Webservice 第二天 cxf+spring

课程安排：

Jaxws开发：

查询三天的天气，返回一个自定义pojo对象

常用注解学习(掌握)

Cxf：

Cxf是一个webservice框架.

Cxf入门程序

Cxf和spring整合（掌握）

发布soap协议的webservice

Cxf和spring整合发布rest webservice（不使用soap协议，使用http，掌握）

综合案例：（学会写）

实现便民查询网站

功能：查询手机号归属地

# 复习

什么是webservice？

Webservice即web服务，是一种跨平台的远程调用技术，可以采用soap协议、http协议进行数据传输。

什么是soap？

Soap是简单对象访问协议，是基于http的应用层协议，协议格式是xml，soap不是webservice专有协议，比如soap还应用于tr069（终端管理协议，基于soap的应用层协议），w3c标准。

什么是wsdl？

Wsdl即网络描述语言，是webservice使用说明书，是xml格式，w3c标准。

阅读时从下往上读：service、binding、portType。。

Webservice的使用场合？

适用场合：

1. 如果开发一个公开接口，面向大众，不考虑客户端类型且不考虑传输性能，采用标准soap协议发布webservice，比如一个产品推广接口。
2. 如果两个公司之间作系统接口，在不考虑接口性能下，可以采用标准soap协议发布webservice接口，公司之间不用再自定义接口协议了。
3. 如果作为客户端需要调用对方已经发布的soap协议的webservice，客户端必须向服务端发soap协议。

不适用场合：

1. 由于webserivce传输soap性能不高，如果接口性能要求较高，不建议使用webservice。
2. 一个公司的相同编程语言开发应用之间传输数据，采用编程语言提供远程调用技术，比如java的RMI。Rmi支持面向对象，速度比webservice快。

Socket使用场合：

Socket的数据传输性能高，采用tcp和udp协议。

使用socket传输数据，必须提前定义接口数据格式，比如xml、json，接口格式 定义好，就可以按照定义的接口格式去流中解析数据。

1. 考虑接口性能，对接口协议格式没有具体特殊要求，采用socket接口通信。比如：一个公司中系统之间的接口。
2. 考虑接口性能，如果跨公司开发socket接口，一般由服务端先定义一个接口格式文档，客户端（调用方）按照接口文档开发socket客户端。

使用socket开发服务端注意：

1. 服务端具有可持续服务的能力。
2. 服务端高并发处理能力。

企业开发中常用webservice传输xml或json数据：

不管webservice是否使用soap都 可以传输xml或json数据，xml或json具有跨平台的特性，公司都认可接口格式，为了方便开发采用webservice传输xml或json数据。

开发注意事项：

服务端接口方法接收一个串（xml或json串）

服务端接口方法返回一个串（xml或json串）

服务端和客户端严格按照定义的接口格式去解析这个串（xml或json串）。

# Jaxws使用自定义pojo发布服务

## 需求

查询三天的天气信息，天气信息包括（天气概况，日期、温度。。）

## 目标

测试jaxws是否支持自定义pojo发布服务。

## 开发

服务端

第一步：自定义pojo包括（天气概况，日期、温度。。）

第二步：开发SEI接口及实现类

第三步：发布服务

客户端：

采用service类标准方法调用webservice

第一步：根据wsdl说明书生成客户端调用代码

第二步：将生成代码拷贝到工程中，编写客户端类。

参考：



## 小结

Jaxws支持自定义的pojo类型

对pojo中的有些属性作一个中间转换，比如java.util.Date会转成javax.xml.datatype.XMLGregorianCalendar

# Jaxws常用注解

如果不使用注解：

默认namespace是服务类包名倒序。

默认portType是服务类的类名

。。。。。

注解的作用：

Jaxws提供的注解可以对webservice的接口规范化。

## @WebService注解

@WebService-定义服务，在public class上边

targetNamespace：指定命名空间

name：portType的名称

portName：port的名称

serviceName：服务名称

endpointInterface：SEI接口地址，如果一个服务类实现了多个接口，只需要发布一个接口的方法，可通过此注解指定要发布服务的接口。

## @WebMethod-定义方法，在公开方法上边

operationName：方法名

exclude：设置为true表示此方法不是webservice方法，反之则表示webservice方法

## @WebResult-定义返回值，在方法返回值前边

name：返回结果值的名称

## @WebParam-定义参数，在方法参数前边

name：指定参数的名称

## 注意

如果修改webservice的接口内容，比如namespace、portType，必须要重新生成客户端调用代码。

# Cxf入门

## 什么是cxf

Cxf，apache下的webservice的开源框架。

Apache CXF = Celtix + Xfire，开始叫 Apache CeltiXfire，后来更名为 Apache CXF 了，以下简称为 CXF。Apache CXF 是一个开源的 web Services 框架，CXF 帮助您构建和开发 web Services ，它支持多种协议，比如：SOAP1.1,1,2、[XML](http://baike.baidu.com/view/63.htm)/[HTTP](http://baike.baidu.com/view/9472.htm)、[REST](http://baike.baidu.com/view/1077487.htm)ful HTTP 或者 [CORBA](http://baike.baidu.com/view/153815.htm)。

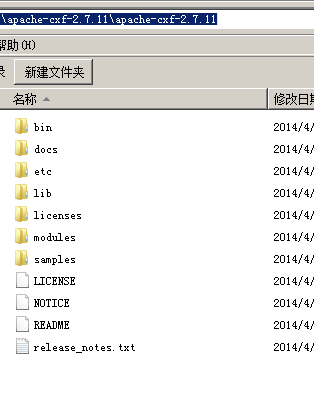
CORBA（Common Object Request Broker Architecture公共对象请求代理体系结构,早期语言使用的WS。C,c++,C#）

Cxf是基于SOA总线结构，依靠spring完成模块的集成，实现SOA方式。

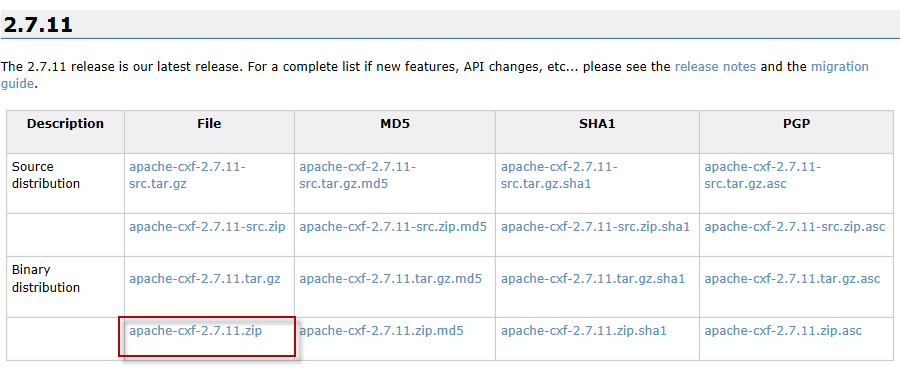
灵活的部署：可以运行有Tomcat,Jboss,Jetty(内置),weblogic上面。

## 环境配置

下载cxf2.7.11版本。



从官网下载2.7.11



环境变量配置：

JAVA\_HOME,

CXF\_HOME=cxf的目录

Path = %JAVA\_HOME%\bin;%CXF\_HOME%\bin;

CLASSPATH=.;%CXF\_HOME%\lib\cxf-manifest.jar

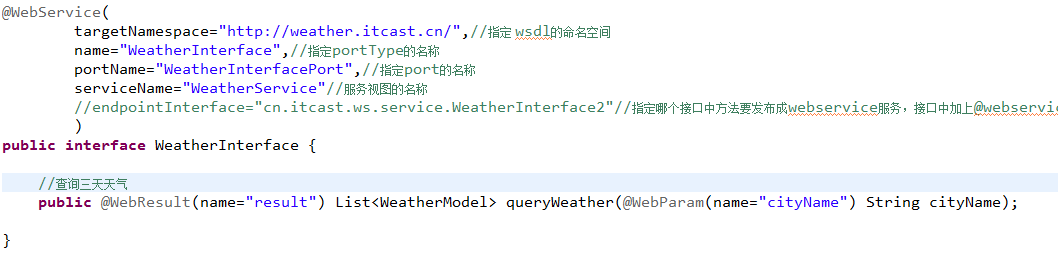
## Cxf第一个程序

服务端

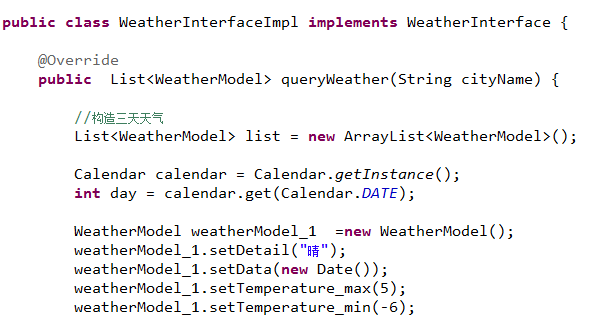
开发SEI及SEI实现类

Cxf开发SEI需要将@webservice注解加在接口中.

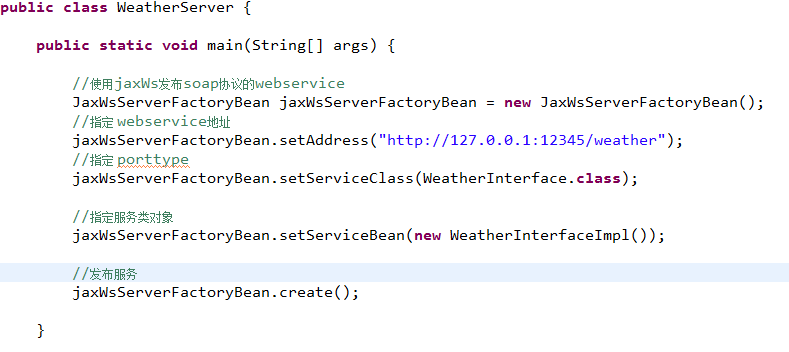
SEI接口：



SEI实现类：



发布服务代码：



客户端

客户端开发技术不受webservice服务端限制，服务端使用cxf，客户端采用多种方法，比如Service类标准方法。

客户端工程中加jar包

如果使用cxf提供客户端类或方法，需要加入cxf的jar包。

Wsdl2java工具生成客户端调用代码

使用cxf的wsdl2java工具生成。

生成源文件：





小结

Cxf编程方式实现webservice：

JaxWsServerFactoryBean：发布webservice服务

JaxWsProxyFactoryBean：调用webservice服务端

Cxf和spring整合后，可以将上边的编程方式修改为spring配置方式实现。

内部 使用JaxWsServerFactoryBean和JaxWsProxyFactoryBean

Cxf发布soap1.2

使用cxf发布soap1.2不需要下载jaxws-ri扩展包。

在服务接口的上面添加

@BindingType(javax.xml.ws.soap.SOAPBinding.SOAP12HTTP\_BINDING)

如下：

@WebService

@BindingType(javax.xml.ws.soap.SOAPBinding.SOAP12HTTP\_BINDING)

public interface WeatherServerInterface

# Cxf和spring整合

## 思路

Cxf框架本身依赖spring，从官方下载cxf包中有spring的jar包。

上边使用JaxWsServerFactoryBean和JaxWsProxyFactoryBean，改为spring配置方式。

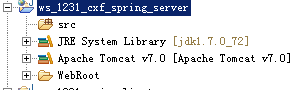
发布服务：使用spring和cxf整合的标签<jaxws:server

客户端调用服务：使用spring和cxf整合的标签<jaxws:client

上边的<jaxws:server和<jaxws:client相当于spring容器的bean，发布服务和客户端调用交给spring容器管理。

## 服务端

创建web工程

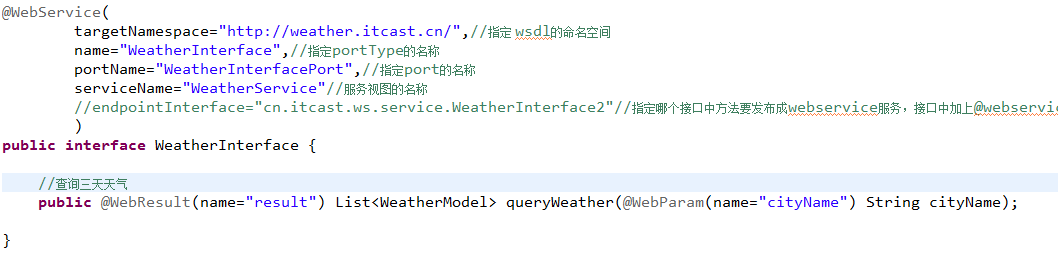


加入cxf的jar包

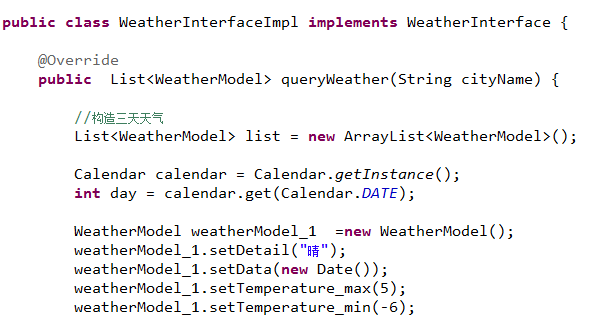
开发SEI和实现类

Cxf开发SEI需要将@webservice注解加在接口中.

SEI接口：

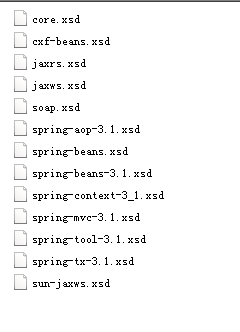


SEI实现类：



编写applicationContext.xml

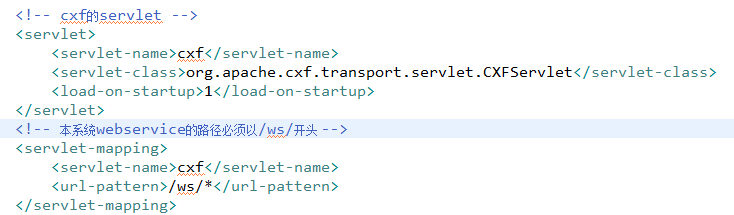
文件中约束包括：spring的、cxf和spring整合



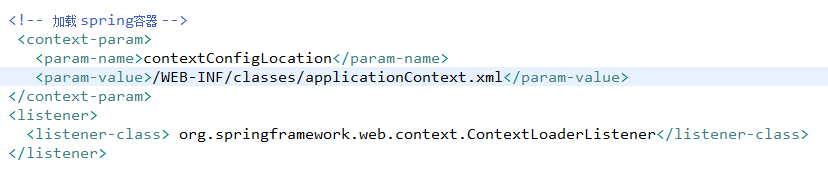


### 在web.xml中配置cxf的servlet

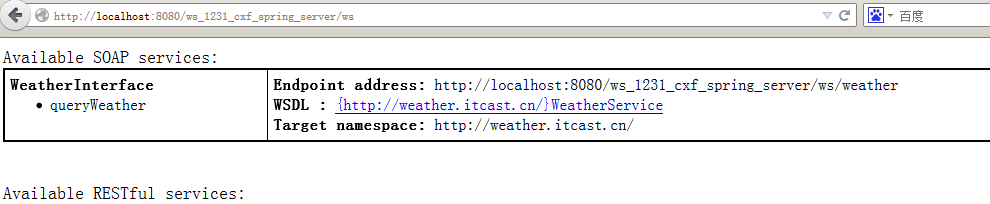
cxf的servlet用于解析cxf发布webservice。



在web.xml中加载spring



启动tomcat



测试注意

因为通过spring和cxf整合将webservice通过tomcat发布，webservice和应用程序共用一个端口都是8080。

测试webservice和应用程序(jsp)是否可以共存（都可以访问）。

正式上线使用80端口。

## 客户端

使用<jaxws:client调用服务端，<jaxws:client需要cxf和spring整合环境。

创建一个java工程

加入cxf的jar包。

使用wsdl2java/wsimport工具生成客户端调用代码

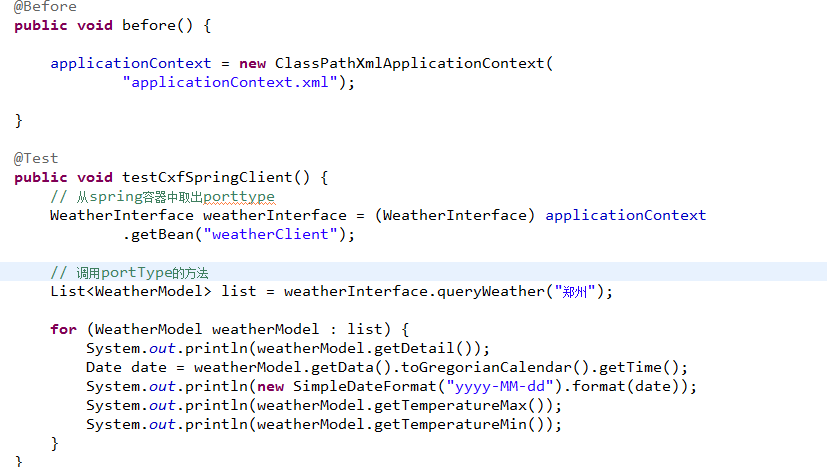


编写applicationContext.xml

使用<jaxws:client配置客户端调用服务端。



测试



# 上午课程回顾

CXF是apache下开源的webservice框架，cxf依赖spring进行模块集成。支持**SOAP1.1,1,2**、[XML](http://baike.baidu.com/view/63.htm)/[HTTP](http://baike.baidu.com/view/9472.htm)、[REST](http://baike.baidu.com/view/1077487.htm)ful HTTP 或者 [CORBA](http://baike.baidu.com/view/153815.htm)。

如何使用 cxf发布soap1.1和soap1.2？

使用jaxws实现**SOAP1.1,1,2。**

方法1：

Cxf编程实现：

* 使用jaxwsServerFactoryBean发布 webservice服务端。

需要设置：

jaxwsServerFactoryBean.setAddress(“webservice地址”)

jaxwsServerFactoryBean.setServiceClass(“porttype类路径”);//由程序员编写的

jaxwsServerFactoryBean.setServiceBean(“porttype类对象”);

jaxwsServerFactoryBean.create();//发布一个服务

* 使用jaxwsProxyFactoryBean实现客户端调用webservice服务

jaxwsServerFactoryBean.setAddress(“webservice的wsdl地址”)

jaxwsServerFactoryBean.setServiceClass(“porttype类路径”);//portType是由wsdl2java工具生成

jaxwsServerFactoryBean.setCreate();//创建porttype对象

方法2：

Cxf和spring整合开发服务端和客户端。

* 使用<jaxws:Server 发布 webservice服务端

在<jaxws:Server设置 Address、serviceClass、serviceBean

* 使用<jaxws:Client 调用webservice服务

在<jaxws:Server设置 Address、serviceClass

# CXF发布rest风格的webservice

## 什么是restFul

REST 是一种软件架构模式，只是一种风格，rest服务采用HTTP 做传输协议，REST 对于HTTP 的利用分为以下两种：资源定位和资源操作。

* 资源定位

更加准确去定位一个互联网资源。使用url定位一个互联网资源。

比如：查询一个学生信息url

资源定位要定位一个学生信息：

不使用rest的url：http://ip:port/queryUser.action?userType=XXX&studentid=001&XXXX

使用 rest的url：<http://ip:port/user/student/001>

使用rest方式表达url更加简单、可以准确表达这个url是一个学生信息查询url

* 资源操作

利用HTTP 的GET、POST、PUT、DELETE 四种操作来表示数据库操作的SELECT、UPDATE、INSERT、DELETE 操作。

比如：

更新学生信息：

设置http请求的方法为post。

定义rest的url：<http://ip:port/user/student/001/张三/13>。。

Rest方式一般要求url中不要有动词。动词采用GET、POST、PUT、DELETE表示。

Rest风格是对http协议很好的诠释。

Rest好处：

定义的url对资源定位更加准确、简单。

Rest缺点：

如果rest的url定义层级比较多，可读性较差。

## Cxf发布rest风格的webservice

目标

使用rest风格表示webservice,cxf发布的webservice地址就是rest风格.

rest风格webservice，不采用soap，直接采用http传输数据。

需求：

Rest Get方法：

查询学生信息，传输参数学生信息id在url中定义。

比如：<http://ip:port/user/student/001> 001就是参数。

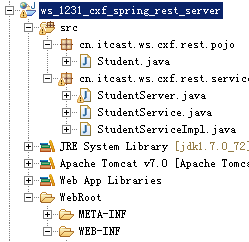
Rest一般返回结果：

xml或json。

实现

使用cxf发布rest的webservice，使用jaxrs（java开发webservice一个规范）

创建web工程

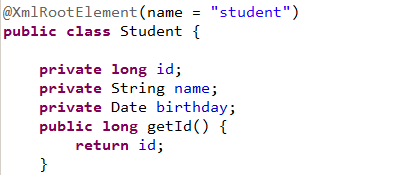


加入cxf的jar包

编写pojo

学生信息的pojo。

要使用@XmlRootElement(name="student")，将pojo转成xml。

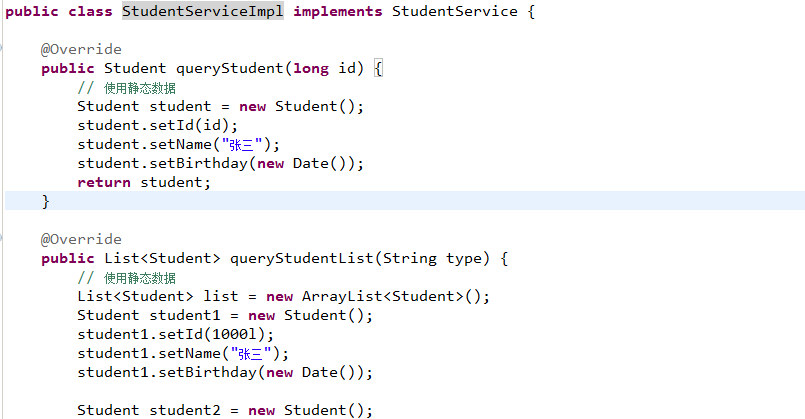


编写SEI



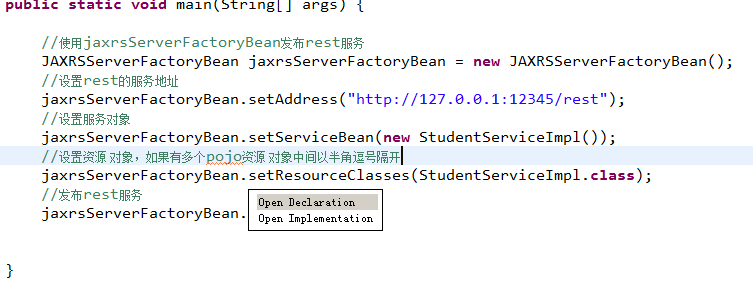
编写SEI实现类

实现类中不用添加rest注解。



发布rest服务

使用JaxRSServerFactoryBean发布rest服务。



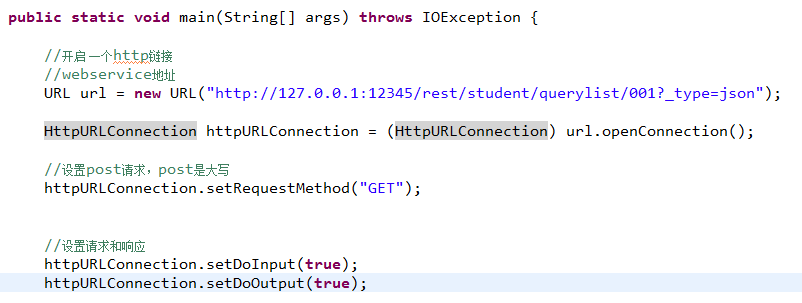
测试 rest服务

<http://127.0.0.1:12345/rest/student/query/001> 查询学生信息

<http://127.0.0.1:12345/rest/student/querylist/001?_type=xml> 查询学生信息列表，返回xml

<http://127.0.0.1:12345/rest/student/querylist/001?_type=json> 查询学生信息列表，返回json

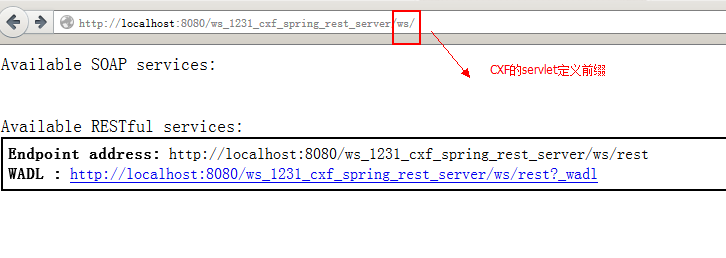
#### Rest服务的客户端httpclient方式



使用spring配置方式发布rest

编写applicationContext.xml





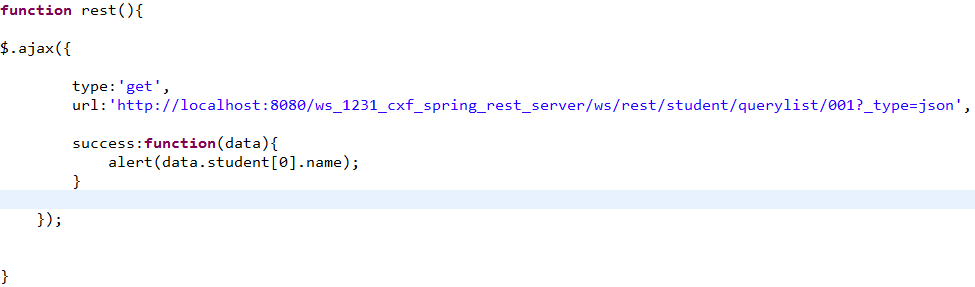
Rest服务的客户端ajax方式

上边发布的http的get服务。

使用http的get请求，调用rest接口(http的get请求接口)。

可以在页面使用ajax请求rest地址

可以使用httpClient方式请求rest地址。



# 综合案例

## 需求

使用cxf和spring开发手机号查询的网站（便民查询网站）。

提供手机号查询页面，用户通过此页面查询手机归属地。

对外发布一个手机号归属地查询的webservice服务。

## 分析

系统架构：

表现层，使用springmvc实现

提供手机号查询页面

业务层，service

手机号归属地查询service

注入dao(查询手机号归属地查询客户端)

功能：

1、供action调用查询手机号归属地

2、对外发布webservice服务(手机号归属地查询)

在spring配置文件中使用<jaxws:server 发布服务

公网手机号归属查询服务

Webservice服务

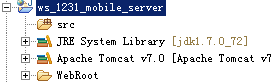
数据层，dao

查询手机号归属地

在spring配置文件使用<jaxws:client（bean） 调用公网服务

## 准备环境

使用web工程：



Jar包：

Spring3.1.4版本(包括springmvc的jar)

Cxf2.7.11版本

将cxf的lib目录 下spring3.0.2删除，使用spring3.1.4替换

## dao

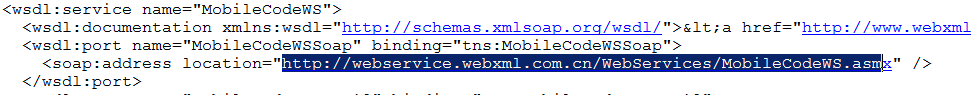
查询手机号归属地，在spring配置文件使用<jaxws:client（bean）调用公网手机号查询webservice服务。

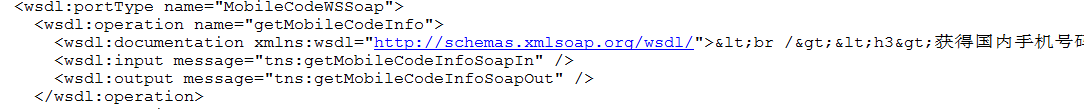
找到手机号归属地查询wsdl

从<http://WebXml.com.cn/>找到手机号归属地查询。

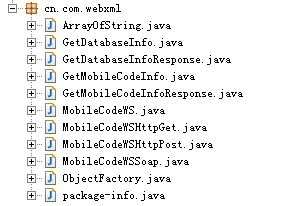
MobileWebService.wsdl

确定调用哪个portType.





使用wsimport/wsdl2java生成客户端的调用代码



在applicationContext.xml配置客户端

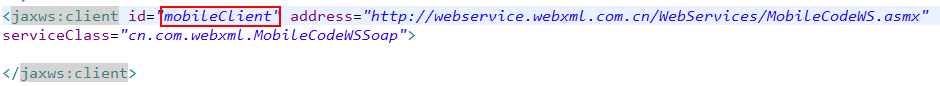
使用<jaxws:client（bean） 调用公网手机号查询。



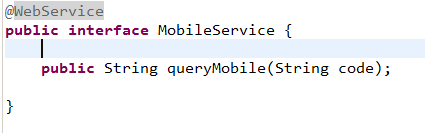
## service

编写SEI及实现类

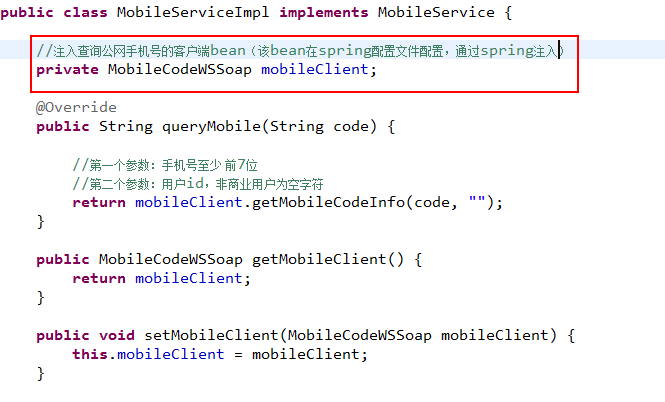
SEI实现类中要注入手机号查询的客户端bean



SEI：



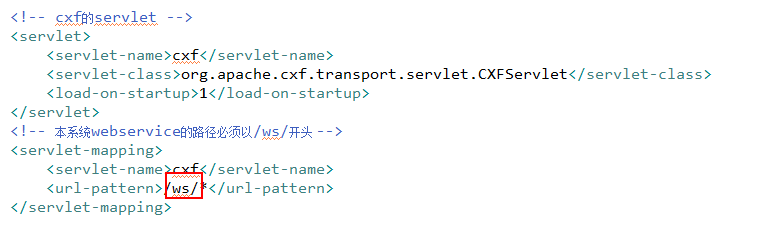
SEI实现类：



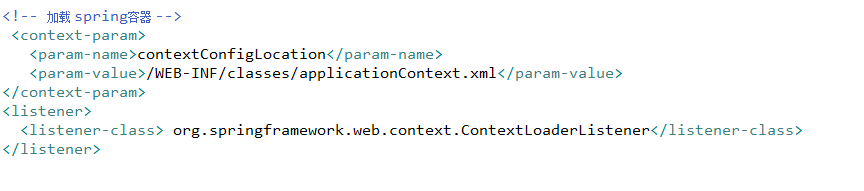
在applicationContext.xml配置发布服务



## 在web.xml中配置cxf的servlet



## 在web.xml中配置加载spring容器



## 表现层

具体配置方法参考springmvc例子工程。

配置springmvc.xml

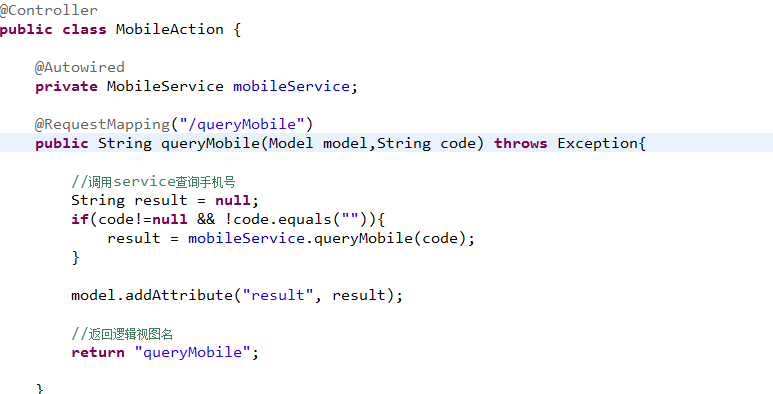
配置处理器适配置器、处理器映射器、视图解析器。



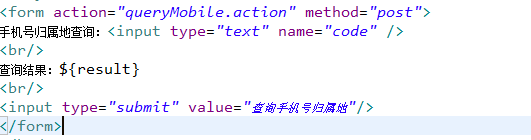
在web.xml配置springmvc的前端控制器



编写MobileAction



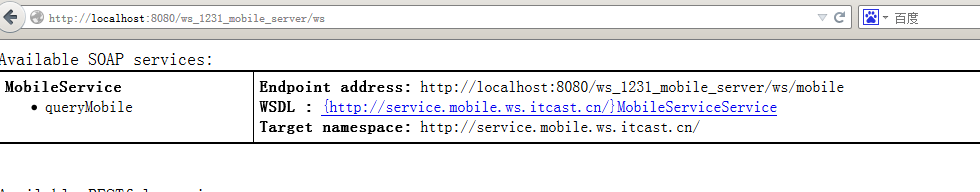
页面



测试：

目标：

1. 手机号查询是否发布为webservice



2、通过页面输入手机号查询归属地