1. OSGI概念

OSGI （面向Java的动态模型系统）

直译过来就是“开放的服务入口(网关)的初始化”

OSGi(Open Service Gateway Initiative)技术是Java动态化模块化系统的一系列规范。OSGi一方面指维护OSGi规范的OSGI官方联盟，另一方面指的是该组织维护的基于Java语言的服务（业务）规范。简单来说，OSGi可以认为是Java平台的模块层。

OSGi服务平台向Java提供服务，这些服务使Java成为软件集成和软件开发的首选环境。Java提供在多个平台支持产品的可移植性。OSGi技术提供允许应用程序使用精炼、可重用和可协作的组件构建的标准化原语，这些组件能够组装进一个应用和部署中。

OSGI的简介：<http://blog.csdn.net/jacktan/article/details/8581208>

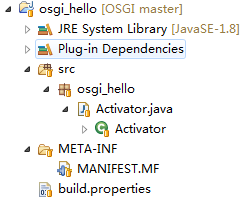
1. HelloWorld

本次开发主要参考了博客：<http://blog.csdn.net/acmman/article/details/50904044>

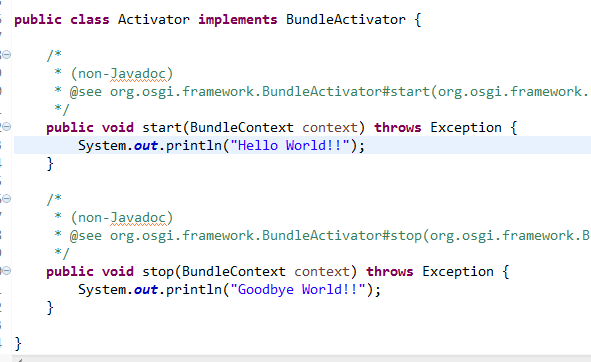
首先建立了第一个Plug-in项目，使用了eclipse自带的OSGI框架，启动插件项目，插件启动的时候在控制台输出Hello World，关闭的时候在控制台输出Goodbye World。



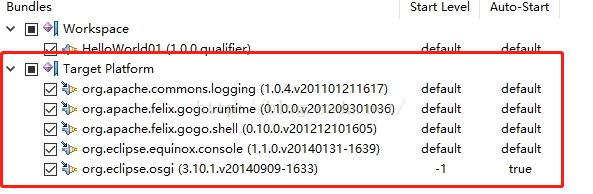
自动生成的代码的层级结构：



Activator控制了bundle的生命周期。其中start和stop方法，分别指定了当此bundle项目ACTIVE的时候和RESOLVED的时候所要执行的动作。



在项目启动的时候需要添加依赖插件，不然无法启动。



1. 翻译助手项目（OSGI）

首先将Service层接口写到TranslateAsslt项目中，这个项目中只需要定义接口即可，然后将接口暴露给其他插件。

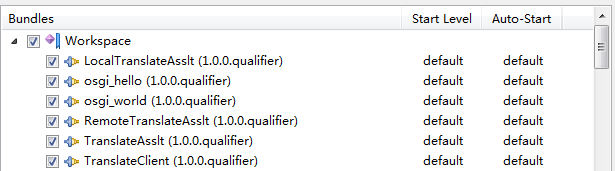
在本地查询LocalTranslateAsslt中进行Service层接口的实现，将TranslateAsslt暴露出来的接口进行引入，再进行实现。

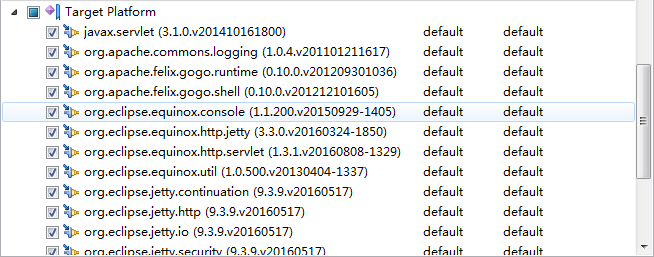
同样，在远程查询RemoteTranslateAsslt中也进行Service层接口的实现。

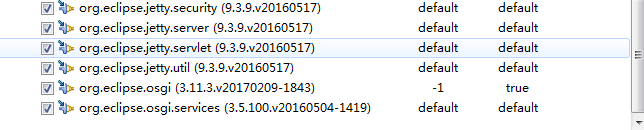
最后进行客户端TranslateClient的实现，首先导入Servlet的相关的包，然后写一个类继承HttpServlet来实现这个Servlet，然后在Activator中实现项目启动和停止的时候分别进行注册和注销Servlet。然后在TranslateServiceServlet中实现doGet和doPost方法。

本地查询和远程查询也需要在Activator中实现插件服务的注册

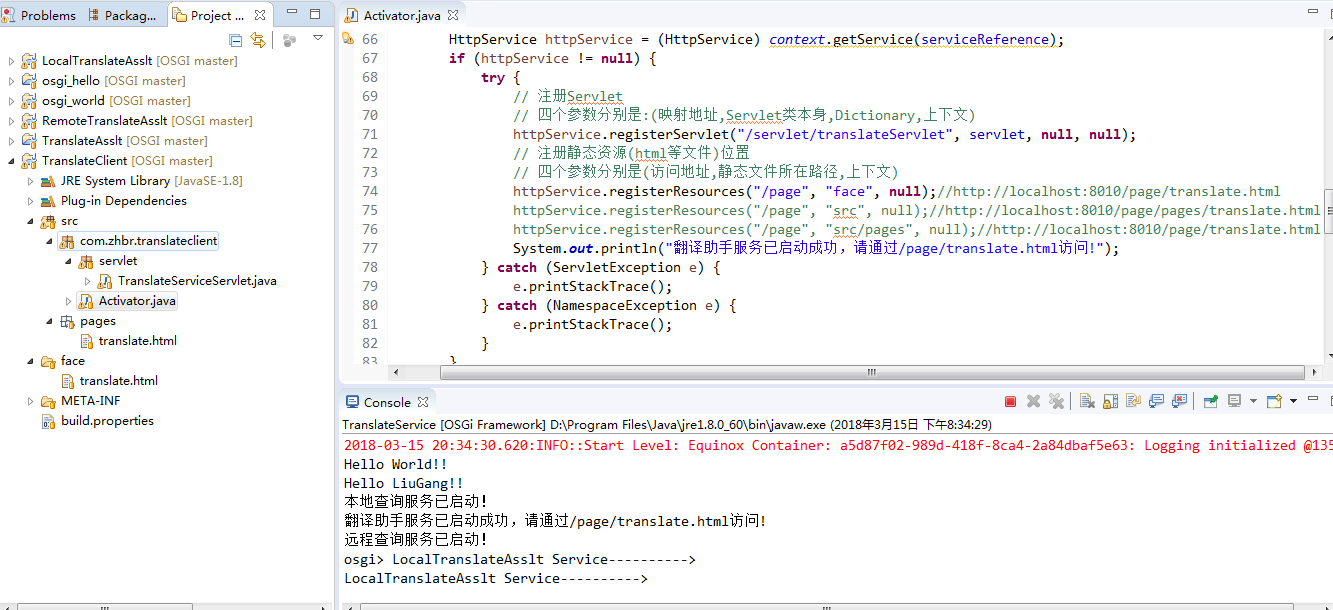
osgi中嵌入的是jetty服务器，所以我们要配置Web的访问环境，将对应的插件进行引入，所有引入的插件如下：





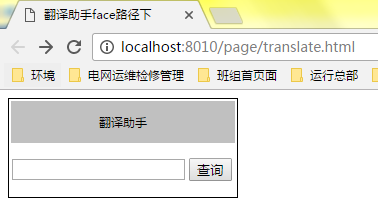


在进行注册静态资源服务路径的时候，按照博客得到了404的页面，后经过试验，发现配置的路径有问题。代码路径及注册路径如下图：



启动服务后，哪个翻译的实现类先启动，在调用的时候就先调用哪个，另一个则无法使用。

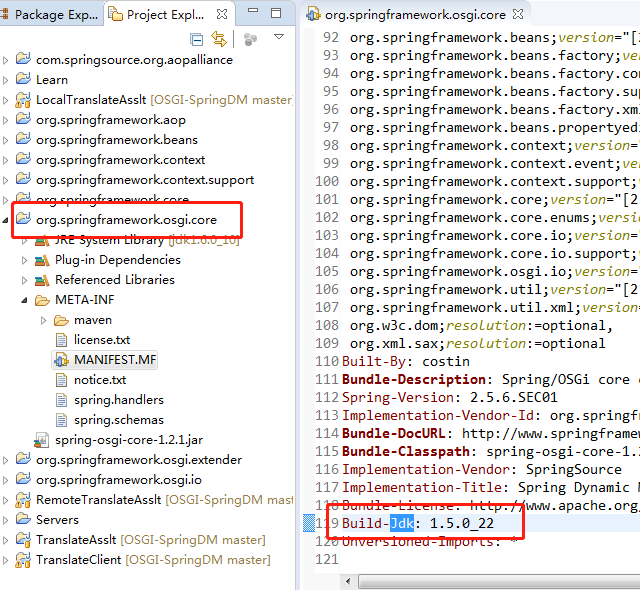
最后页面的展示如图：



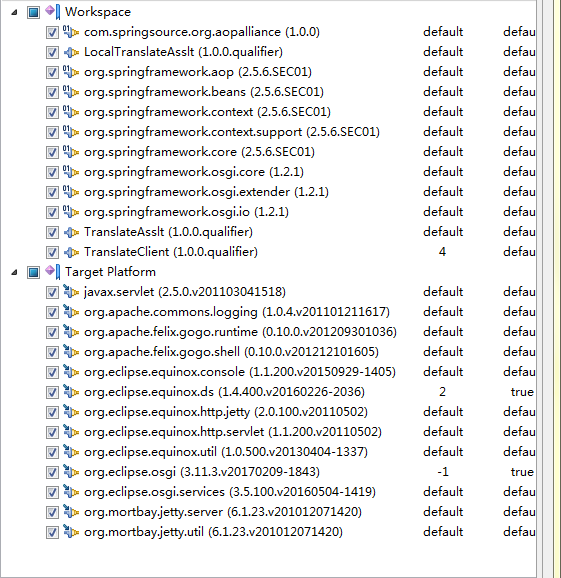
1. 翻译助手项目（SpringDM+OSGI）

在之前的项目中由于并没有引入其他的jar包，我的eclipse中使用的是jdk1.8，所以未出现版本不兼容的问题，在整合SpringDM+OSGI的时候引入了低版本的SpringDM的项目包，导致出现了版本不兼容的问题，SpringDM中org.springframework.osgi.core的构建JDK为1.5，所以只能使用jdk1.6的版本。

查询到的博客：http://blog.csdn.net/bbc2005/article/details/72581280/



同样，由于eclipse中自带的插件都是属于高版本的，在使用了jdk1.6以后，部分jar包出现了版本不兼容的问题，只好重新将不兼容的jar包替换为低版本jar包，所有jar包版本如下：



根据博主的步骤一步一步做完后出现服务可以正常启动，但是翻译助手项目并没有启动，后经研究发现缺少jar包导致，但是在Validate的时候却没有进行提示。



对应的plug插件为

我的eclipse中自带的为3.1.0，需要jdk1.7以上启动，但是引用的framewor包最高支持jdk1.6，所以需要引用低版本的Servlet包。

之后修改完成配置和导入低版本的jar包后服务可以正常启动：

