## 异构系统接入OCIP代码以及DEMO说明

1. 功能以及交换步骤说明

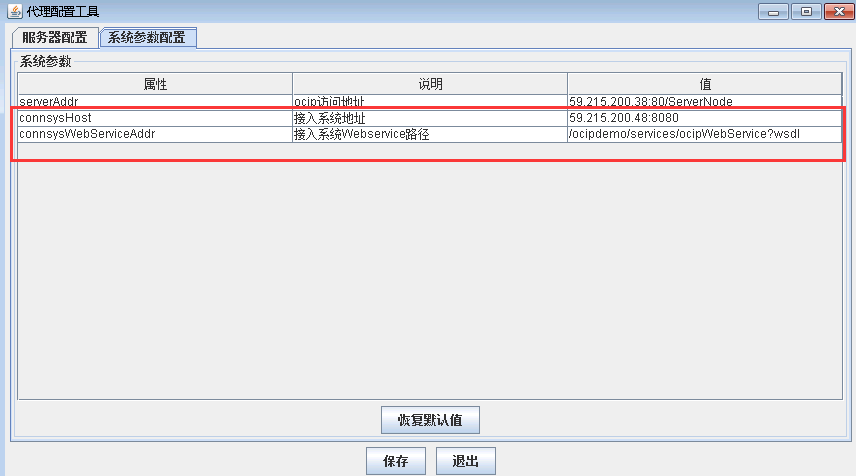
* 功能：针对贵州的办公系统同，能与其它第三方的异构系统相互发送公文，以及公文的签收、回退、撤销动作。
* 交换步骤：

第三方的异构系统要与贵州的办公系统相互发文以及公文的签收、回退、撤销动作主要包含一下3个步骤：

1. 注册OCIP

注册时需要致远方提供代理信息以及ocip.reource-key文件。

接入系统需要提供IP地址和端口，在代理上需要配置信息：



1. 同步组织机构到OCIP

注意：接入系统要给致远发送公文，需要从惠智公司获取组织机构的数据。

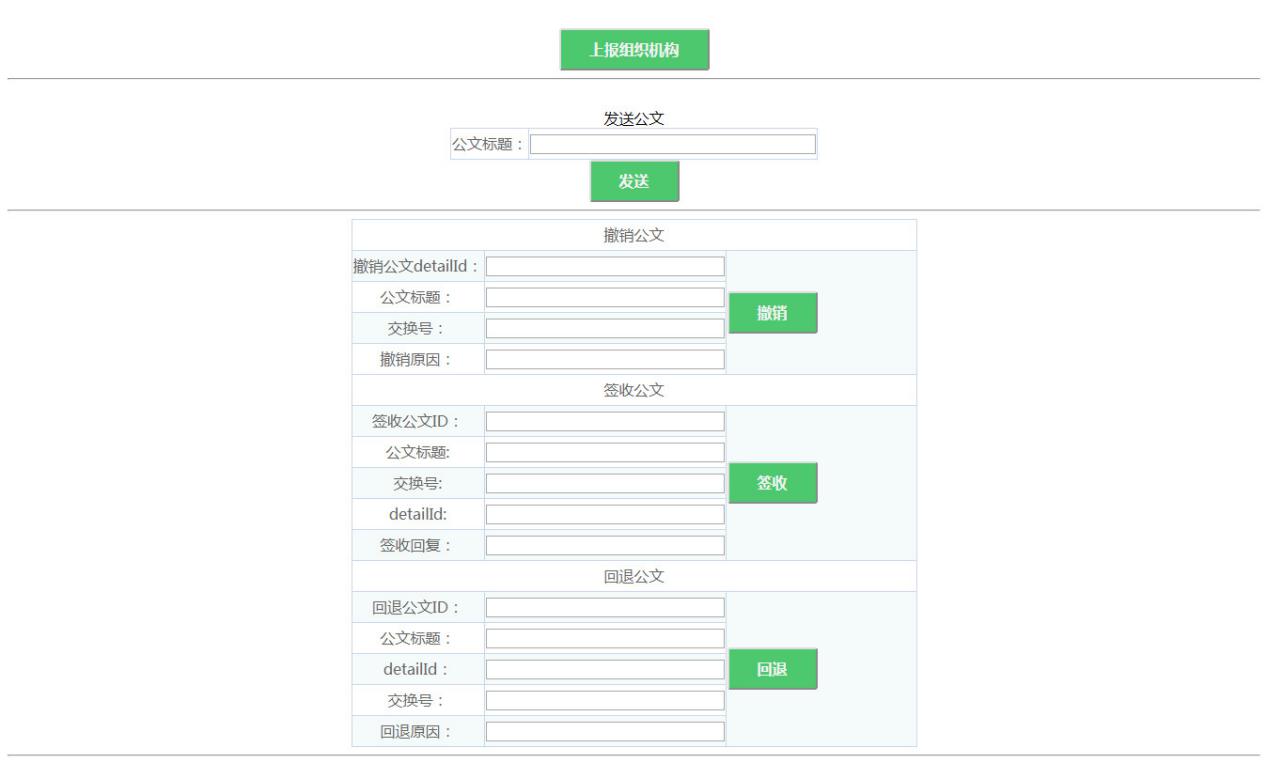
1. 发布WebService
2. 进行交换
3. DEMO使用说明

说明：DEMO功能包含OCIP的注册、组织机构上传、和交换(包括公文交换和公文的回执)功能。

DEMO采用spring实现，在tomcat中启动demo，在启动过程中会自动实现OCIP的注册功能，访问地址：<http://59.215.200.48:8080/ocipdemo/> ，出现页面:

1. 上报组织机构

点击按钮上报组织机构，出现同步成功即可：





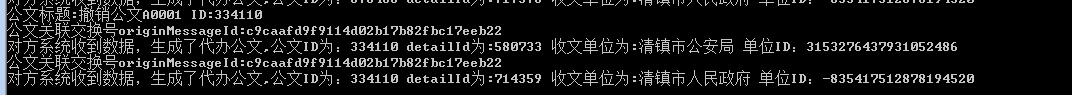
注意：组织机构上报成功后，致远方需要在OCIP上新建异构系统的组织机构视图。

1. 发送公文

在发送公文输入公文的标题，点击发送按钮，发送成功，界面如下：



公文发送成功后，在控制台会打印以下信息：



ID:951854 //公文文号

originMessageId:88e9d413c6a24fd396d8343e2d590515 //OCIP返回的交换号

异构发送公文测试A00001 //公文标题

detailID

注意：交换成功后，OICP会返回detailID、交换号、公文ID，这2个值需要保存下，在异构系统进行公文撤销时会用到。

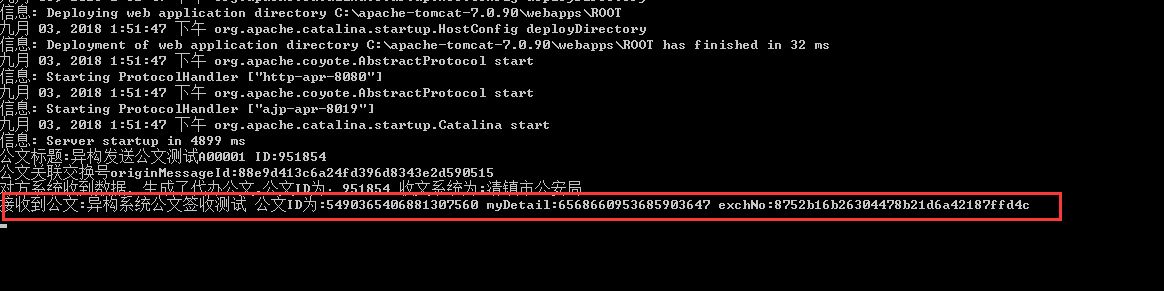
1. 撤销公文

在上一步中，发送公文成功后，OCIP会返回detailID、交换号，公文ID，在撤销公文栏目填写撤销公文ID(发送公文成功后，OCIP返回的detailID)、公文标题、公文ID，交换号(发送公文成功后，OCIP会返回的交换号)、撤销原因，点击撤销按钮，撤销成功如下：



1. 签收公文

当致远方系统给异构系统发送公文时，DEMO收到数据生成代办公文时(异构方进行签收公文操作)，在控制台会打印以下信息：



异构系统公文签收测试 //公文标题

5490365406881307560 //公文ID

6568660953685903647 //myDetail

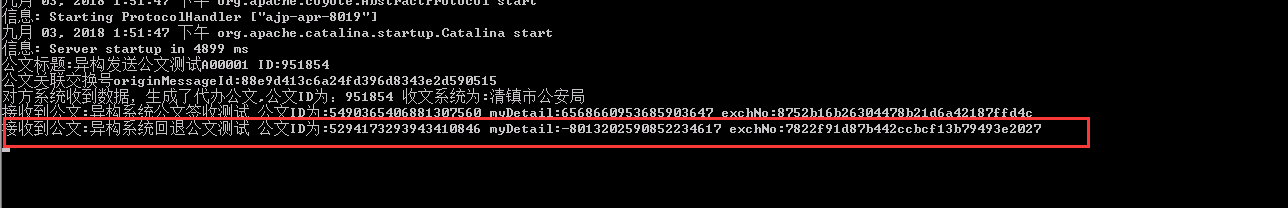
8752b16b26304478b21d6a42187ffd4c //交换号

以上几个参数需要保存下来，签收时需要以上参数，签收成功如下：



1. 回退公文

当致远方系统给异构系统发送公文时，DEMO收到数据生成代办公文时(异构方进行签收公文操作)，在控制台会打印以下信息：



异构系统回退公文测试 //公文标题

5294173293943410846 //公文ID

-8013202590852234617 //myDetail

7822f91d87b442ccbcf13b79493e2027//交换号

以上几个参数需要保存下来，回退时要以上参数，回退成功如下：



1. 代码说明

DEMO是使用spring来实现，使用到了axis2 WebService，接入系统在使用时，注意DEMO中jar包是否冲突,DMEO中spring的相关配置均在applicationContext.xml文件中。

ocip-common.jar

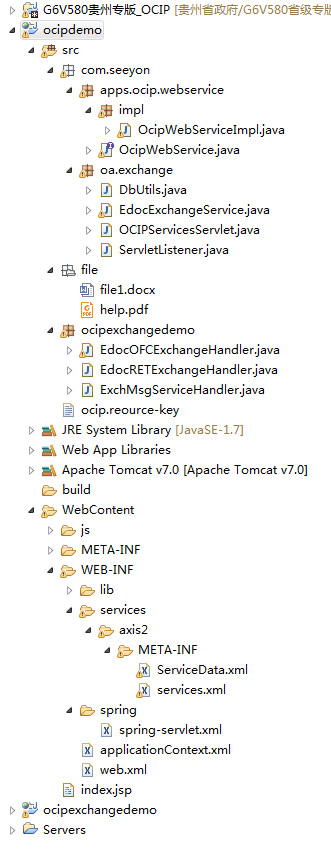
ocip-common-mime.jar

ocip-exchange-axis2.jar

ocip-sdk.jar

是致远方提供的，需要导入，以及axis2的相关jar包。

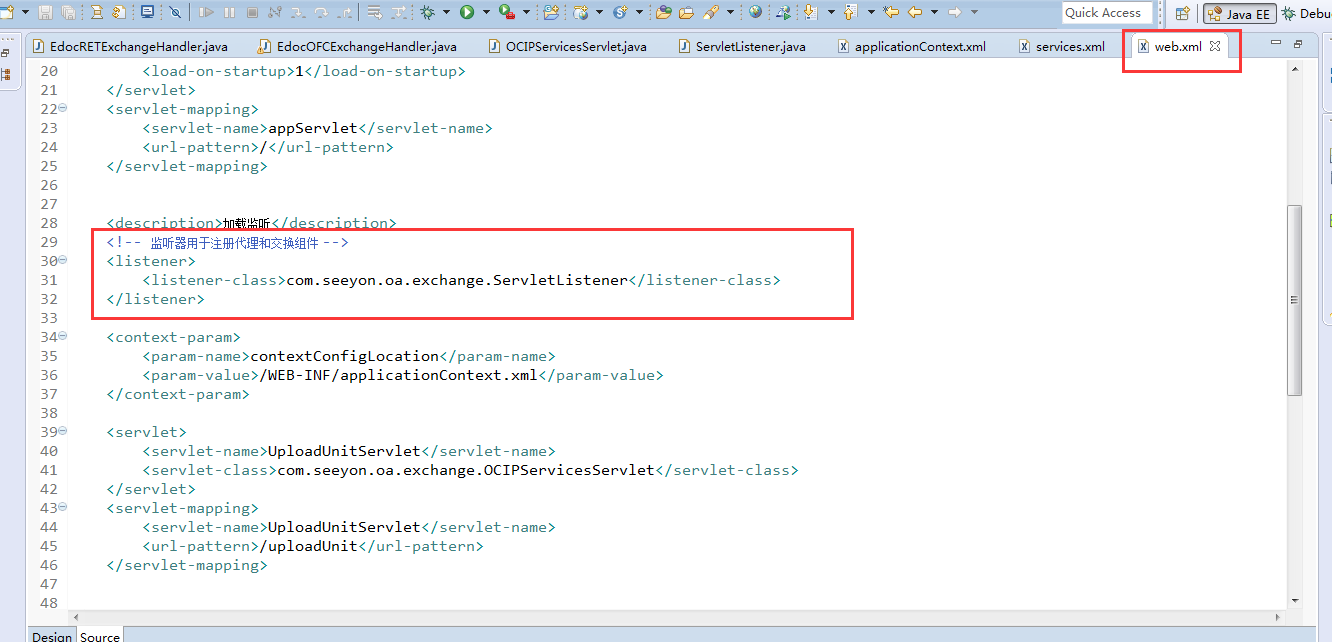
DEMO结构目录如下：



1. 注册OCIP代码实现说明

注册OCIP参考文件ServletListener.java。

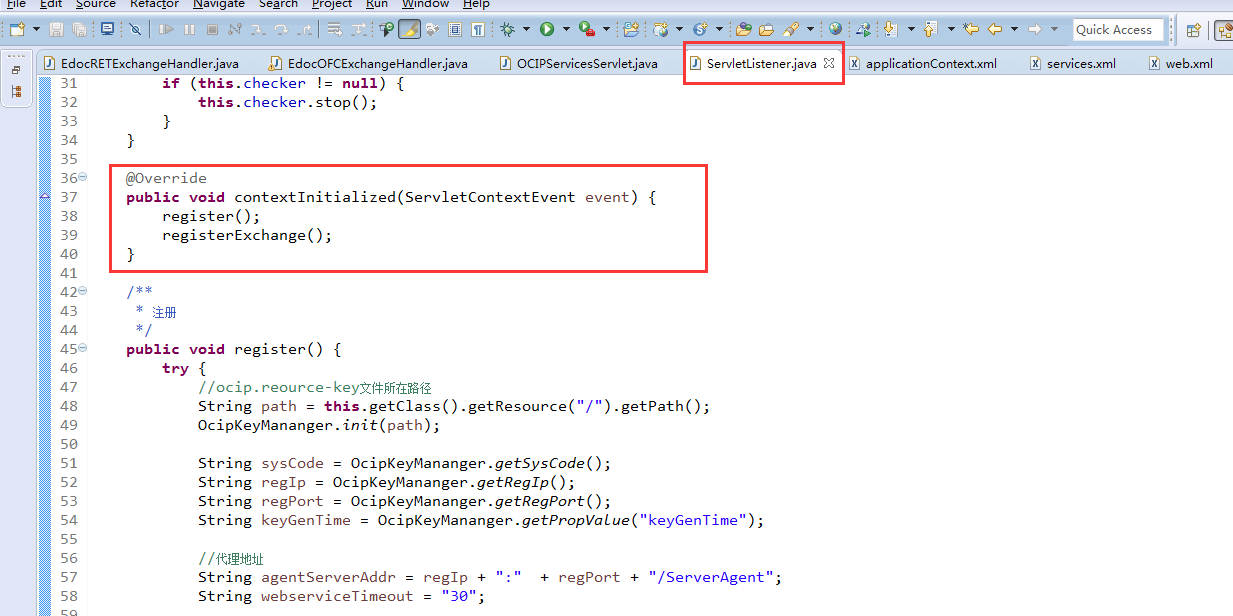
DEMO中注册是使用监听器实现的（第三方系统可以自行实现，只要保证注册的逻辑在服务启动时完成即可），在web.xml中配置如下：



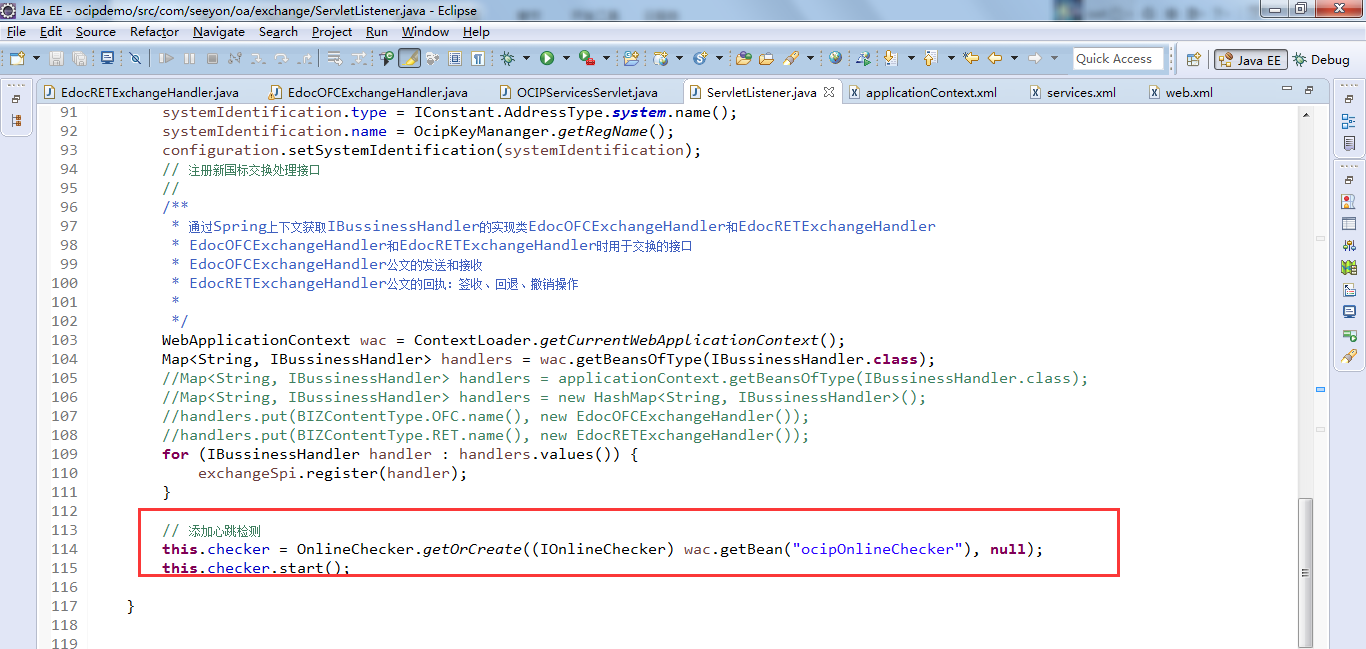
注册包含2个部分：

* OCIP的注册
* 交换组件才组成

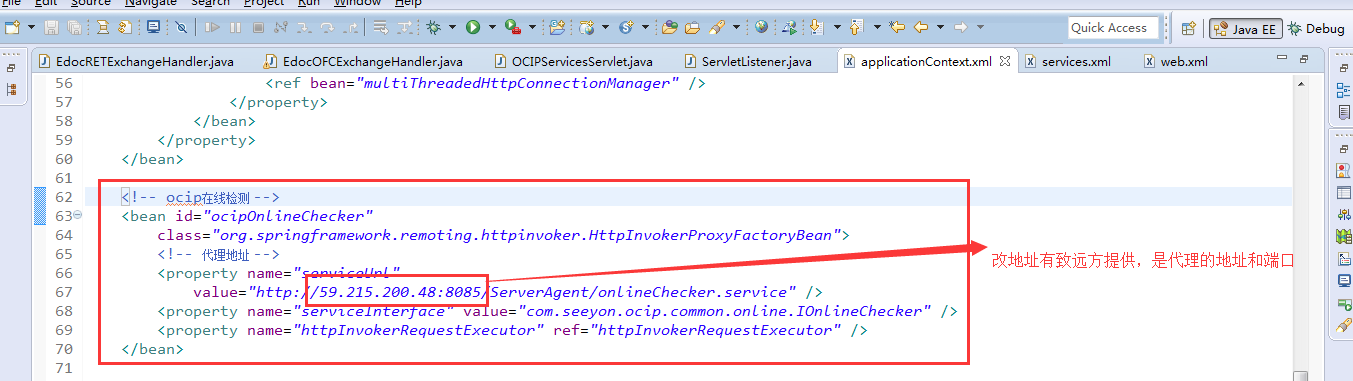
实现参考文件：ServletListener.java



说明：在注册交换组件时，要添加心跳检测，参考ServletListener的registerExchange方法：



ocipOnlineChecker的实现在文件applicationContext.xml中：



注意：

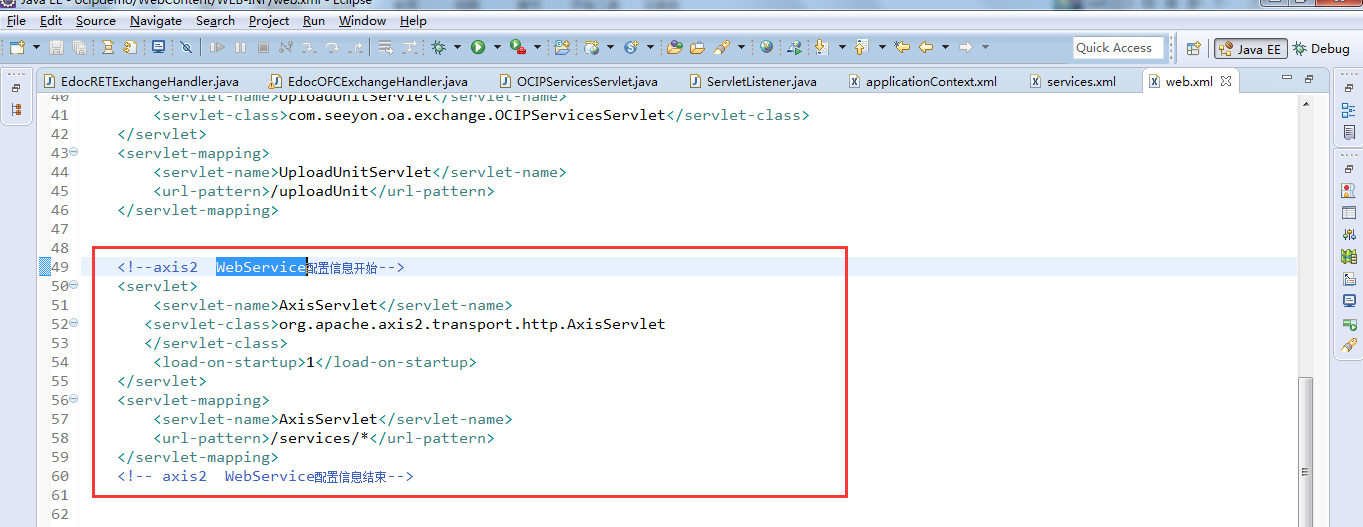
*<http://59.215.200.48:8085/ServerAgent/onlineChecker.service>*

*这个值中的59.215.200.48:8085是代理的IP和端口，由致远提供。*

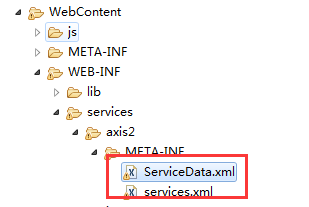
1. 注册WebService服务

由于OCIP与接入系统采用WebService进行通信，因此接入系统需要发布WebService服务。

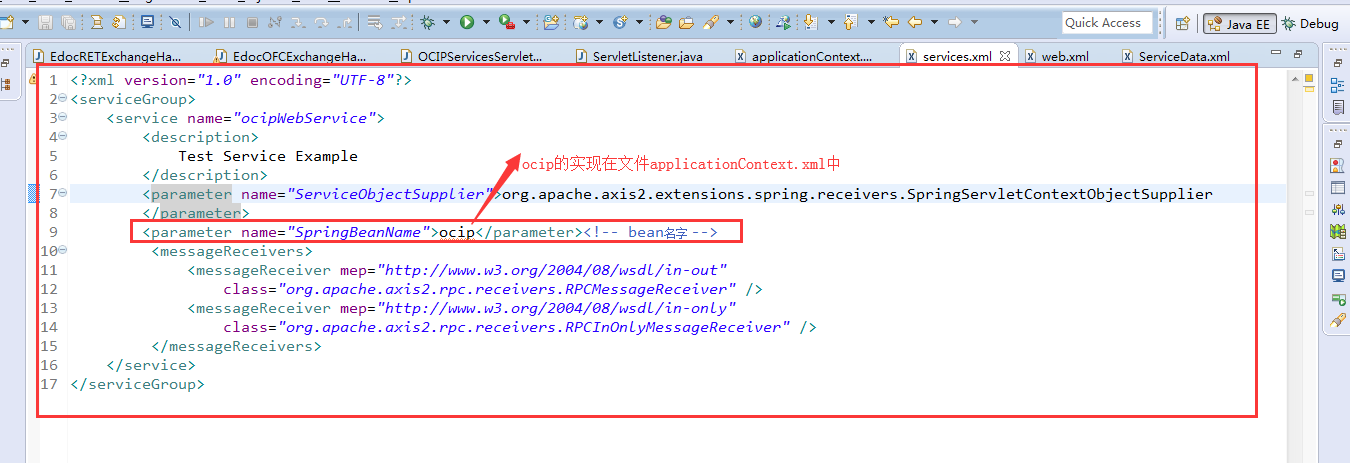
在web.xml中添加配置：



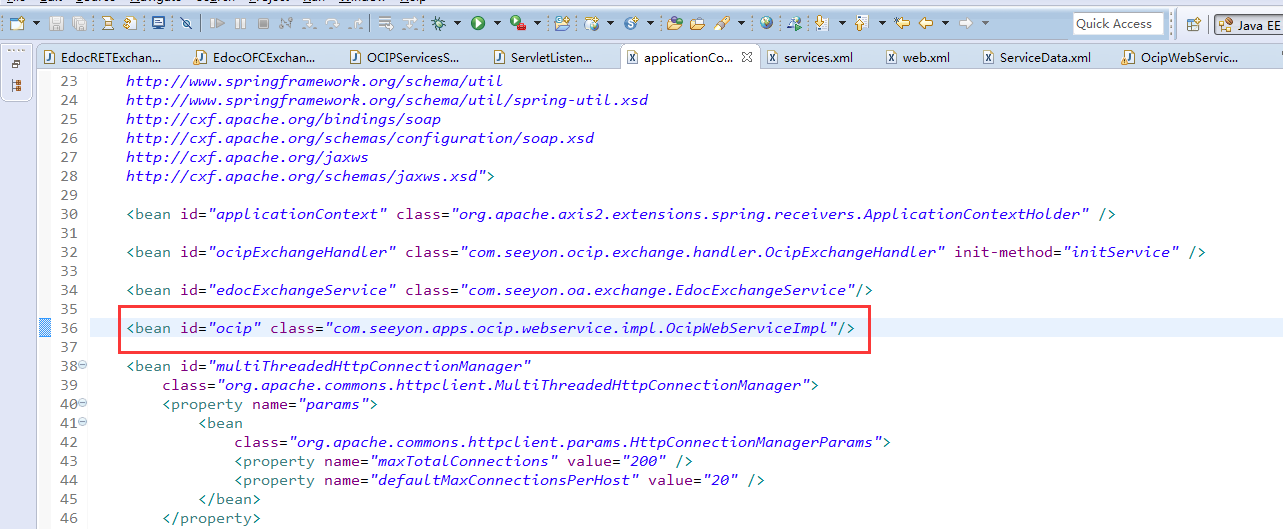
新建services.xml和ServiceData.xml，demo中的WebService可以直接使用，这2个文件需要在如图路径下：



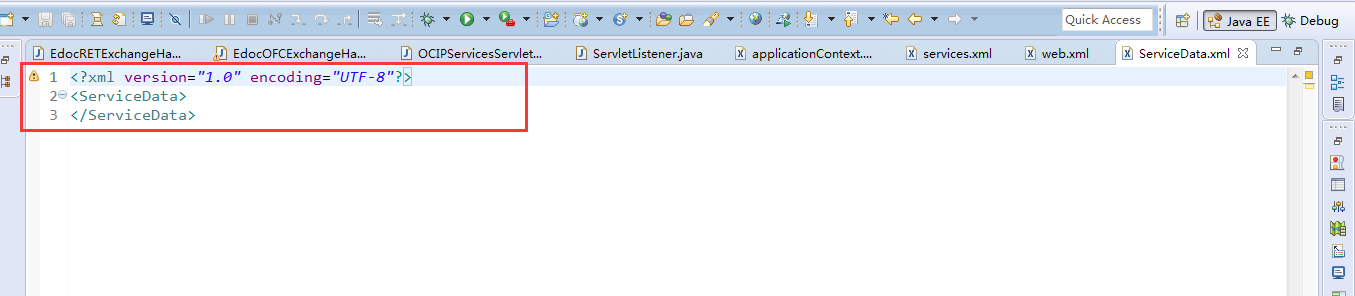
services.xml配置内容：



applicationContext.xml中实例ocip(ocip的实现逻辑参考文件OcipWebServiceImpl，这个类可以直接使用，不必自己实现)的实现如下：



ServiceData.xml，改文件格式如下：



Webservice发布成功与否，可以访问：

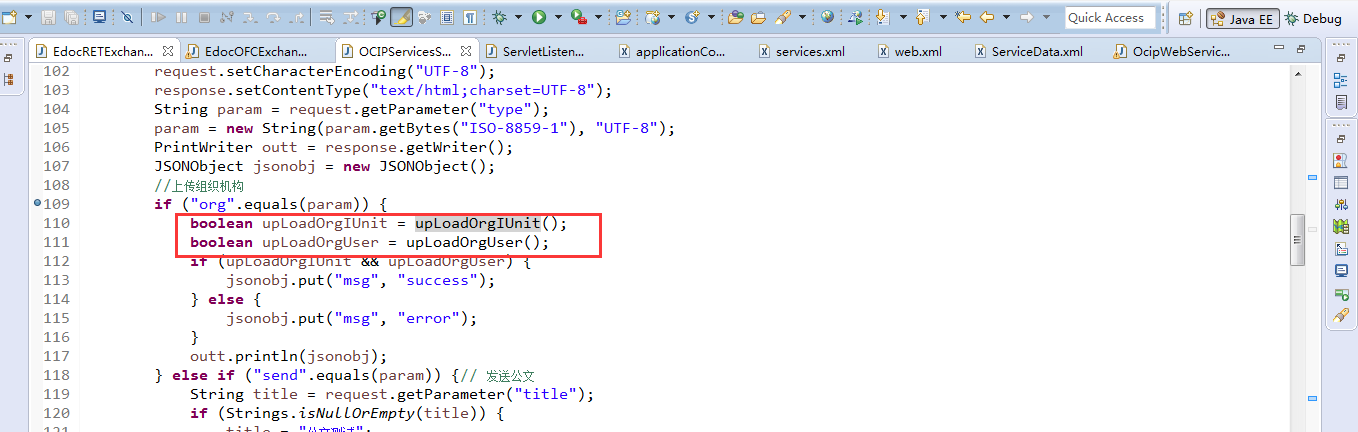
<http://59.215.200.48:8080/ocipdemo/services/ocipWebService?wsdl>



表示发布成功。

1. 组织机构同步

参考文件：OCIPServicesServlet.java的uploadOrgIUnit()和uploadOrgUser()方法：



upLoadOrgIUnit()//上报单位，注意上报单位时必须要将根节点上报

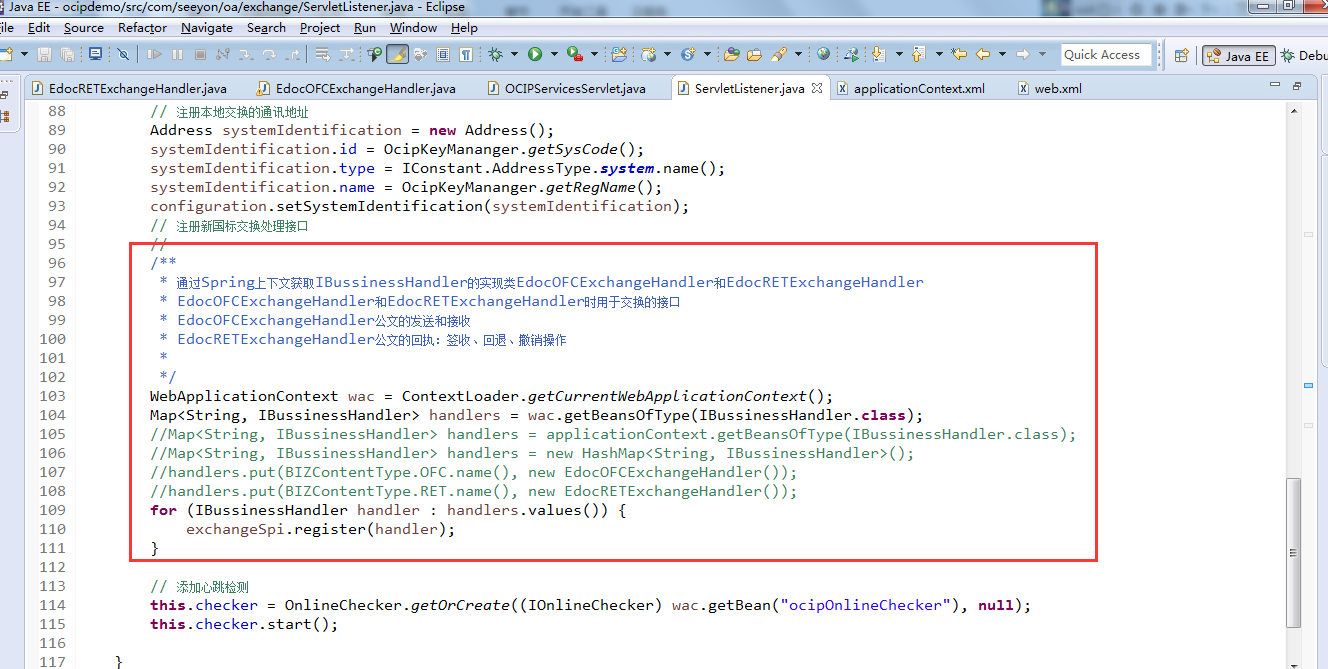
upLoadOrgUser()//上报人员

1. 公文交换以及回执操作

公文的交换和回执需要实现接口IBussinessHandler，DEMO中的实现为：EdocOFCExchangeHandler：公文的发送和接收

EdocRETExchangeHandler：公文的签收、回退、撤销操作。

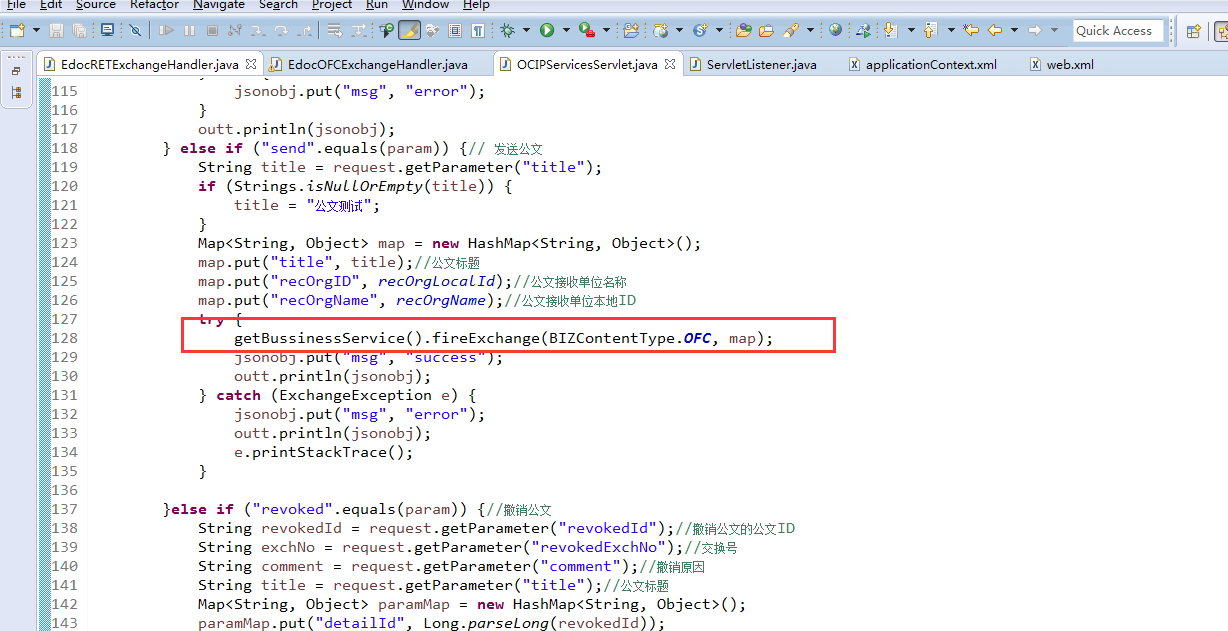
这2个实现类在注册交换组件时需要进行注册：参考ServletListener的方法registerExchange：



实现接口后，在进行公文交换或者回执操作时调用接口(只是传递的参数不一样)：

getBussinessService().fireExchange(BIZContentType.***OFC***, map);

调用接口参考文件:OCIPServicesServlet



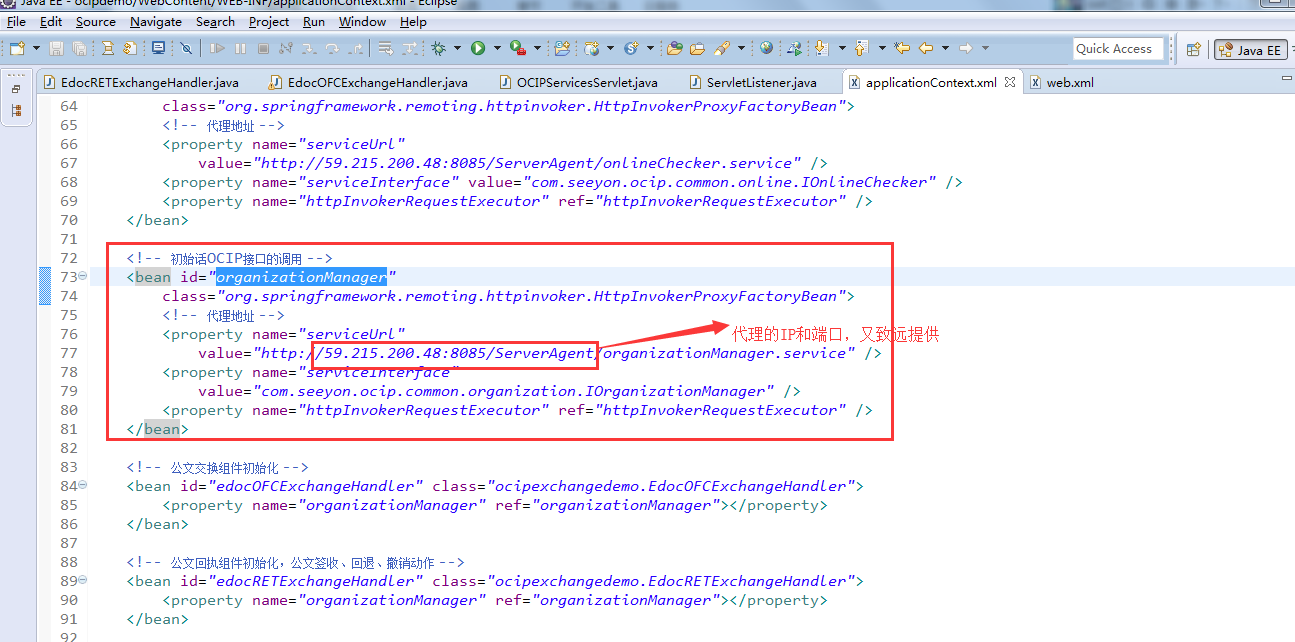
EdocOFCExchangeHandler说明：

该文件实现的公文的发送和接收公文，具体实现参考DEMO：

|  |  |
| --- | --- |
| 方法 | 功能 |
| exchangeReceive | 接收致远方发送的公文 |
| exchangeSend | 发送公文到致远方 |

注意：该实现类中有参数organizationManager，这个参数提供的方法可以将从惠智获取到的人员或者单位转换为OCIP平台对应的对象。

该类的实现在文件applicationContext.xml:



EdocRETExchangeHandler说明：

该文件实现的公文的签收、回退、撤销操作，具体实现参考DEMO：

|  |  |
| --- | --- |
| 方法 | 功能 |
| exchangeReceive | 接收致远方签收、回退、撤销操作 |
| exchangeSend | 异构系统签收、回退、撤销操作发送数据到OCIP |

EdocRETExchangeHandler中的参数说明跟EdocOFCExchangeHandler中的一样。

代理访问地址：

http://59.215.200.38:8085/ServerAgent/?wsdl