**[Java使用MyEclipse构建webService简单案例](http://www.cnblogs.com/lee0oo0/archive/2013/01/25/2876357.html)**

转：http://www.cnblogs.com/jasoncc/archive/2011/12/22/2296052.html

今天主要和大家分享，如何搭建一个Web服务，做Android开发，不可避免会涉及到客户端开发，我们怎么样来实现一个服务端，怎么样来实现一个客户端，并相互传递数据。就算调用别人的服务时，也能知道大概是怎么样实现的。

      WebService一般分为.Net版和Java版，今天我们主要来实现Java版的WebService，.Net版本的还是比较简单的。

     照例，轻松一笑：

     波音777是有史以来第一架完全在电脑虚拟现实中设计制造的飞机，所用的设备完全由IBM公司所提供。试飞前，波音公司的总裁非常热情的邀请IBM的技术主管去参加试飞，可那位主管却说道:“啊，非常荣幸，可惜那天是我妻子的生日，所以...”   
波音公司的总载一听就生气了:“胆小鬼，我还没告诉你试飞的日期呢!”

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211649014549.png)

**什么是WebServices？**

    它是一种构建应用程序的普遍模型,可以在任何支持网络通信的操作系统中实施运行;它是一种新的web应用程序分支，是自包含、自描述、模块化的应用，可以发布、定位、通过web调用。Web Service是一个应用组件,它逻辑性的为其他应用程序提供数据与服务.各应用程序通过网络协议和规定的一些标准数据格式（Http，XML，Soap)来访问Web Service,通过Web Service内部执行得到所需结果.Web Service可以执行从简单的请求到复杂商务处理的任何功能。一旦部署以后，其他Web Service应用程序可以发现并调用它部署的服务。

**关键的技术和规则**

   在构建和使用Web Service时,主要用到以下几个关键的技术和规则:

   1.XML:描述数据的标准方法.

  2.SOAP:表示信息交换的协议.

  3.WSDL:Web服务描述语言.

   4.UDDI:通用描述、发现与集成，它是一种独立于平台的，基于XML语言的用于在互联网上描述商务的协议。

**XML**

   可扩展的标记语言(XML)是Web service平台中表示数据的基本格式。除了易于建立和易于分析外，XML主要的优点在于它既是平台无关的，又是厂商无关的。无关性是比技术优越性更重要的：软件厂商是不会选择一个由竞争对手所发明的技术的。

**SOAP**

  SOAP是web service的标准通信协议，SOAP为simple object access protocoll的缩写，简单对象访问协议. 它是一种标准化的传输消息的XML消息格式。

**WSDL**

   WSDL的全称是web service Description Language,是一种基于XML格式的关于web服务的描述语言。其主要目的在于web service的提供者将自己的web服务的所有相关内容,如所提供的服务的传输方式，服务方法接口，接口参数，服务路径等，生成相应的完全文档，发布给使用者。使用者可以通过这个WSDL文档，创建相应的SOAP请求消息，通过HTTP传递给webservice提供者；web服务在完成服务请求后，将SOAP返回消息传回请求者，服务请求者再根据WSDL文档将SOAP返回消息解析成自己能够理解的内容。

**UDDI**

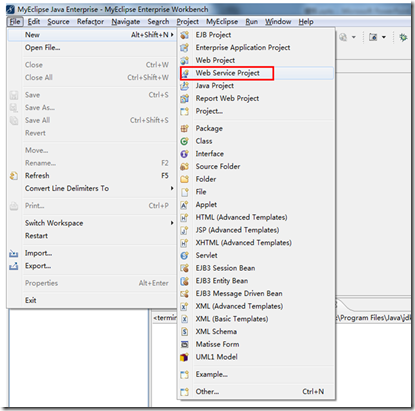
   将web service进行UDDI注册发布,UDDI是一种创建注册表服务的规范,以便大家将自己的web service进行注册发布供使用者查找.然而当服务提供者想将自己的web service向全世界公布，以便外部找到其服务时，那么服务提供者可以将自己的web service注册到相应的UDDI商用注册网站，目前全球有IBM等4家UDDI商用注册网站。因为WSDL文件中已经给定了web service的地址URI，外部可以直接通过WSDL提供的URI进行相应的web service调用。所以UDDI并不是一个必需的web service组件，服务方完全可以不进行UDDI的注册。

**创建并发布自己的WS**

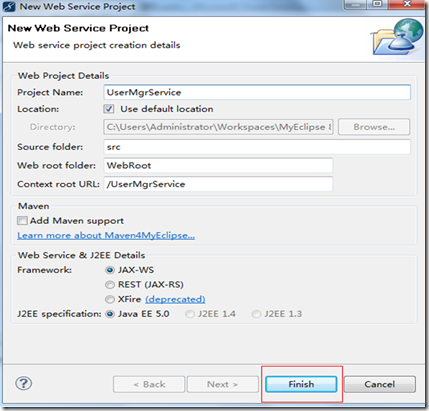
  1 安装MyEclipse

  2 配置Tomcat

   打开MyEclipse

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/20111221164910838.png)

   菜单栏 –> File –> New –> Web Service Project

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/20111221164917202.png)

   在这里，我们使用JAX-WS方式。

**JAX-WS**

  Jax-WS是Java1.6中才有的,新的WebService模式,基于注解的方式配置WebService,很类似Asp中的WebService,难度已经比Xfire方式的配置降低了很多.

**REST(JAX-RS)**

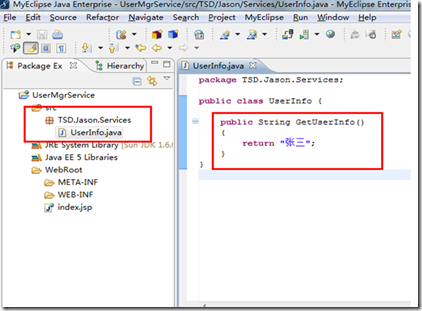
   用的比较少

**Xfire（已过时）**

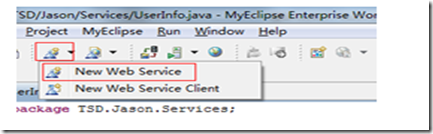
   配置比较繁琐

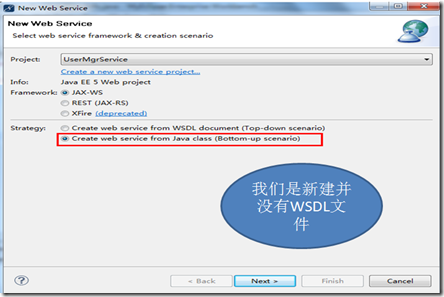
  创建Service中的包及类库，实现自己的业务。

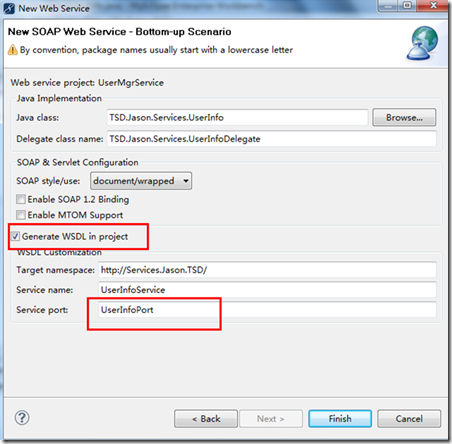
  我们创建的是一个用户管理系统，自然需要来管理用户。

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211649269524.png)

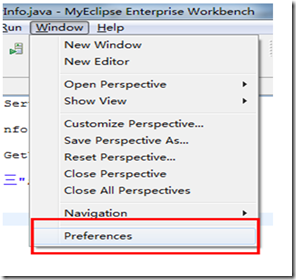
  发布WS

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211649326705.png)

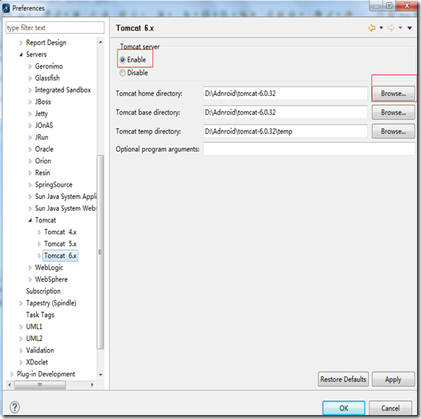
[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211649541590.png)

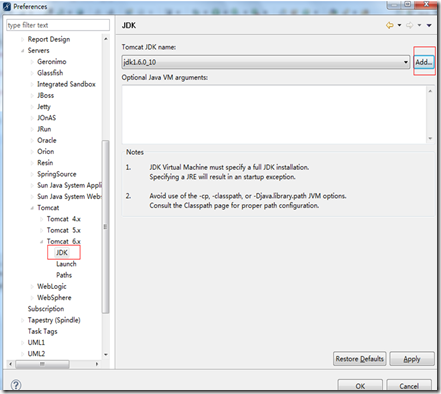
[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211650015730.png)

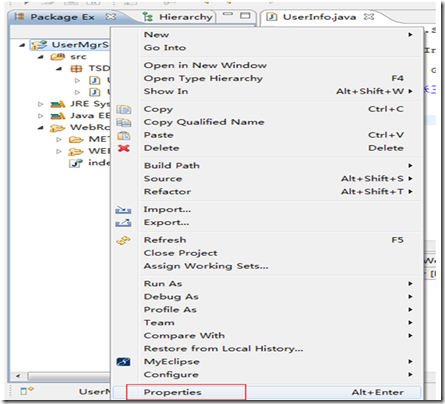
  到此WS发布完成，接下来配置TomCat

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211650319390.png)

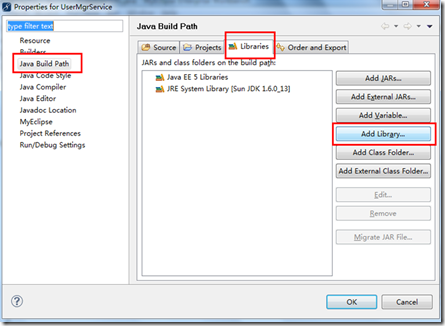
  在弹出的窗体中依次展开 MyEclipse ->Servers –>Tomcat

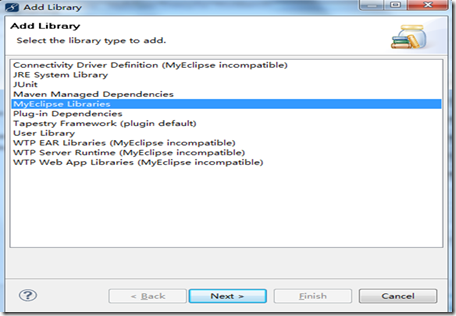
[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211650438726.png)

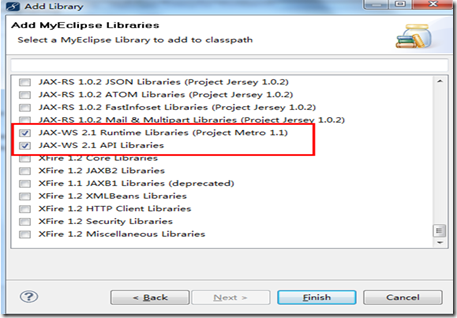
[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211650516030.png)

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211650599887.png)

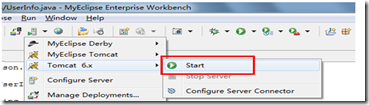
  对着项目右击添加库文件

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211651102405.png)

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211651194758.png)

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211651439319.png)

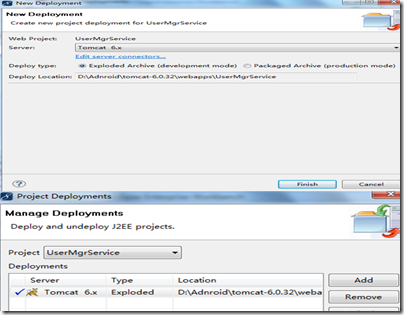
  到此，所有步骤都以配置完毕 开始部署

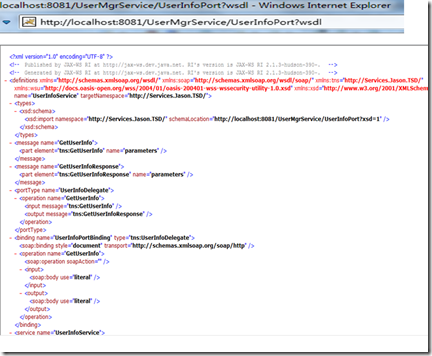
[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211651541554.png)

  启动TomCat

[17](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211652037528.png)

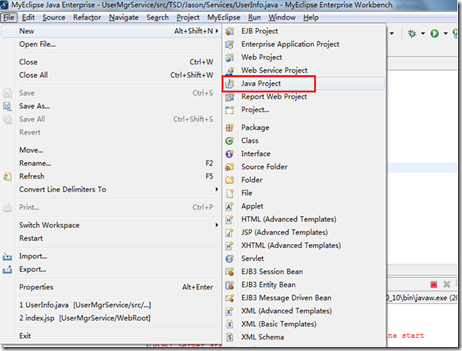
  在工具栏中找到部署图标

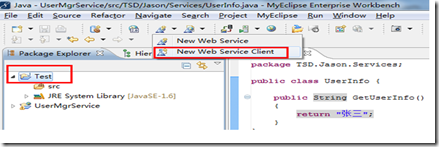
[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211652083040.png)

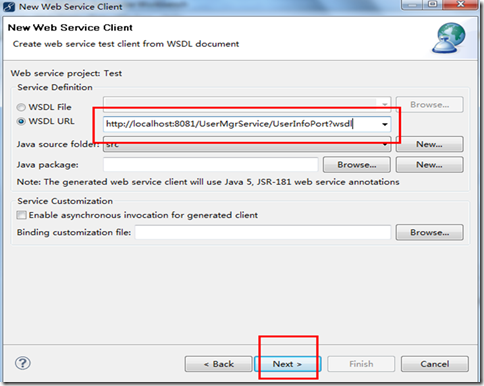
[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211652163342.png)

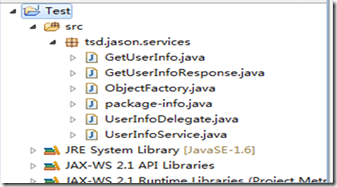
  部署完成，打开IE进行测试

**创建客户端进行测试**

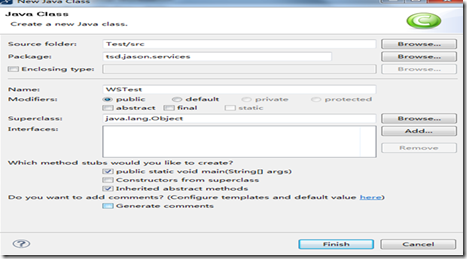
**[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211654036346.png)**

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211654129529.png)

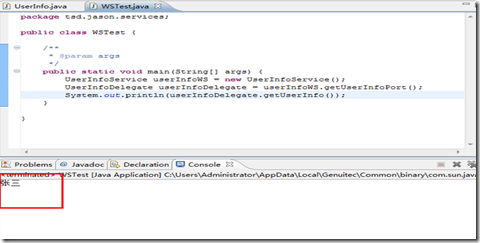
[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211654273382.png)

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211654384570.png)

   此时，项目中将会生成WS所需的类库文件

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211654506905.png)

    创建一个调用测试类

[](http://images.cnblogs.com/cnblogs_com/jasoncc/201112/201112211654567565.png)