



[首页](#) [新闻](#) [论坛](#) [问答](#) [专栏](#) [博客](#) [文摘](#) [圈子](#) [招聘](#) [服务](#) [搜索](#)

[Java](#) [Web](#) [Ruby](#) [Python](#) [敏捷](#) [MySQL](#) [普元](#) [Dorado](#) [图书](#) [MSUP开放日](#)

[专栏首页](#) → [Java编程](#) → [SOA](#) → [专栏: Apache CXF 专栏](#)

CXF 2.0 webservice 学习笔记系列_1

原创作者: [evanz](#) 阅读:1531次 评论:1条 更新时间:2007-08-03

对比了一下CXF2 和 AXIS 2 还是选择在框架中集成CXF2。不过CXF2的文档更新的真是慢啊，试了一下，目前CXF2支持的两种绑定方式jaxb2和aegis，看了半天源码总算调通了。又忍不住批一下他的文档，给的都是半调子的例子，希望开发小组能尽快完善啊！

先放一个HelloWold的aegis的配置:

一、server 端:

1. HelloWorld.java

Java代码

```
1. package demo.spring;
2.
3. import javax.jws.WebService;
4.
5. @WebService
6. public interface HelloWorld {
7.     String sayHi(String text);
8. }
```

2. HelloWorldImpl.java

Java代码

```
1. package demo.spring;
2.
3. import javax.jws.WebService;
4.
```

文章信息



专栏: [Apache CXF 专栏](#)

- 由[YuLimin](#)在2007-08-03创建
- 由[YuLimin](#)在2007-08-03更新

相关新闻

- [EJB 3.0和Spring 2.5 : 在Spring中使用EJB 3.0](#)
- [CXF 2.0 发布了](#)
- [60秒创建RESTful Web Services](#)

相关讨论

- [CXF 2.0 webservice 学习笔记系列_1](#)
- [cxf+spring+struts2的helloWorld](#)
- [WebService开发笔记 1 -- 利用cxf开发WebService竟然如此简单](#)
- [如何让CXF导出重载的方法](#)
- [CXF WebService视频教程，自己录的（更新第7集 异步调用 2009.1.15）。](#)

```

5.  @WebService(endpointInterface = "demo.spring.HelloWorld")
6.  public class HelloWorldImpl implements HelloWorld {
7.
8.      public String sayHi(String text) {
9.          return "Hello " + text;
10.     }
11. }

```

相关博客

- [Spring中使用CXF](#)
- [cxf+spring](#)
- [Build Web Service using CXF](#)
- [cxf应用](#)
- [C X F 客户端启动过程](#)

3.beans.xml

Java代码

```

1.  <beans xmlns="http://www.springframework.org/schema/beans"
2.      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3.      xmlns:jaxws="http://cxf.apache.org/jaxws"
4.      xsi:schemaLocation="
5.  http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring
6.  -beans.xsd
7.  http://cxf.apache.org/jaxws http://cxf.apache.org/schemas/jaxws.xsd">
8.
9.      <import resource="classpath:META-INF/cxf/cxf.xml" />
10.     <import resource="classpath:META-INF/cxf/cxf-extension-soap.xml" />
11.     <import resource="classpath:META-INF/cxf/cxf-servlet.xml" />
12.
13.     <bean id="jaxWsServiceFactoryBean"
14.         class="org.apache.cxf.jaxws.support.JaxWsServiceFactoryBean">
15.         <property name="wrapped" value="true" />
16.         <property name="dataBinding" ref="aegisBean" />
17.     </bean>
18.
19.     <bean id="aegisBean"
20.         class="org.apache.cxf.aegis.databinding.AegisDatabinding" />
21.
22.     <jaxws:endpoint id="CollectiveServices"
23.         implementor="demo.spring.HelloWorldImpl" address="/HelloWorld">
24.         <jaxws:serviceFactory>
25.             <ref bean="jaxWsServiceFactoryBean"/>
26.         </jaxws:serviceFactory>
27.     </jaxws:endpoint>
28. </beans>

```

4.web.xml

Java代码

```
1. <web-app version="2.4" xmlns="http://java.sun.com/xml/ns/j2ee"
2.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3.     xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">
4.
5.     <context-param>
6.         <param-name>contextConfigLocation</param-name>
7.         <param-value>WEB-INF/beans.xml</param-value>
8.     </context-param>
9.
10.    <listener>
11.        <listener-class>
12.            org.springframework.web.context.ContextLoaderListener
13.        </listener-class>
14.    </listener>
15.
16.    <servlet>
17.        <servlet-name>CXFServlet</servlet-name>
18.        <servlet-class>
19.            org.apache.cxf.transport.servlet.CXFServlet
20.        </servlet-class>
21.        <load-on-startup>1</load-on-startup>
22.    </servlet>
23.
24.    <servlet-mapping>
25.        <servlet-name>CXFServlet</servlet-name>
26.        <url-pattern>/ws/*</url-pattern>
27.    </servlet-mapping>
28.
29. </web-app>
```

二、客户端

1.ClientTest.java

Java代码

```
1. package demo.spring.client;
2.
3. import static org.junit.Assert.assertEquals;
4.
5. import org.apache.cxf.aegis.databinding.AegisDatabinding;
```

```
6. import org.apache.cxf.frontend.ClientProxyFactoryBean;
7. import org.junit.AfterClass;
8. import org.junit.BeforeClass;
9. import org.junit.Test;
10. import org.springframework.context.support.ClassPathXmlApplicationContext;
11.
12. import demo.spring.HelloWorld;
13.
14. public final class ClientTest {
15.
16.     private static ClassPathXmlApplicationContext context;
17.
18.     @BeforeClass
19.     public static void beforeClass() {
20.         context = new ClassPathXmlApplicationContext(
21.             new String[] { "demo/spring/client/client-beans.xml" });
22.     }
23.
24.     @AfterClass
25.     public static void afterClass() {
26.         context = null;
27.     }
28.
29.     @Test
30.     public void testSayHiWithSpringConfig() throws Exception {
31.
32.         HelloWorld client = (HelloWorld) context.getBean("client");
33.
34.         String response = client.sayHi("Joe");
35.         assertEquals("Hello Joe", response);
36.     }
37.
38.     /**
39.      * 这个官方有例子 (http://cwiki.apache.org/CXF20DOC/aegis-databinding.html)
40.      *
41.      * @throws Exception
42.      */
43.     @Test
44.     public void testSayHiByCode() throws Exception {
45.
46.         ClientProxyFactoryBean factory = new ClientProxyFactoryBean();
47.         factory.setServiceClass(HelloWorld.class);
48.         factory.setAddress("http://localhost:8080/testCxf/ws/HelloWorld");
```

```

49.         factory.getServiceFactory().setDataBinding(new AegisDataBinding());
50.         HelloWorld client = (HelloWorld) factory.create();
51.
52.         assertEquals("Hello Joe", client.sayHi("Joe"));
53.     }
54. }

```

2.client_beans.xml(testSayHiWithSpringConfig())中需要使用的配置文件)

Java代码

```

1. <beans xmlns="http://www.springframework.org/schema/beans"
2.       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3.       xmlns:jaxws="http://cxf.apache.org/jaxws"
4.       xsi:schemaLocation="
5. http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring
6. -beans-2.0.xsd
7. http://cxf.apache.org/jaxws http://cxf.apache.org/schema/jaxws.xsd">
8.
9.     <bean id="aegisBean"
10.          class="org.apache.cxf.aegis.databinding.AegisDataBinding" />
11.
12.     <bean id="serviceFactoryBean" class="org.apache.cxf.jaxws.support.JaxWsServiceFactory
13. Bean">
14.         <property name="dataBinding" ref="aegisBean" />
15.     </bean>
16.
17.     <bean id="client" class="demo.spring.HelloWorld"
18.           factory-bean="clientFactory" factory-method="create" />
19.
20.     <bean id="clientFactory"
21.           class="org.apache.cxf.frontend.ClientProxyFactoryBean">
22.         <property name="serviceFactory" ref="serviceFactoryBean" />
23.         <property name="serviceClass"
24.             value="demo.spring.HelloWorld" />
25.         <property name="address"
26.             value="http://localhost:8080/testCxf/ws/HelloWorld" />
27.     </bean>
28. </beans>

```

[学Java? 找环球培训。](#)

环球培训提供超过5000种IT国际课程, 引导教学, 小班培训, 免费提供课程资料
[peixun.it](#)

Google 提供的广告

[CXF jaxws spring configuration](#) | [我眼中的CXF之 Interceptor \(上\)](#)

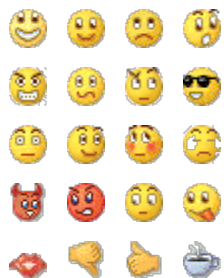
评论 共 1 条 [发表评论](#)

1 楼 [yongsky](#) 2010-02-05 19:20 [引用](#)

这样就完了? 没有总结?

发表评论

表情图标



字体颜色:

字体大小:

对齐:

提示: 选择您需要装饰的文字, 按上列按钮即可添加上相应的标签

您还没有登录, 请[登录](#)后发表评论(快捷键 Alt+S / Ctrl+Enter)

[广告服务](#) | [JavaEye黑板报](#) | [关于我们](#) | [联系我们](#) | [友情链接](#)

© 2003-2010 JavaEye.com. 上海炯耐计算机软件有限公司版权所有 [[沪ICP备05023328号](#)]