Atitit 动态调用webservice与客户端代理方式调用

方式1： 使用call.invoke 直接调用WSDL，缺点：麻烦，不推荐……特别是JAVA调用.NET的WS时，会有不少的问题需要解决。

方式2： 生成测试STUB……此种方法更快捷。如果用.NET开发WS客户端，就是用的这种方式……

****客户端调用web service方法****首先要使用cxf的wsdl2java命令工具生成服务代理类,如:wsdl2java -p com.charles.client http://localhost:8080/queryUser?wsdl  
具体命令格式可使用wsdl2java /？查看。生成后的代理类名是：服务实现类名+Service

****方法一：使用标准的JAX-WS的API完成客户端调用****

//注意。此处http:http://www.itfad.net/queryUser是服务的命名空间，可查看wsdl文件中的targetNamespace

QName qName = new QName("http://www.itfad.net/queryUser","QueryUserImplService");  
QueryUserImplService ImplService =new QueryUserImplService(new URL("http://localhost:8080/CXFDemo/queryUser?wsdl"),qName);  
IQueryUser service=(IQueryUser)ImplService.getPort(IQueryUser.class);

****方法二：javax.xml.ws.Service****

String endPointAddress = "http://localhost:8080/queryUser";//服务实际地址  
//此处http://charles.com/为命名空间，默认是包名的倒序。IQueryUserService=服务接口名+service  
javax.xml.ws.Service service = javax.xml.ws.Service.create(new QName("http://charles.com/","IQueryUserService"));  
//IQueryUserPort=服务接口名+Port  
service.addPort(new QName("http://charles.com/","IQueryUserPort"),javax.xml.ws.soap.SOAPBinding.SOAP11HTTP\_BINDING, endPointAddress);  
IQueryUser queryService1 =service.getPort(IQueryUser.class);

System.out.println(queryService1.query(user));

这里其实利用wsdl2java工具为我们生成类时已经包含了这种方法。比如我们的服务接口是IQueryUser，那么我们会发现生成的文件下有个IQueryUserService类  
该类正是继承至javax.xml.ws.Service，并封装以上[代码](http://www.xuebuyuan.com/" \o "代码" \t "http://www.xuebuyuan.com/_blank)的功能，调用代码如下：  
IQueryUserService queryUserService=new IQueryUserService();//默认构造函数里使用默认wsdl地址和服务的QName  
IQueryUser queryService1 =queryUserService.getIQueryUserPort();//方法内部调用了super.getPort(IQueryUserPort, IQueryUser.class);  
System.out.println(queryService1.query(user));

****方法三：使用CXF中JaxWsProxyFactoryBean客户端代理工厂调用web服务****

JaxWsProxyFactoryBean soapFactoryBean = newJaxWsProxyFactoryBean();  
soapFactoryBean.setAddress("http://localhost:8080/queryUser");  
soapFactoryBean.setServiceClass(IQueryUser.class);  
Object o = soapFactoryBean.create();  
IQueryUser service = (IQueryUser)o;

****方法四：使用CXF的JaxWsDynamicClientFactory动态代理工厂****

JaxWsDynamicClientFactory dcf = JaxWsDynamicClientFactory.newInstance();  
Client client = dcf.createClient("people.wsdl", classLoader);  
Object person =Thread.currentThread().getContextClassLoader().loadClass("com.acme.Person").newInstance();  
Method m = person.getClass().getMethod("setName", String.class);  
m.invoke(person, "Joe Schmoe");  
client.invoke("addPerson", person);

## Axis动态调用

Service service = **new** Service();

Call call = (Call) service.createCall();

call.setTargetEndpointAddress(**new** java.net.URL(endpoint));

QName q = **new** QName(namespace, "GetCXInfoByVIN");

call.setOperationName(q);// LSJW26H35BS047598

call.addParameter(**new** QName(namespace, "vin"),

org.apache.axis.encoding.XMLType.*XSD\_STRING*,

javax.xml.rpc.ParameterMode.*IN*);

// call.setReturnClass(org.w3c.dom.Element.class);

Object obj = call.invoke(**new** Object[] { "LSJW26H35BS147598" });// LSJW26H35BS147598

参考

webservice 开发总结.docx

作者:: 绰号:老哇的爪子claw of Eagle 偶像破坏者Iconoclast image-smasher

捕鸟王"Bird Catcher 王中之王King of Kings 虔诚者Pious 宗教信仰捍卫者 Defender of the Faith. 卡拉卡拉红斗篷 Caracalla red cloak

简称：： Emir Attilax Akbar 埃米尔 阿提拉克斯 阿克巴

全名：：Emir Attilax Akbar bin Mahmud bin attila bin Solomon bin adam Al Rapanui

埃米尔 阿提拉克斯 阿克巴 本 马哈茂德 本 阿提拉 本 所罗门 本亚当 阿尔 拉帕努伊

常用名：艾提拉（艾龙），   EMAIL:1466519819@qq.com

头衔：uke总部o2o负责人，全球网格化项目创始人，uke宗教与文化融合事务部部长，Uke部落首席大酋长，uke制度与重大会议委员会委员长，uke保安部首席大队长,uke制度检查委员会副会长，奶牛科技cto ，uke波利尼西亚区大区连锁负责人，克尔格伦群岛区连锁负责人，莱恩群岛区连锁负责人，uke汤加王国区域负责人。布维岛和南乔治亚和南桑威奇群岛大区连锁负责人

转载请注明来源：attilax的专栏  http://blog.csdn.net/attilax

--Atiend