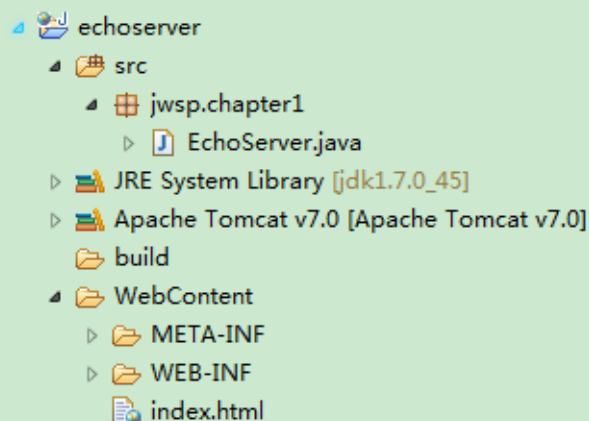
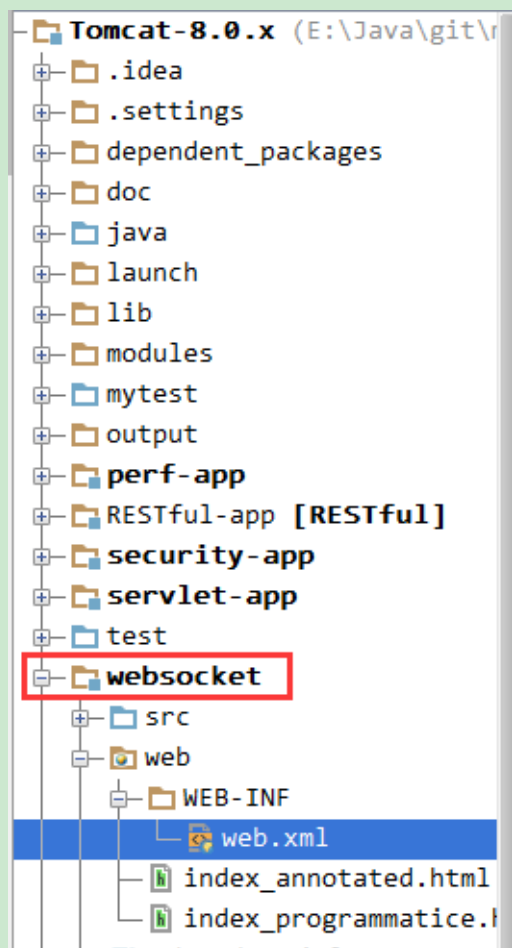


在 Tomcat 源码中调试 websocket 的时候,有一些额外的工作要做.如果没有这些操作,则 websocket 程序可能无法被正确部署.如果我们并没有在源代码中调试 websocket,而是将 websocket 应用程序直接部署到 Tomcat 中,如下图:



这里我们直接部署 echoserver 程序即可运行 websocket 示例,但是我们在 Tomcat 调试程序中直接这么做的话是不行的,如下图:



会发现示例一直无法运行,出现如下的错误:

Echo Server

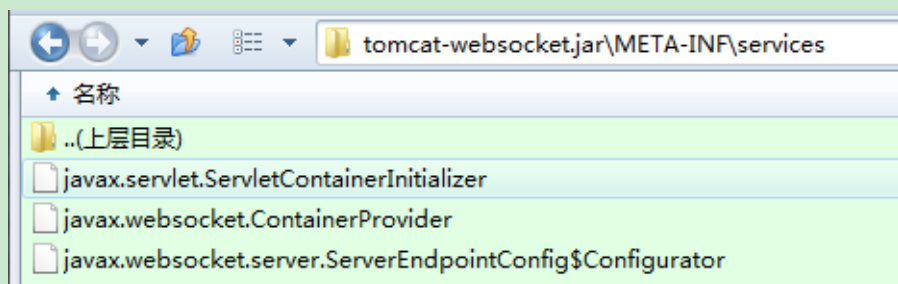
Press to send

Hello Web Sockets

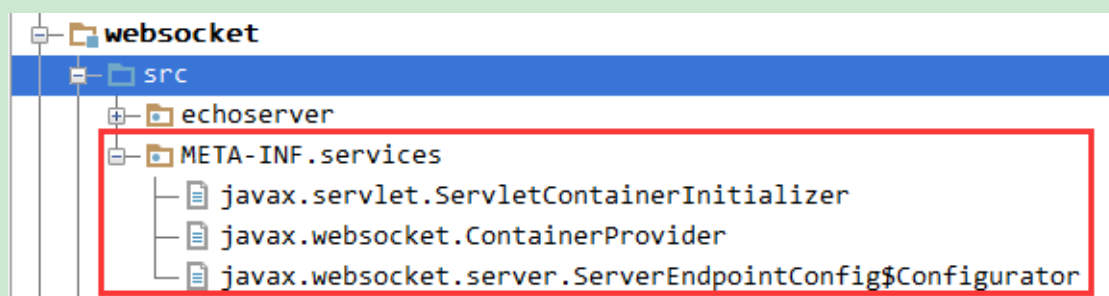
Connecting to ws://localhost:8080/websocket/annotatedecho

ERROR: undefined

原因是因为在二进制包的 Tomcat 的 lib 目录中的 `tomcat-websocket.jar` 文件,打开之后可以发现有如下的文件:



这就不用多说了,很简单,将这些文件拷贝到我们的项目中,当然也可以放到 Tomcat 的源码中。



不过此时启动 Tomcat 部署 websocket 的时候还是有问题的,会出现重复路径的问题:

```
Caused by: javax.websocket.DeploymentException: Multiple Endpoints may not be deployed to the same path [/]  
    at org.apache.tomcat.websocket.server.WsServerContainer.addEndpoint(WsServerContainer.java:190)  
    at org.apache.tomcat.websocket.server.WsServerContainer.addEndpoint(WsServerContainer.java:237)  
    at org.apache.tomcat.websocket.server.WsSci.onStartup(WsSci.java:112)  
    ... 8 more
```

这里不得不吐槽一下,这个错误实在是让人没法看,只知道有重复路径,但是如何找出相应的配置.却无从下手.

先根据异常定位到 `WsServerContainer#addEndpoint()` 方法,在如下的代码中添加一条打印语句:

```
1. System.out.println("path: " + path + ", EndpointClass: " + sec.getEndpointClass());
```

```
// Exact match  
ServerEndpointConfig old = configExactMatchMap.put(path, sec);  
System.out.println("path: " + path + ", EndpointClass: " + sec.getEndpointClass());  
if (old != null) {  
    // Duplicate path mappings  
    // 重复的路径映射  
    throw new DeploymentException(sm.getString("serverContainer.duplicatePaths", path));  
}
```

此时再运行,就会可以看到是两个端点的路径重复:

```
path: /websocket/echoAnnotation, EndpointClass: class websocket.echo.EchoAnnotation  
path: /, EndpointClass: class org.apache.tomcat.websocket.pojo.TestPojoEndpointBase$OnOpenServerEndpoint  
path: /echoAsync, EndpointClass: class org.apache.tomcat.websocket.TesterEchoServer$Async  
path: /block, EndpointClass: class org.apache.tomcat.websocket.TestWsWebSocketContainer$BlockingPojo  
path: /, EndpointClass: class org.apache.tomcat.websocket.pojo.TestPojoEndpointBase$Bug54716
```

看下 `TestPojoEndpointBase.OnOpenServerEndpoint` 类，随便改个名称即可。

```
@ServerEndpoint("/")
public static class OnOpenServerEndpoint {
```

不过既然我们已经知道错误信息不太友好，就可以对代码进行修改，主要是要确定是哪两个端点的路径重复，做如下的调整：

- 修改 `org/apache/tomcat/websocket/server/LocalStrings.properties`：

```
1. #serverContainer.duplicatePaths=Multiple Endpoints may not be deployed to the same path [{0}]
2. serverContainer.duplicatePaths=Multiple Endpoints may not be deployed to the same path [{0}], the first endpoint is [{1}], the second endpoint is [{2}]
```

- 修改 `WsServerContainer#addEndpoint()` 方法：

```
1. if (old != null) {
2.     // Duplicate path mappings
3.     // 重复的路径映射
4.     throw new DeploymentException(sm.getString("serverContainer.duplicatePaths", path, old.
        getEndpointClass().getName(), sec.getEndpointClass().getName()));
5. }
```

此时启动错误信息就比较明显了：

```
Caused by: javax.websocket.DeploymentException: Multiple Endpoints may not be deployed to the same path [/], the first endpoint is [org.apache.tomcat.websocket
.pojo.TestPojoEndpointBase$OnOpenServerEndpoint], the second endpoint is [org.apache.tomcat.websocket.pojo.TestPojoEndpointBase$Bug54716]
at org.apache.tomcat.websocket.server.WsServerContainer.addEndpoint(WsServerContainer.java:189)
at org.apache.tomcat.websocket.server.WsServerContainer.addEndpoint(WsServerContainer.java:236)
at org.apache.tomcat.websocket.server.WsSci.onStartup(WsSci.java:112)
... 8 more
```

不过我们仍然需要考虑另外一种情况，就是通过编程式方式配置 websocket 路径的情况。此时我们需要找到 `ServerApplicationConfig` 实现的配置。这样就可以定位到路径和端点的映射。