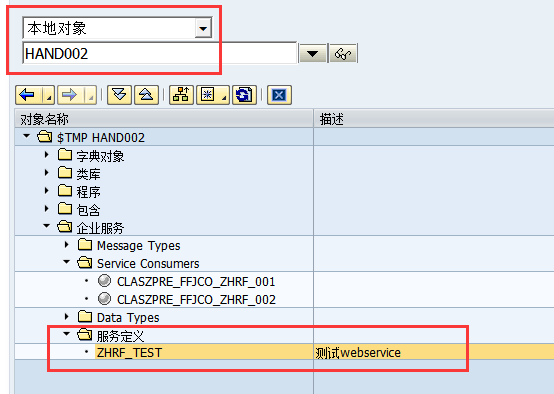


将上图的URL（WSDL）贴到IE上打开，如果打开如下图所示，表示成功



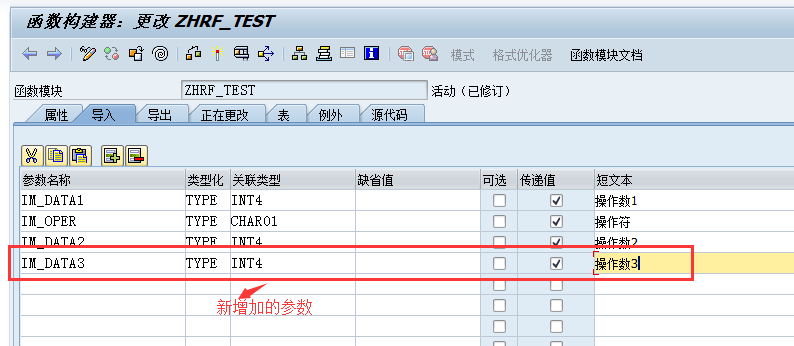
# 【4】如何查看创建的webservice

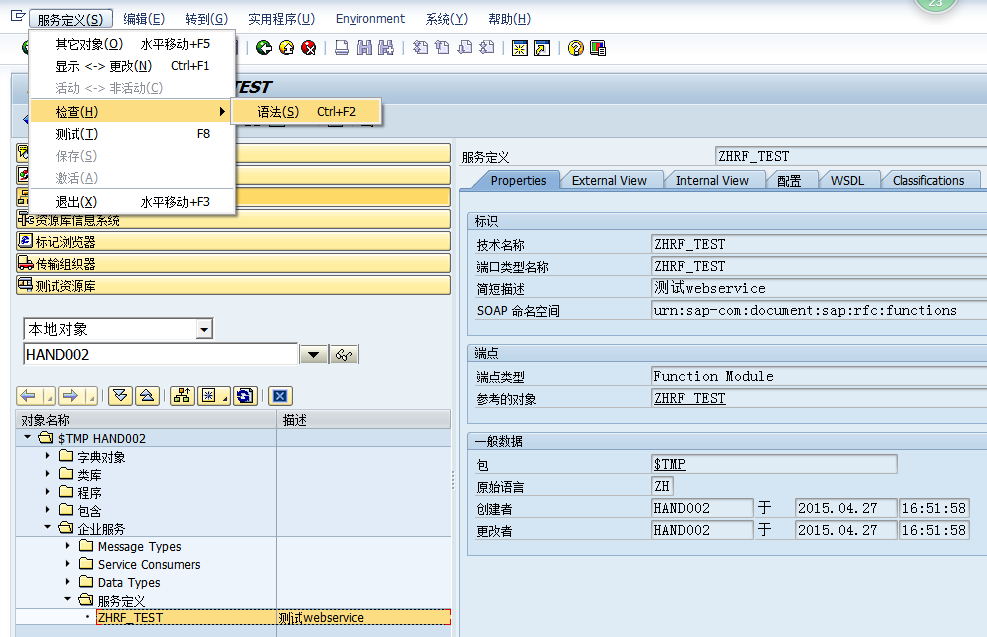
通过SE80，选择自己的开发包或者本地对象，点开服务定义就可以看到对应的服务了

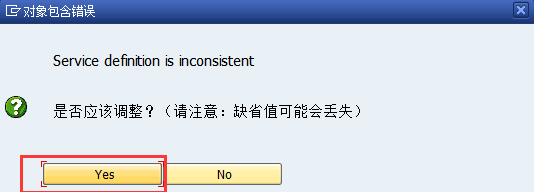


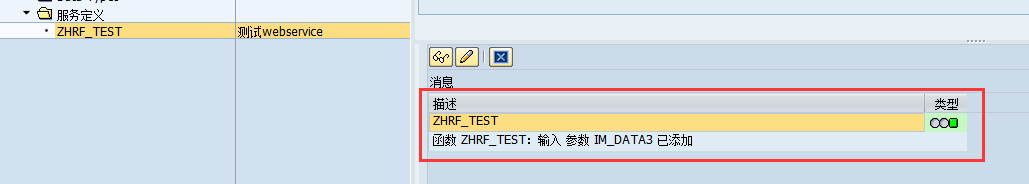
# 【5】函数的参数发生变化处理

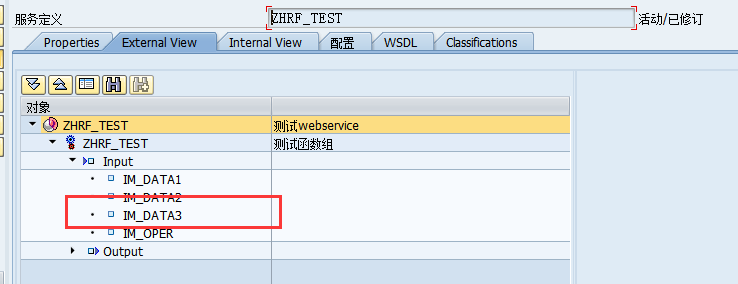
## 新增加参数





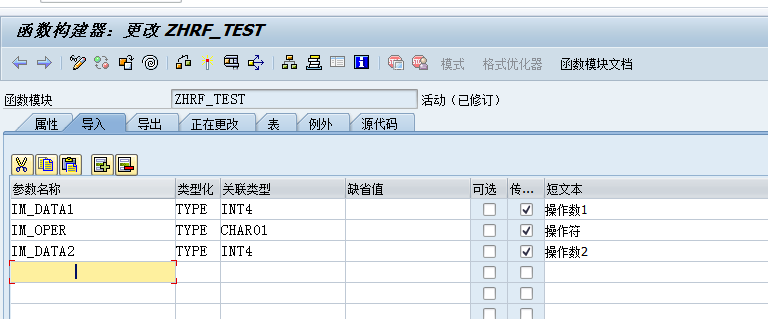


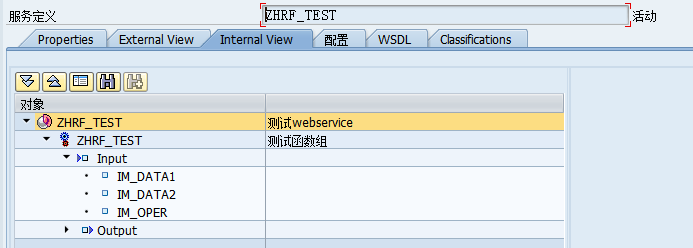




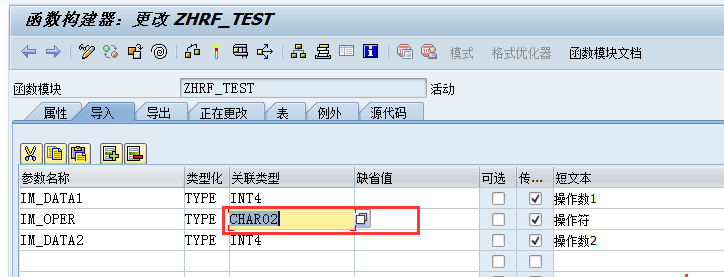
## 删除参数

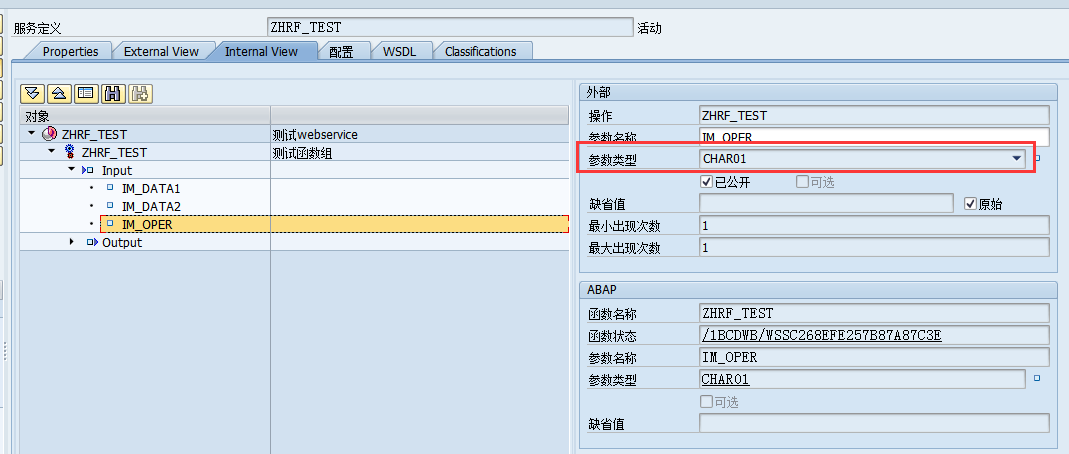
删除参数会自动同步到服务，无须手动同步

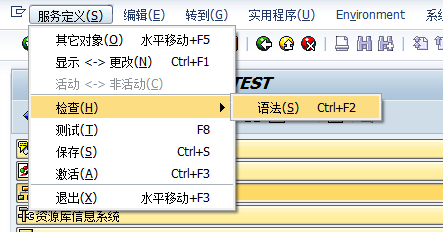


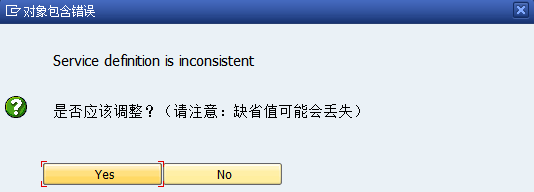


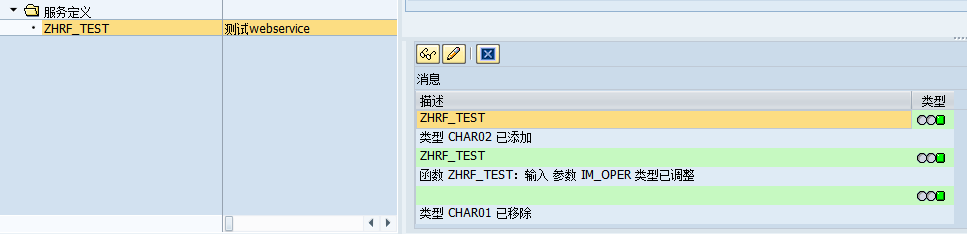
## 改变参数的数据类型











## 修改代码无须重新生成webservice

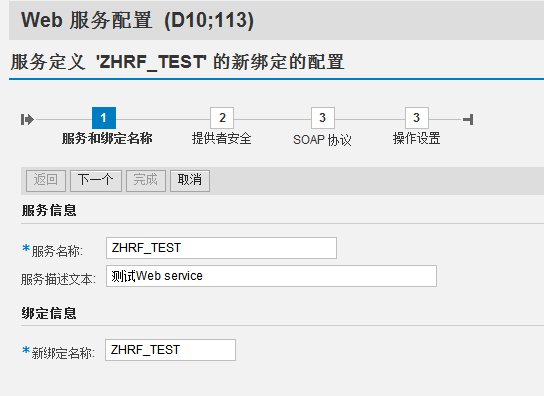
# 【6】启动SOAMANAGER完成webservice的配置

以下操作会因版本的不同而有所不同

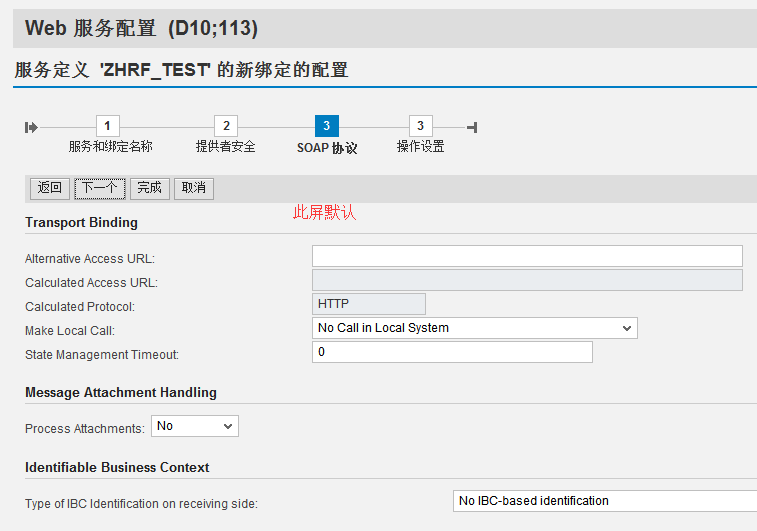


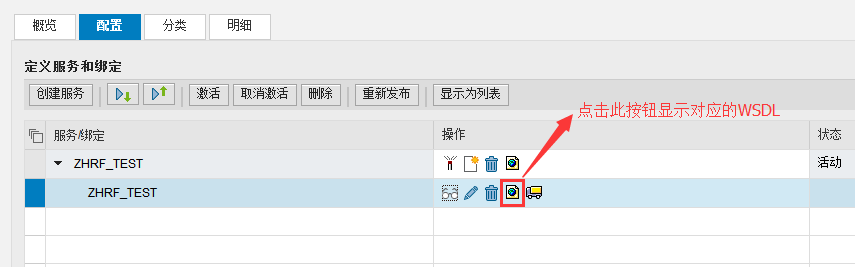


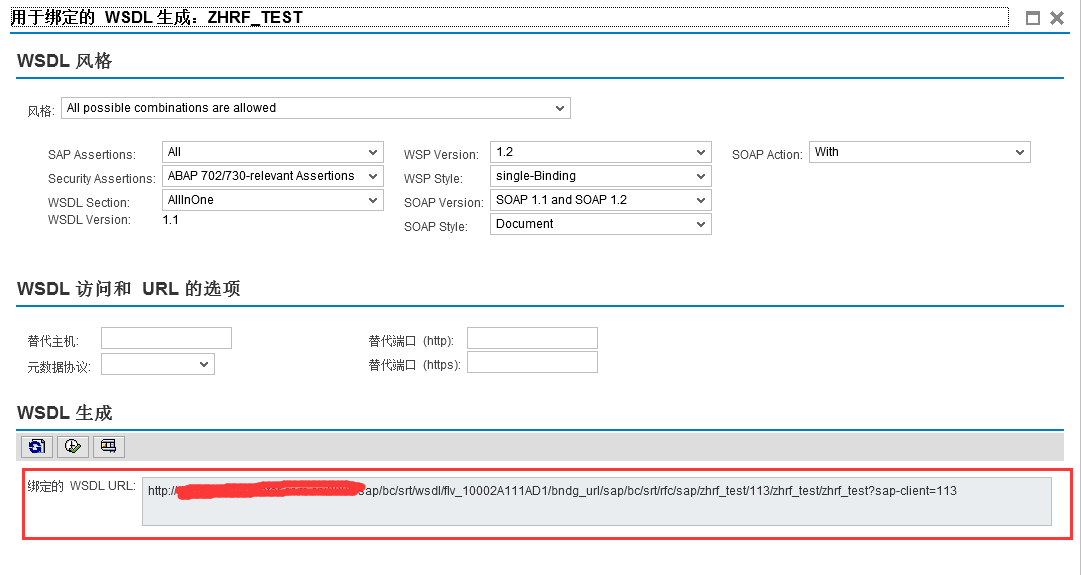








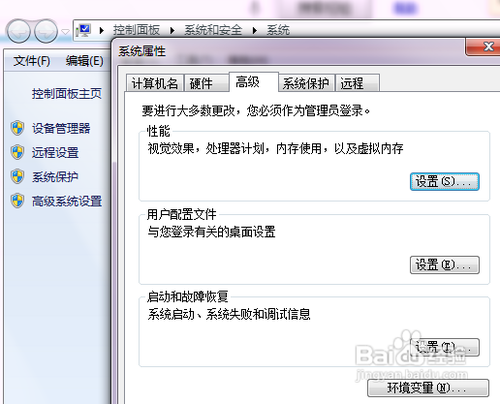




# 【7】SOAPUI与JDK安装与配置

## JDK的配置

计算机→属性→高级系统设置→高级→环境变量

[](http://jingyan.baidu.com/album/6dad5075d1dc40a123e36ea3.html?picindex=4)

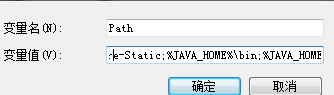
### 系统变量→新建 JAVA\_HOME 变量

变量值填写jdk的安装目录（本人是 E:\Java\jdk1.7.0)

### 修改Path系统变量

在变量值最后输入 %JAVA\_HOME%\bin;%JAVA\_HOME%\jre\bin;

（注意原来Path的变量值末尾有没有;号，如果没有，先输入；号再输入上面的代码）

[](http://jingyan.baidu.com/album/6dad5075d1dc40a123e36ea3.html?picindex=5)

### 系统变量→新建 CLASSPATH 变量

变量值填写   .;%JAVA\_HOME%\lib;%JAVA\_HOME%\lib\tools.jar（注意最前面有一点）

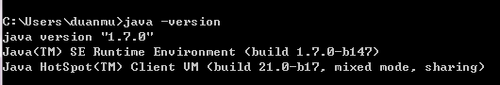
系统变量配置完毕

[](http://jingyan.baidu.com/album/6dad5075d1dc40a123e36ea3.html?picindex=6)

### 检验JDK是否配置成功

运行cmd 输入 java -version （java 和 -version 之间有空格）

若如图所示 显示版本信息 则说明安装和配置成功。

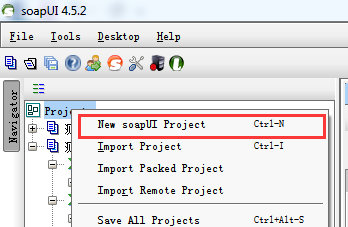
[](http://jingyan.baidu.com/album/6dad5075d1dc40a123e36ea3.html?picindex=7)

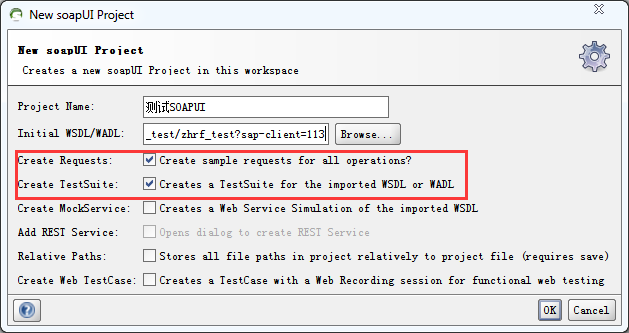
## SOAPUI的安装

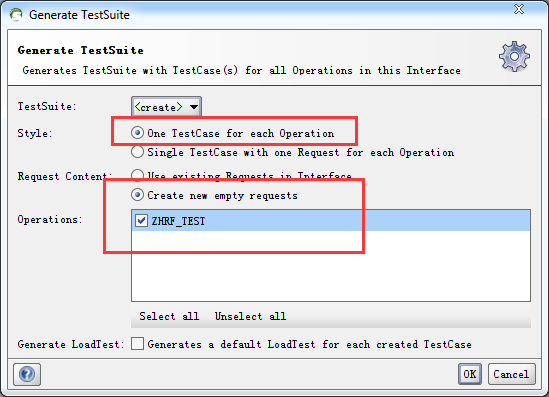
一般只须将下载的SOAPUI文件解压出来就可以用了

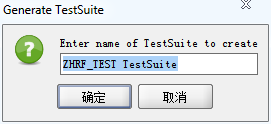
# 【8】通过SOAPUI调用WEBSERVICE

运行SOAPUI->BIN文件夹下的SOAPUI的批处理文件，点击新建Project

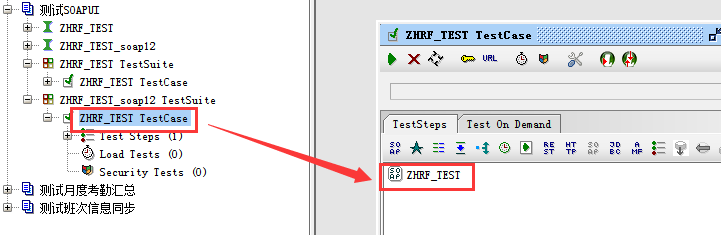




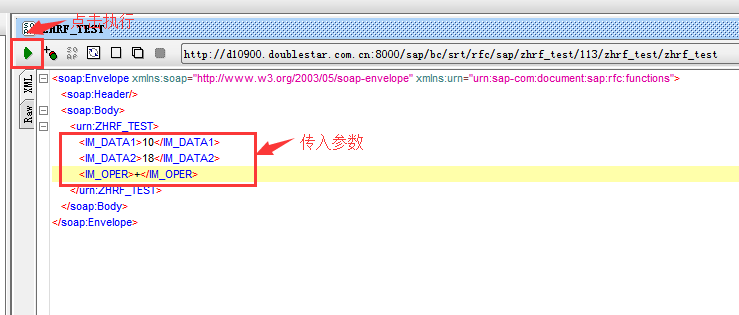




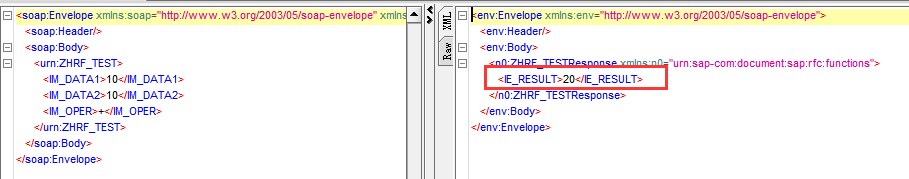
双击ZHR\_TEST TestCase



双击ZHRF\_TEST，输入传入参数



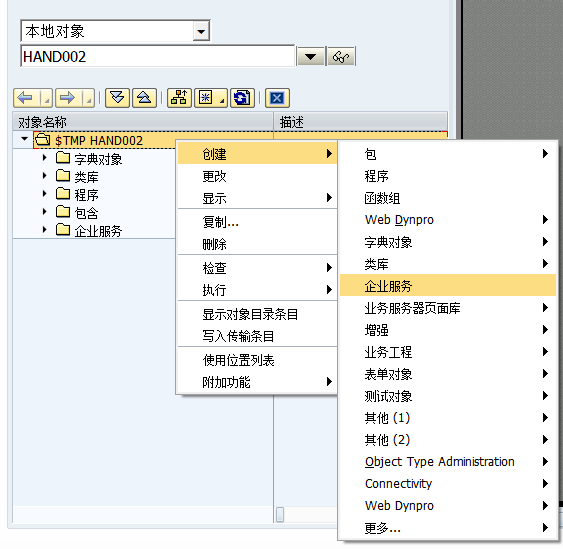
点击执行返回的结果如下



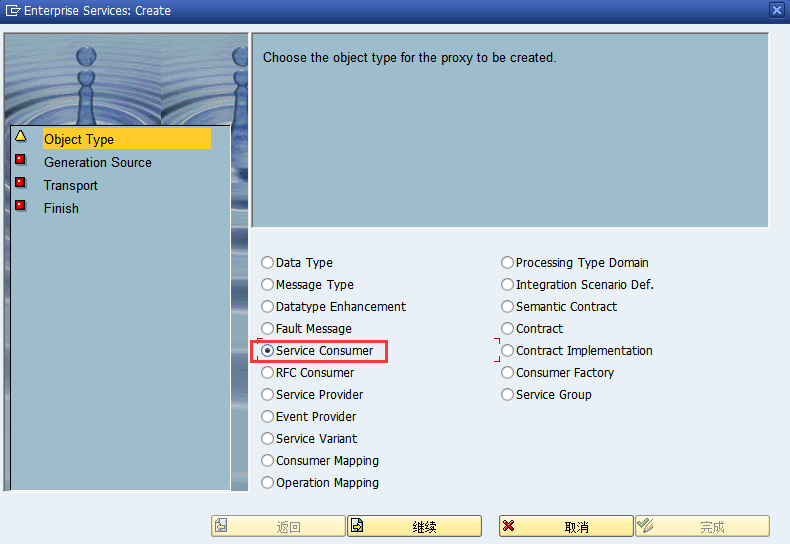
# 【9】SAP调用WebService

## SAP创建消费类

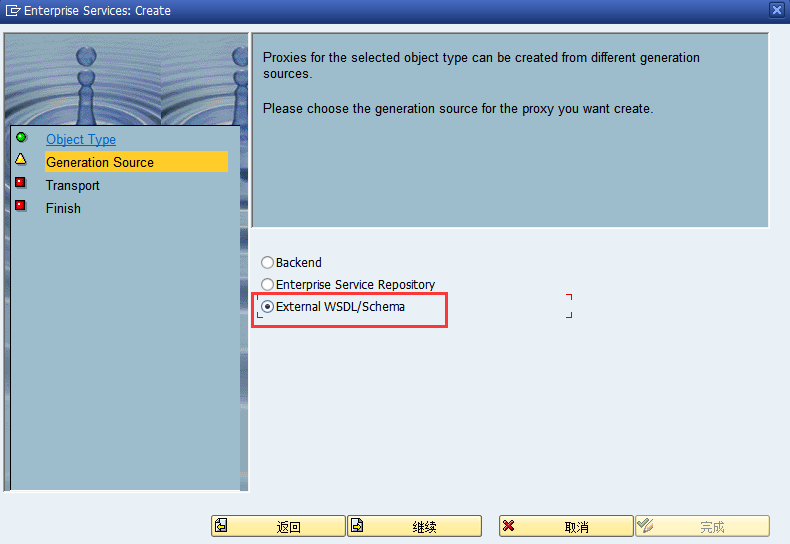
执行SE80，选择对应的开发包或者本地对象



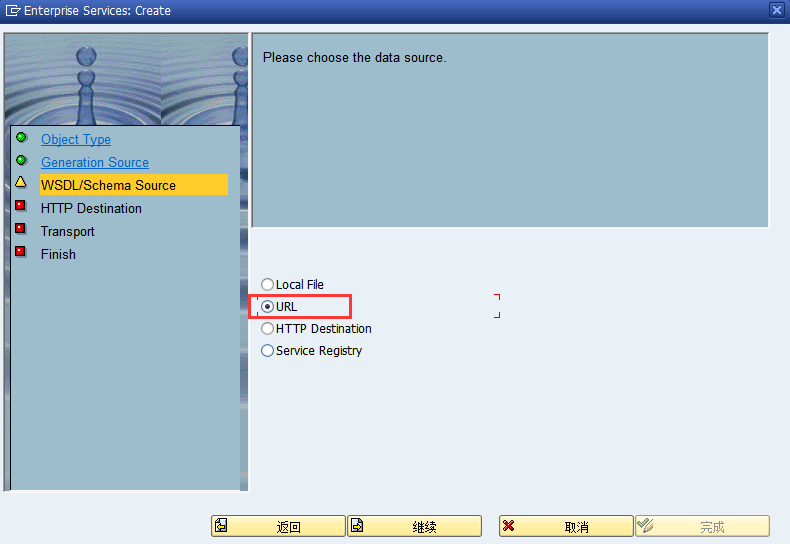
选择企业服务，选择对应对象类型



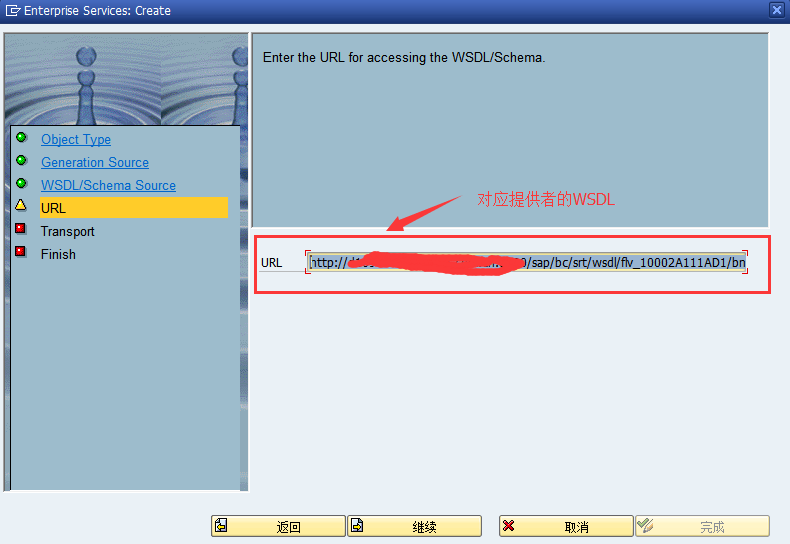
选择生成源对象



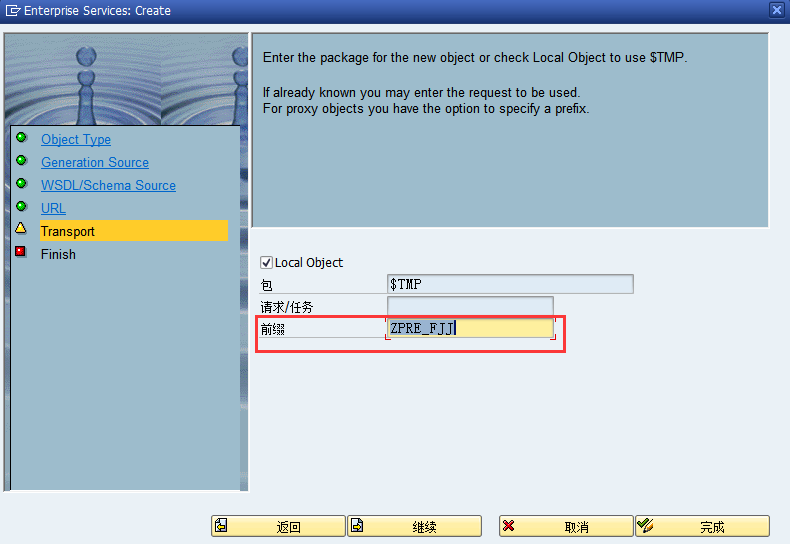
选择URL



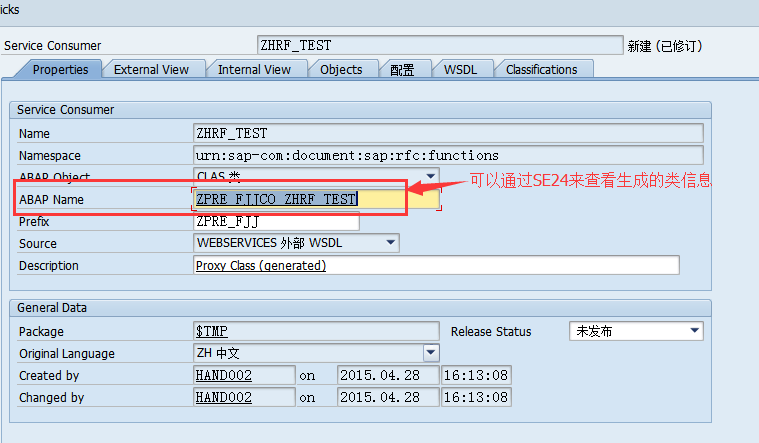
填入提供者WebService的WSDL



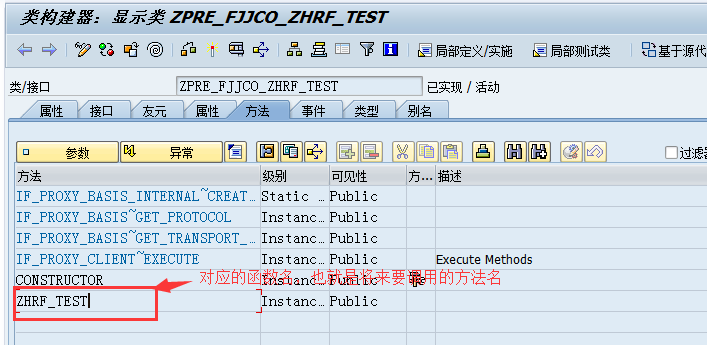
根据实际情况选择对应的传输包和前缀

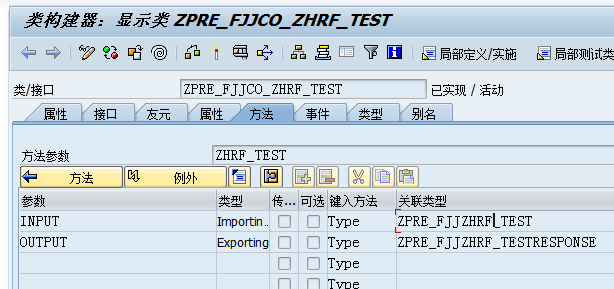


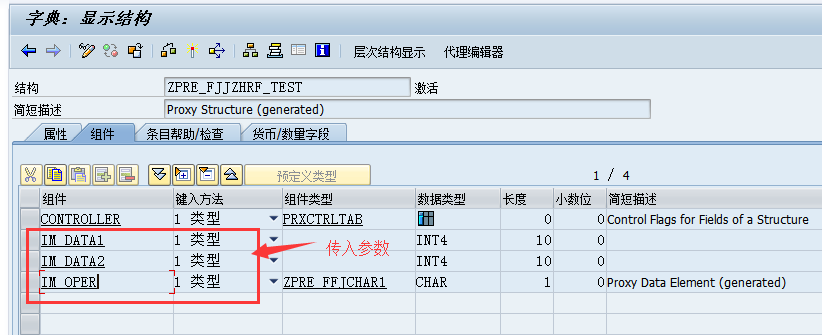
最后点击完成生成系统会根据WSDL生成对应的消费类，同时需要做激活操作

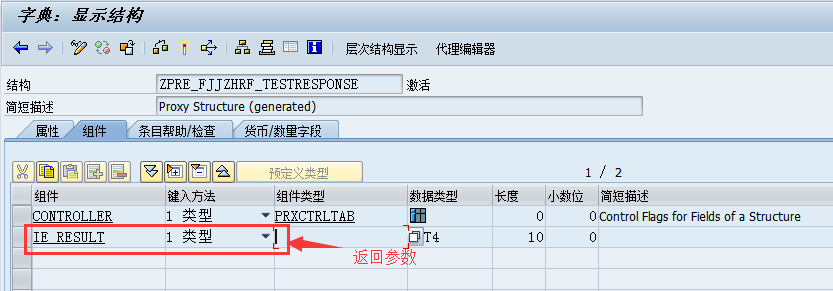


## SE24查看消费类信息







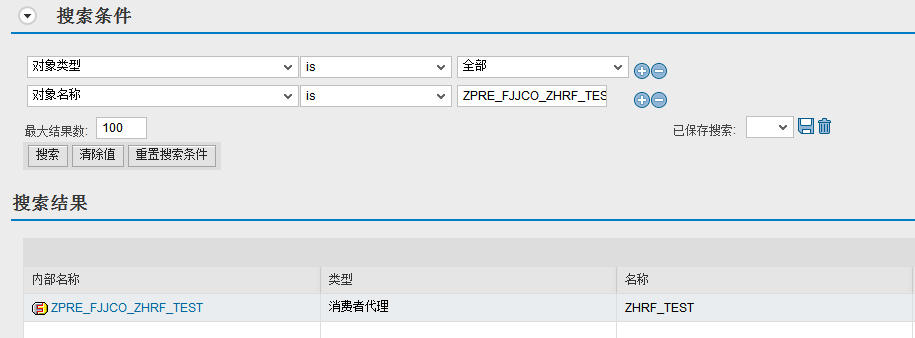


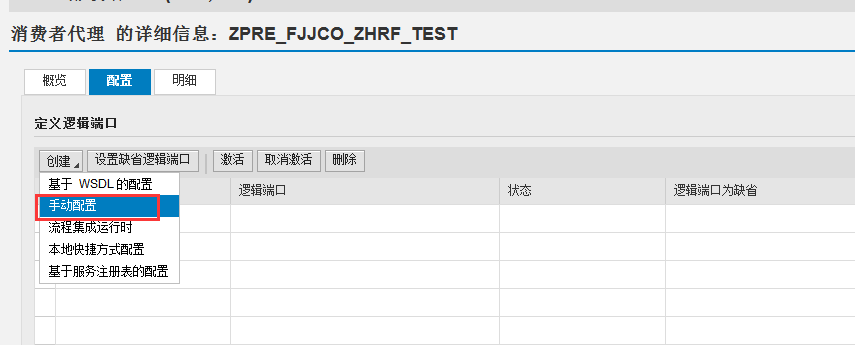
## 通过SOAMANAGER配置端口信息

**执行SOAMANAGER**



**选择WEB服务配置，输入消费类名进行查找**



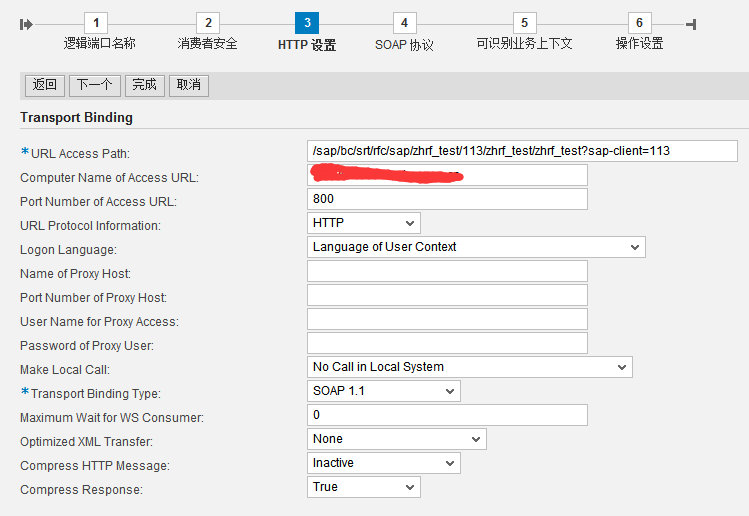


**选择创建下的手动配置，输入端口的相关信息，并设置为缺省**



**输入默认的用户和密码**





**上图的URL Access Path（不是WSDL）,可以通过如下方式查找**

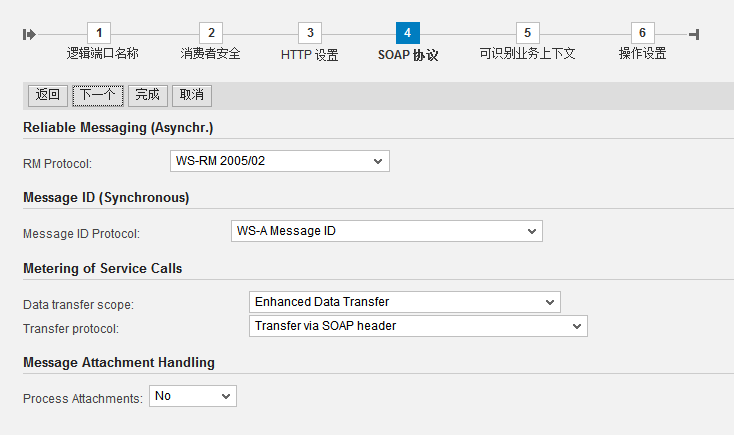
**执行SICF，输入提供者的服务名**



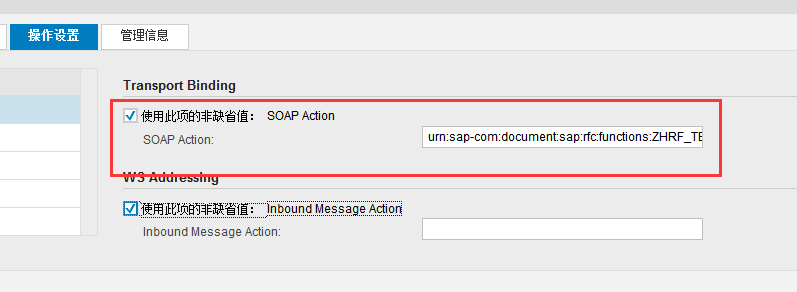
**然后右键点击测试服务，将IE地址端口后的那串填充到URL那栏**



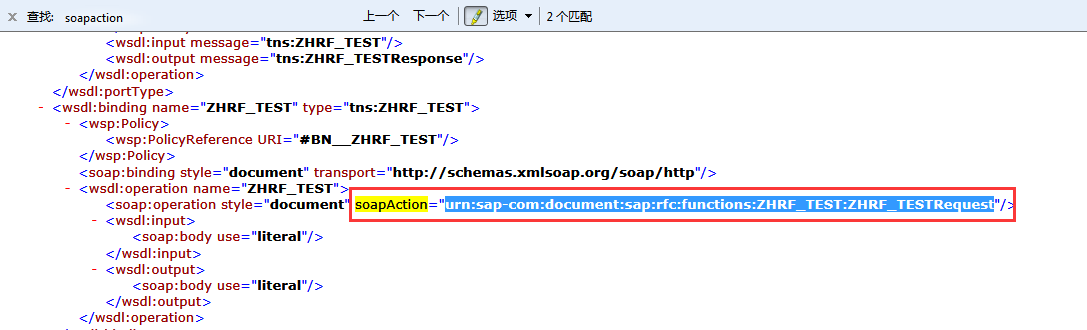
**以下屏幕默认**



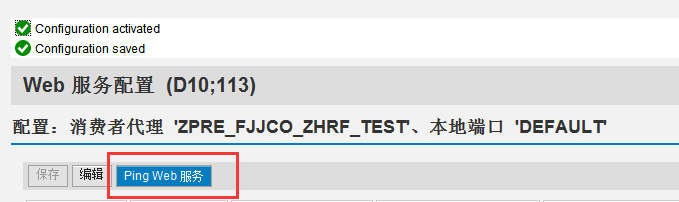


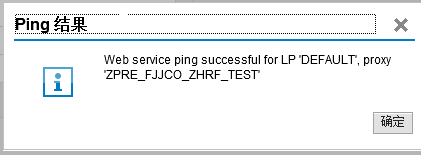


**其中SOAP Action可以通过WSDL文件中查找到**



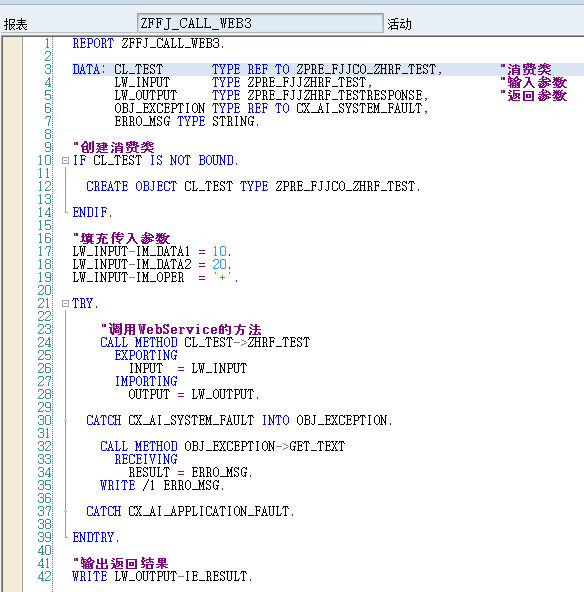
**可以通过点击按钮测试服务是否配通**





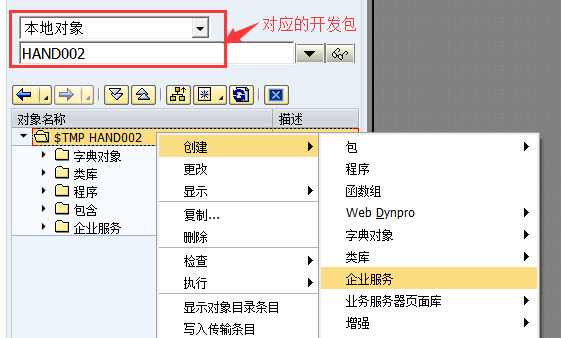
## 编写ABAP程序调用WebService

**执行SE38，新建程序测试**

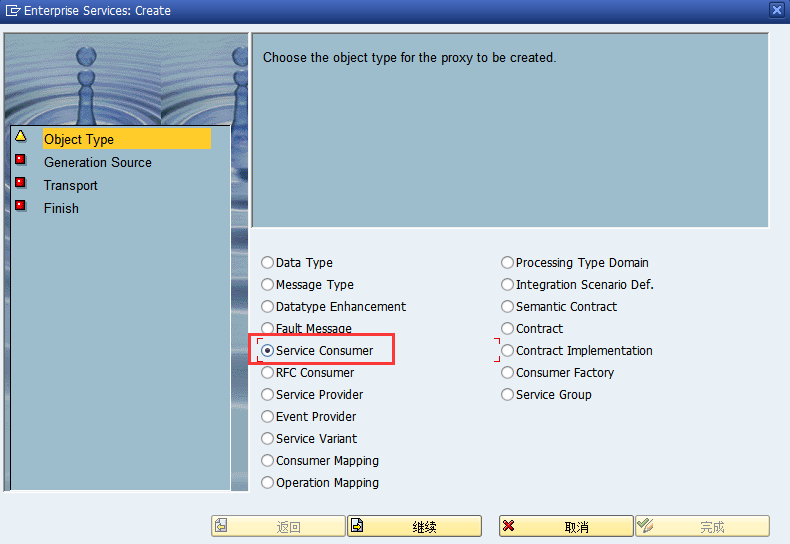


# 【10】SAP调用外部系统的WebService(非SAP)

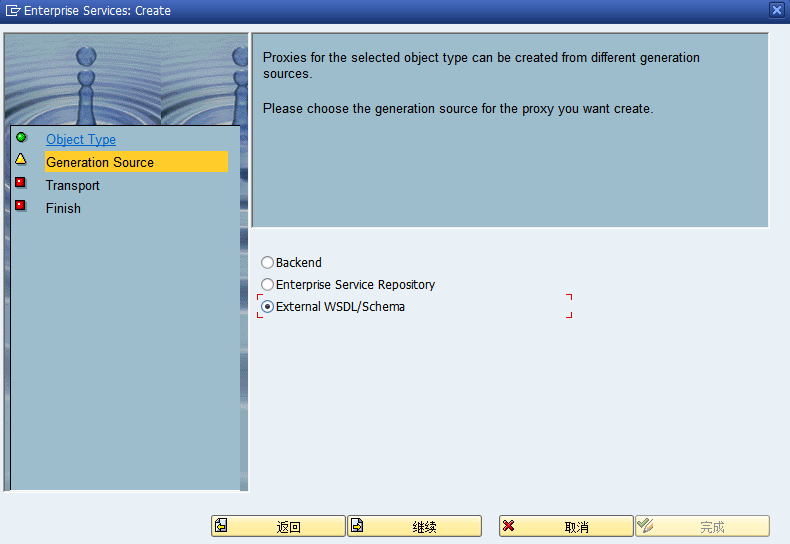
## SAP创建消费类



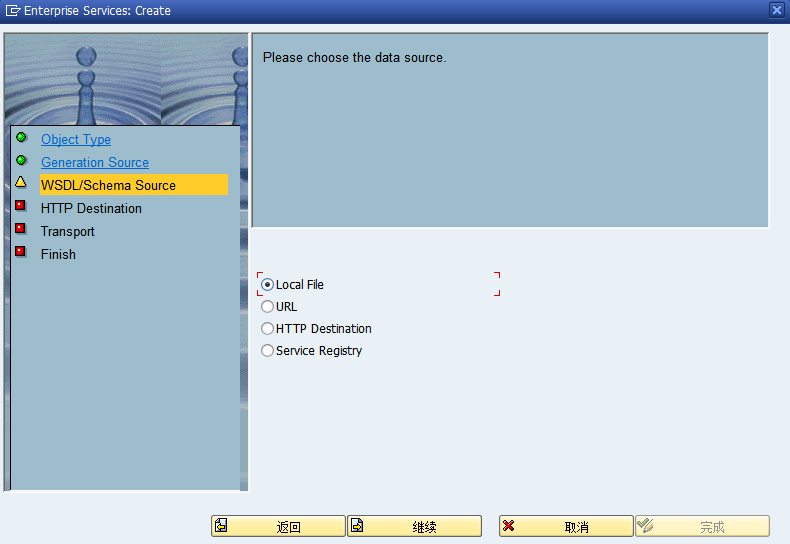
**选择Service Consumer类型**



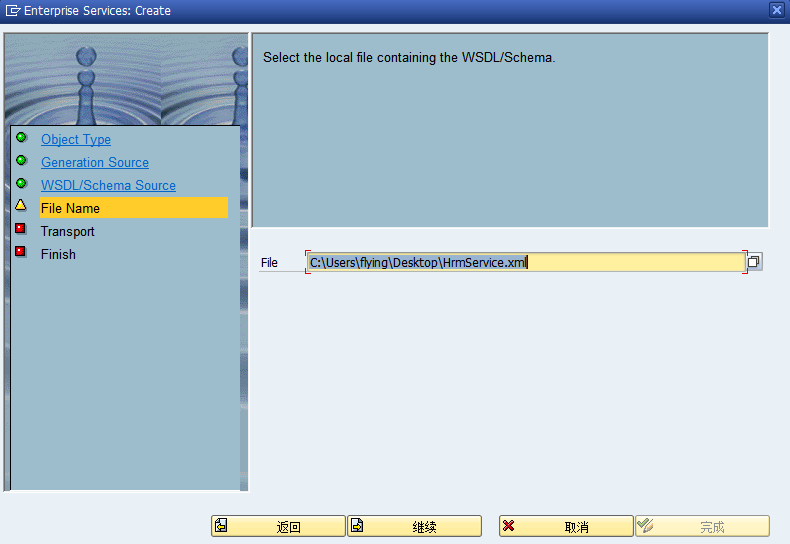
**选择外部WSDL**



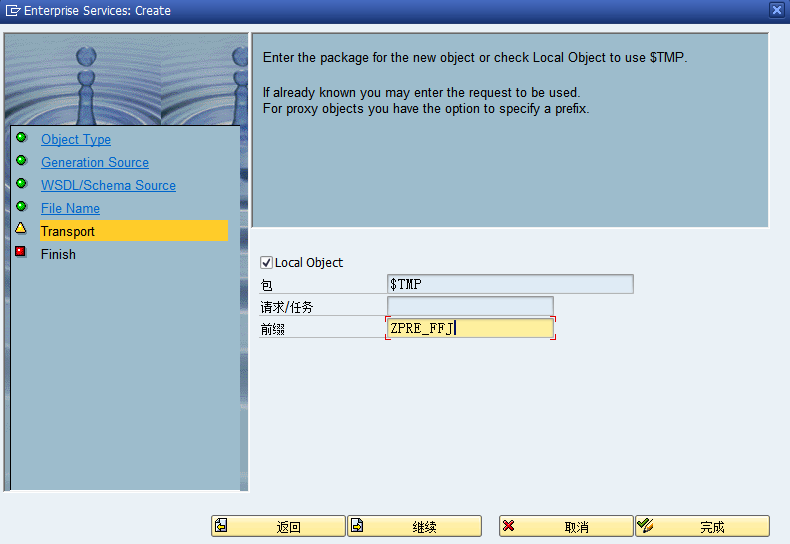
**这里可以一般选择Local File或者URL，由于SAP系统对外部WS要求比较高，一般都要手工调整WSDL文件用于支持SAP系统，如果是文件导入的情况可以通过**[**WSDL文件检查**](#_WSDL语法检查)**来检查WSDL是否满足SAP**

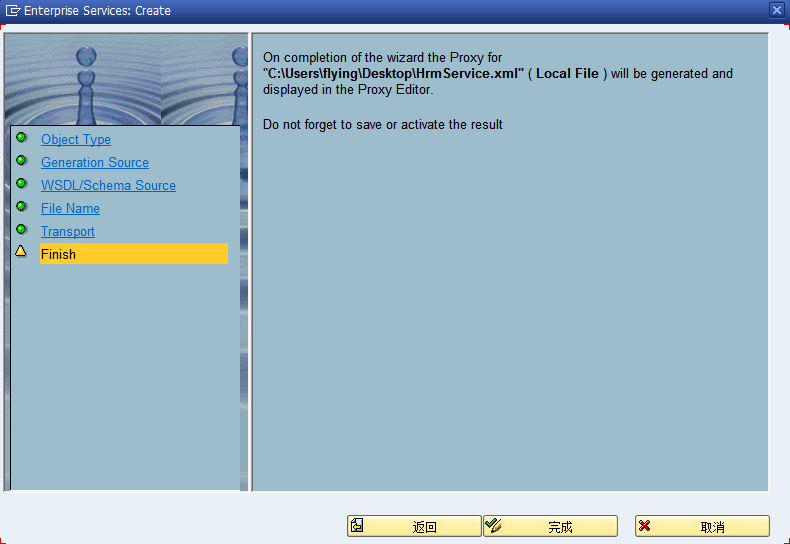


**选择WSDL文件路径**

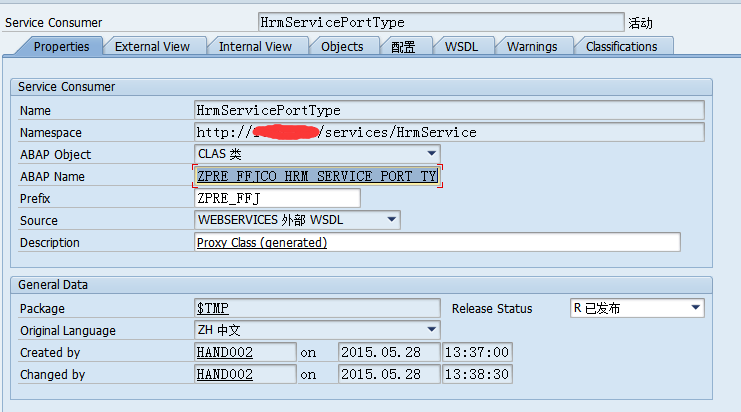


**选择生成对象的开发包**



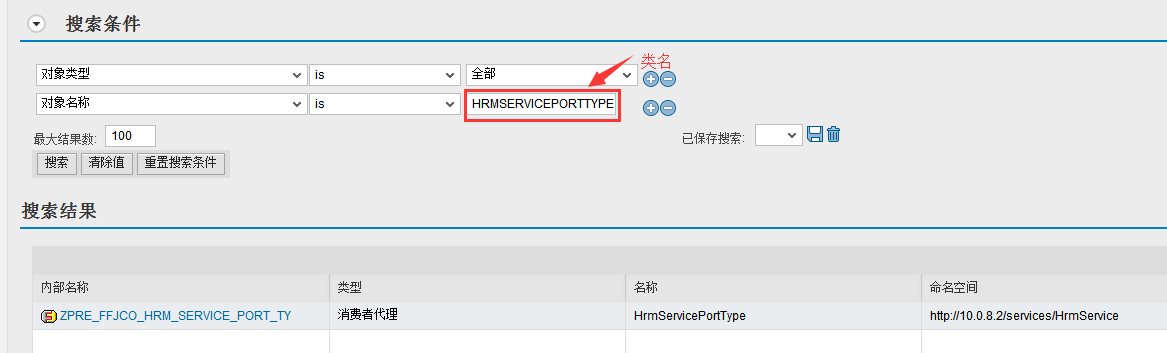


**在此界面激活生成对象**

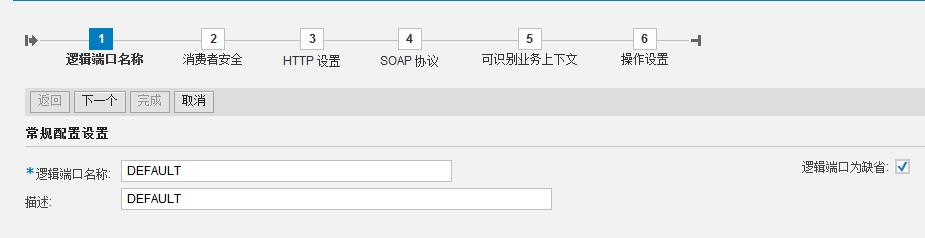


## 通过SOAMANAGER配置端口信息

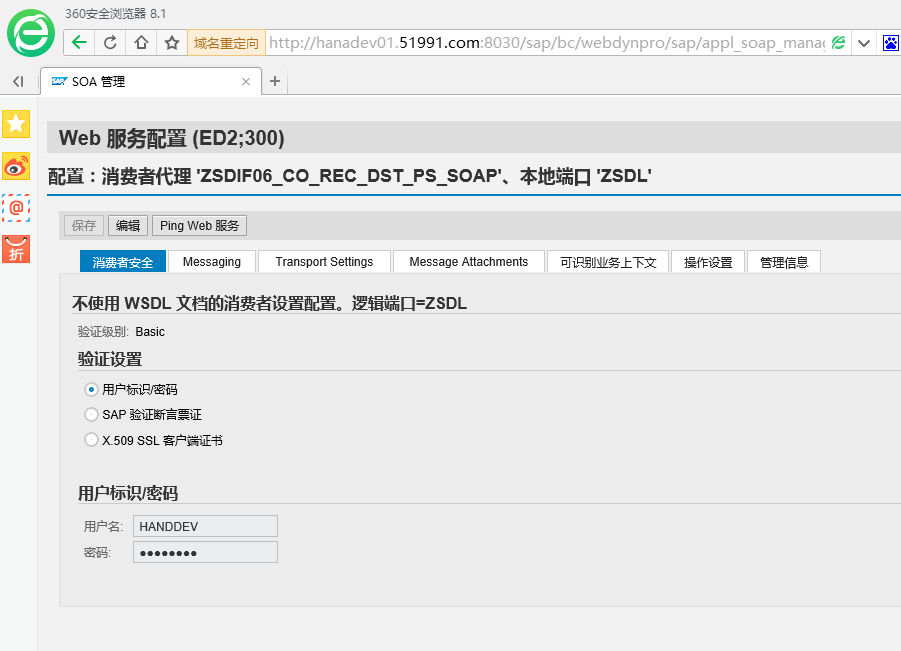


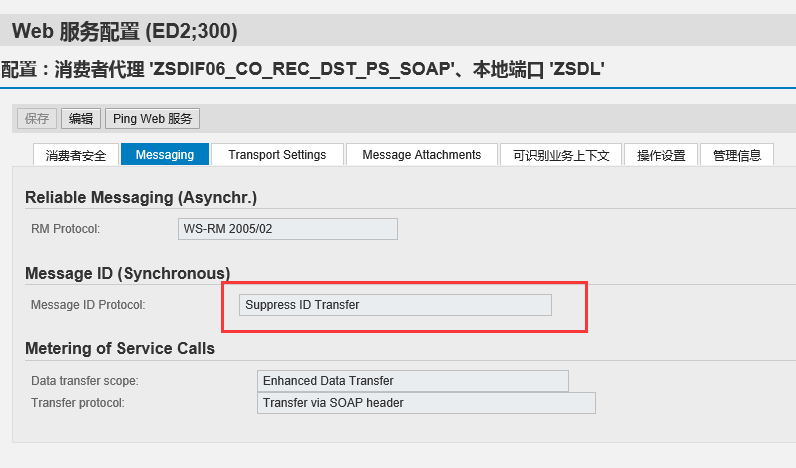


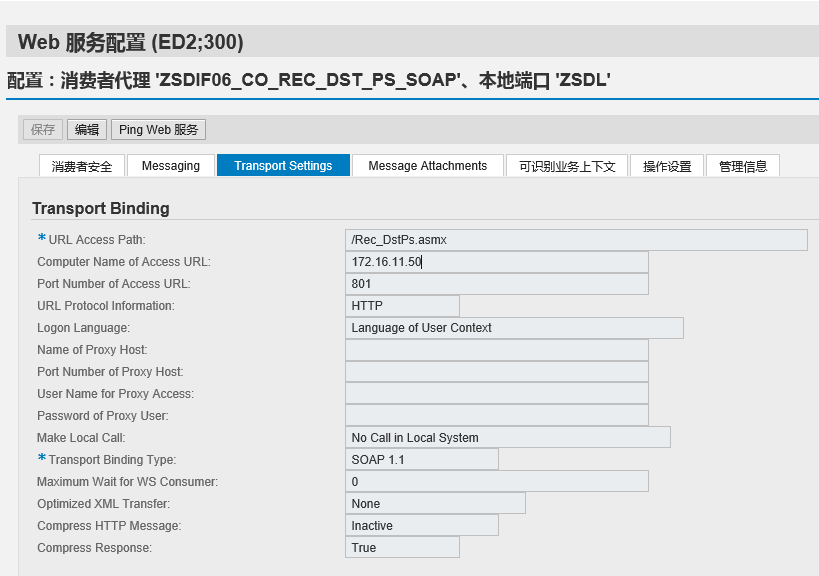




**所有的调用外部webserver的逻辑端口统一用命名一个，并打上勾**



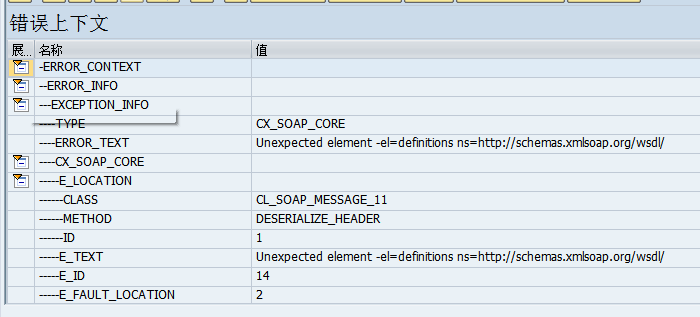


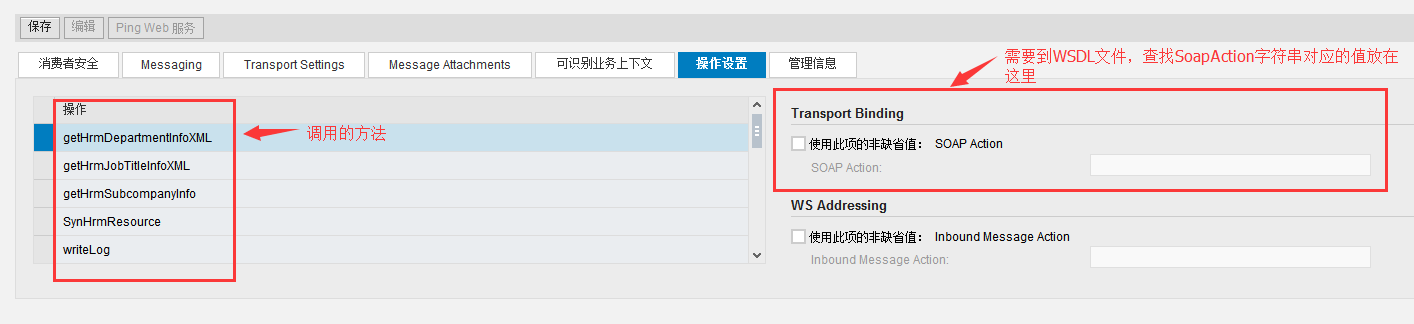


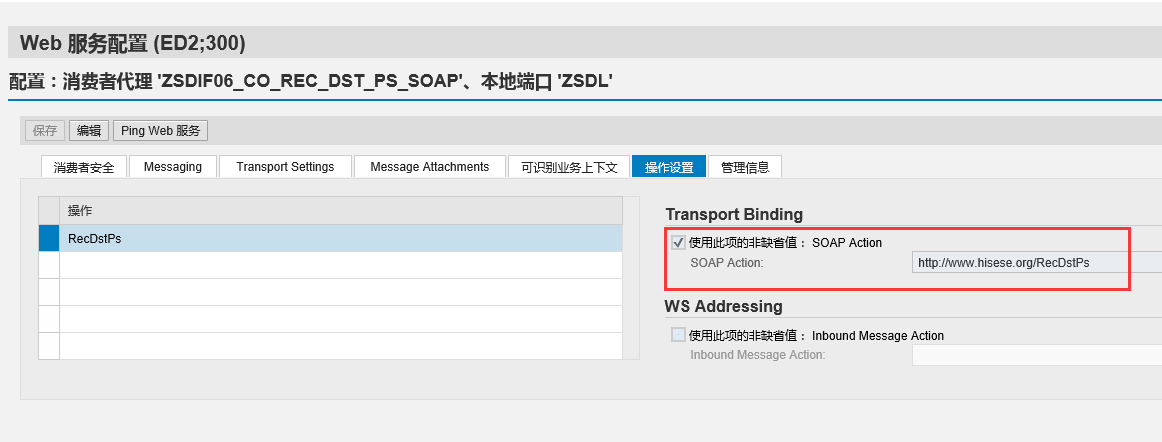
**其中URL通过对应的WSDL文件查找字符串‘address location’，如下所示对应的地址**



**这里的URL不是WSDL的地址，如果错误填写了WSDL的地址，会报如下的错误**

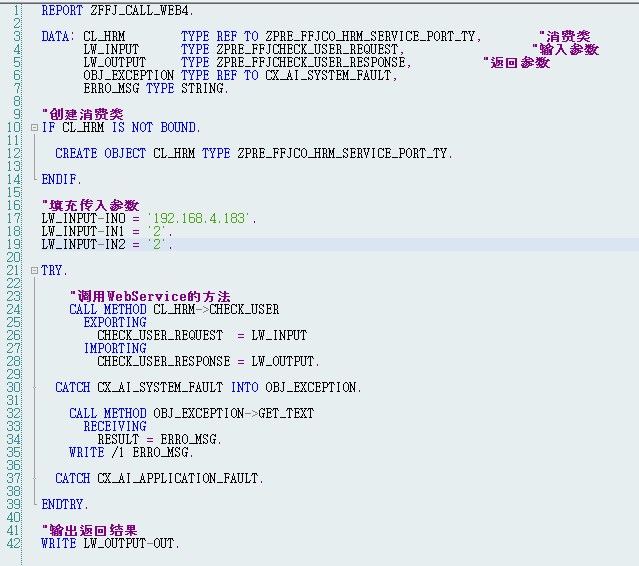








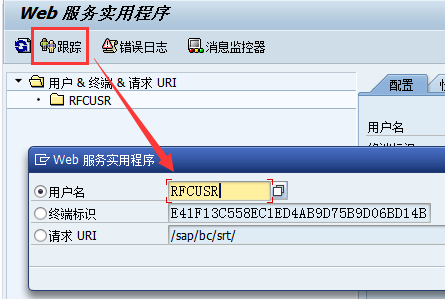
## 编写ABAP程序调用外部WebService

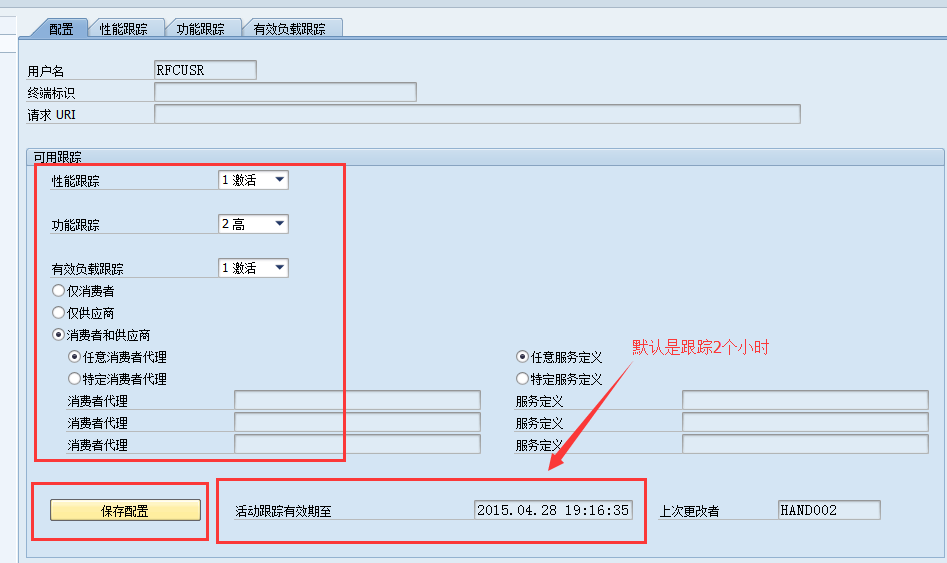


# 【11】调用SAP WebService的日志监控

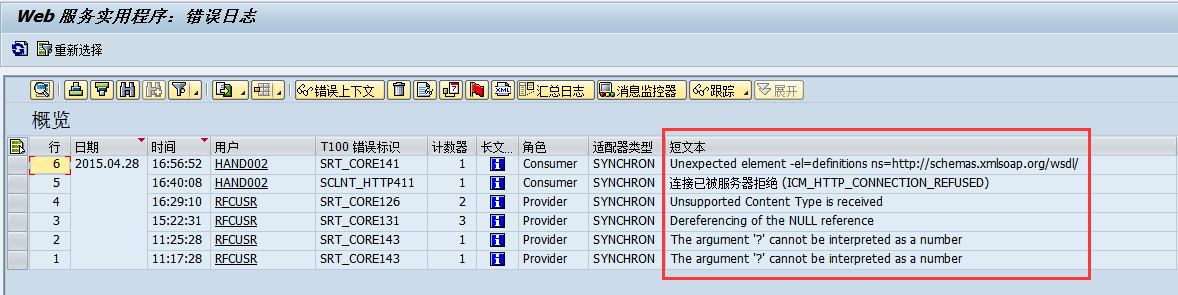
**通过TCODE：SRT\_UTIL**

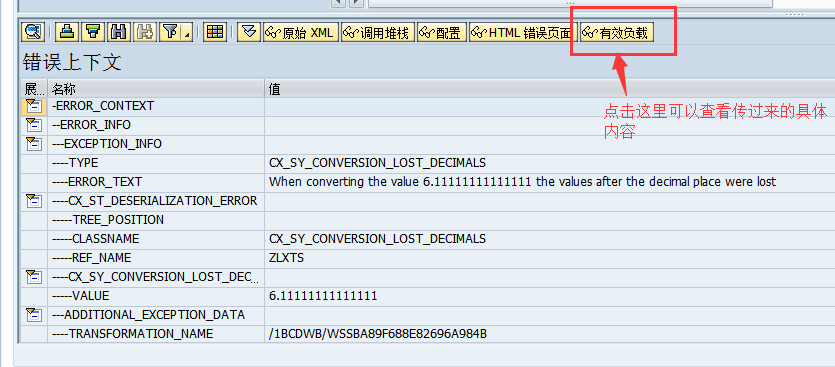
**启用跟踪**





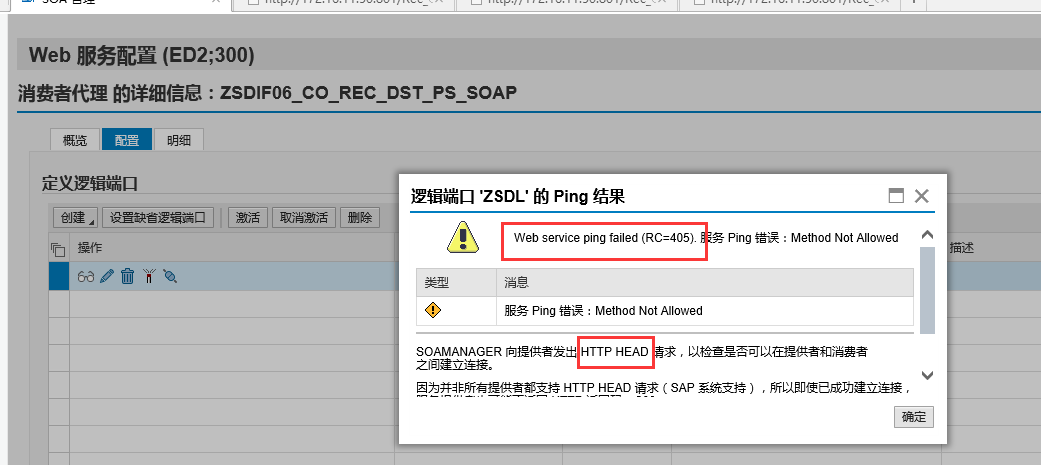
**错误日志**



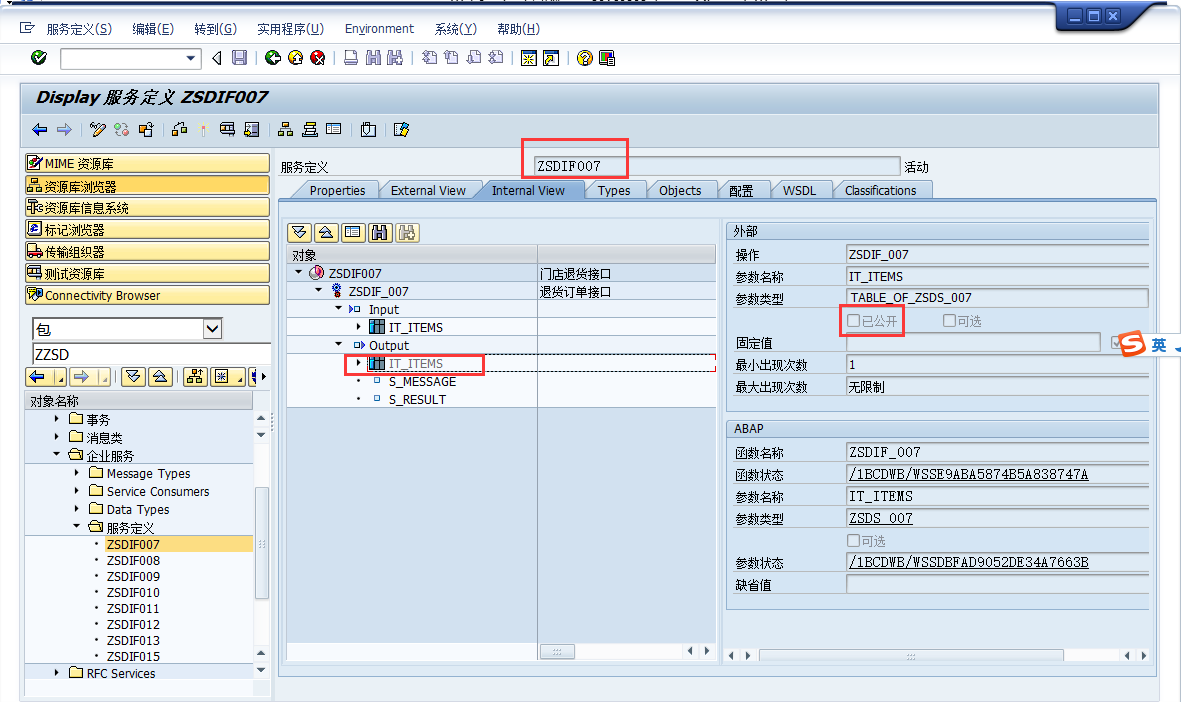


# 【12】常见问题

1. soamanager的配置无法传输，生产机要重新配，前序的一些激活操作也要重新做
2. soamanager上测试405报错，并不能说明不通，自己写程序调用对方的webserver，看结果



1. se37 里面用的是表参数的话，那么表既是输入有是输出，对方调用会发现返回参数里包含他们发送的表，优化点在：



1. 接口里面对方传多个表，当有些表没数据时，直接整个表都不会传过来，这时，这个表作为参数是可选的

