

转 Red5开发第一步-Hello World

2013年10月09日 23:39:58 阅读数：6647

安装好Red5服务器之后,接下来就是测试运行众人皆知的Hello World了。按照教程上一步一步来做,可惜一直没运行起来,老是NetConnection.Connect.Rejected,NetConnection.Connect.Closed,郁闷了半天,最后通过与Red5自带的例子SOSample对比,终于给运行起来了。其过程如下:

第一步:把Red5安装目录中doc/templates文件夹下的myapp文件夹copy到webapps目录下,并改名为hello(或任意名称)。

第二步:把hello目录下的log4j.properties文件删掉,修改web.xml,删除不需要的配置信息,修改后的文件内容如下图。

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <web-app
3   xmlns="http://java.sun.com/xml/ns/j2ee"
4   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5   xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd"
6   version="2.4">
7
8   <display-name>hello world</display-name>
9
10  <context-param>
11    <param-name>contextConfigLocation</param-name>
12    <param-value>/WEB-INF/red5-*.xml</param-value>
13  </context-param>
14
15  <context-param>
16    <param-name>webAppRootKey</param-name>
17    <param-value>/hello</param-value>
18  </context-param>
19
20 </web-app>
```

修改red5-web.xml, 修改后的文件内容如下图。

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE beans PUBLIC "-//SPRING/DTD BEAN//EN" "http://www.springframework.org/dtd/spring-beans.dtd">
3 <beans>
4
5   <bean id="placeholderConfig" class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">
6     <property name="location" value="/WEB-INF/red5-web.properties" />
7   </bean>
8
9   <bean id="web.context" class="org.red5.server.Context"
10     autowire="byType" />
11
12   <bean id="web.scope" class="org.red5.server.WebScope"
13     init-method="register">
14     <property name="server" ref="red5.server" />
15     <property name="parent" ref="global.scope" />
16     <property name="context" ref="web.context" />
17     <property name="handler" ref="web.handler" />
18     <property name="contextPath" value="/${webapp.contextPath}" />
19     <property name="virtualHosts" value="${webapp.virtualHosts}" />
20   </bean>
21
22   <bean id="web.handler"
23     class="com.flexer.Hello"
24     singleton="true" />
25
26 </beans>
```

修改red5-web.properties, 将webapp.contextPath的值改为/hello。

第三步:写Java代码

打开Eclipse,建立一个Java Project,命名为hello,并将该Project地址设为我们刚才在Red5安装目录的webapps目录下新建的hello文件夹,点击Next按钮,在这个界面中点击Browse按钮,在WEB-INF中新增classes目录,将其设置为output folder,然后点击Finish。(此处要注意,output folder一定要设置在WEB-INF目录下)。建好Project后别忘了引入位于Red5安装目录下的red5.jar。Java代码如下。

```
[java]
1. package com.flexer;
2. import org.red5.server.adapter.ApplicationAdapter;
3. public class Hello extends ApplicationAdapter{
4.     public String hello()
5.     {
6.         return "Hello World!";
7.     }
8. }
```

然后编译,编译完成后重启Red5服务。

第四步:写Flex代码。

打开Flex Builder 3, 建立Flex Project, 代码如下。

```
[html]  
1. <?xml version="1.0" encoding="utf-8"?>
2. <mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute" initialize="init()" fontSize="12">
3.   <mx:Script>
4.     <![CDATA[
5.       import mx.controls.Alert;
6.       private var nc:NetConnection=null;
7.       private var responder:Responder=null;
8.       private function init():void
9.       {
10.         nc=new NetConnection();
11.         nc.addEventListener(NetStatusEvent.NET_STATUS,ncHandler);
12.         nc.connect("rtmp://localhost/hello");
13.       }
14.       private function ncHandler(event:NetStatusEvent):void
15.       {
16.         if(event.info.code=="NetConnection.Connect.Success")
17.         {
18.           responder=new Responder(result,error);
19.           nc.call("hello",responder);
20.         }
21.       }
22.       private function result(obj:Object):void
23.       {
24.         Alert.show(obj.toString());
25.       }
26.       private function error(err:Object):void
27.       {
28.         Alert.show(err.toString());
29.       }
30.     ]]>
31.   </mx:Script>
32. </mx:Application>
33.
```

完成后，编译运行。

至此，Hello World完工！

原文地址：<http://blog.csdn.net/wuguoyao/article/details/4385667>

文章标签：[red5](#) [flex builder](#) [服务器](#) [eclipse](#)

个人分类：[libRTMP](#) [J2EE](#) [Flash相关](#)

此PDF由spygg生成,请尊重原作者版权!!!

我的邮箱:liushidc@163.com