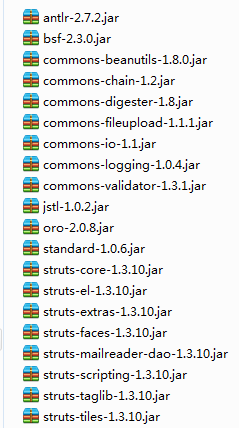
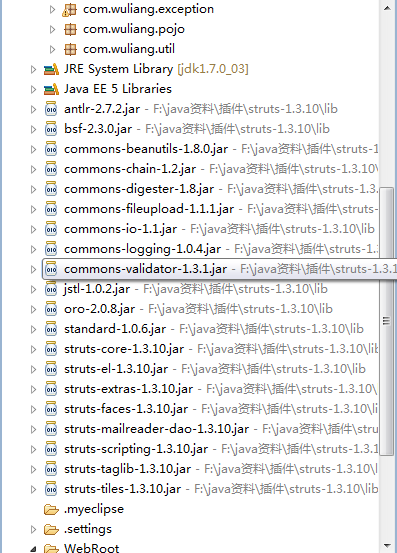
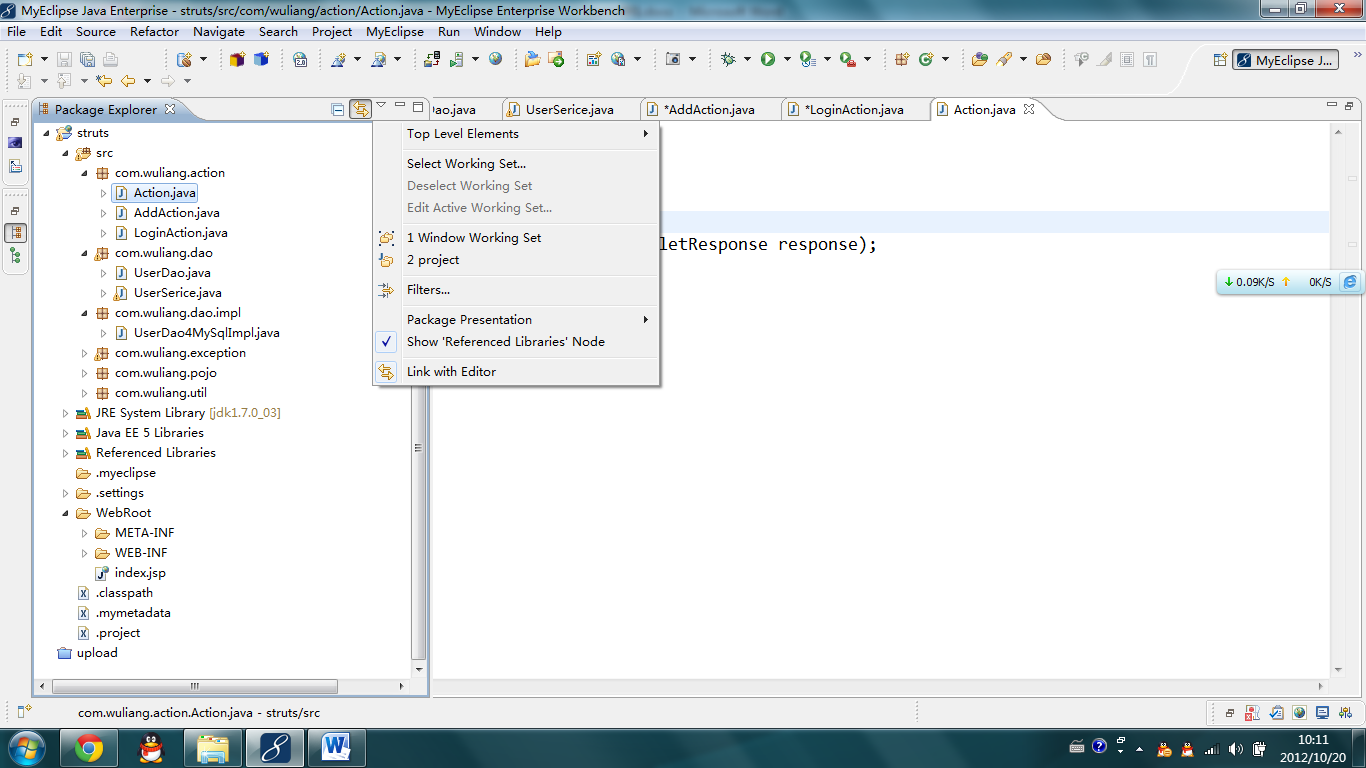
Struts1

1. 添加架包（Strust全部架包）

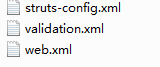
出现问题

就是所有架包都裸露在外，解决方法





就解决了

1. 解压空架包F:\java资料\插件\struts-1.3.10\apps\struts-blank-1.3.10.war解压找到WEB-INF得到复制web.xml到项目中的web.xml，在把struts-config.xml整个文件复制到项目中WEB-INF中，再删除其中的文件，只保留<struts-config>标签，其中
2. 自己建LoginAction，继承类Action，重写execute方法

其中重写executet中的方法是Httpservlet上的

1. 重写以后需要调用其中的方法

**public** **class** LoginAction **extends** Action {

@Override

**public** ActionForward execute(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response)

**throws** Exception {

LoginActionForm la = (LoginActionForm)form;

String username = la.getUsername();

String password = la.getPassword();

//String username = request.getParameter("username");

//String password = request.getParameter("password");

String path = "no";

**try**{

UserDao.*login*(username, password);

path = "succee";

}**catch**(UsernameNotFoundException e) {

}**catch**(PasswordNotFoundException e) {

}

**return** mapping.findForward(path);

}

}

5．其中的<struts-config>

<struts-config>

<form-beans>

<form-bean name=*"LoginForm"*

type=*" com.wuliang.actionForm.LoginActionForm"*>

</form-bean>

//name=区别接收的url参数名称，type是请求接收的类名

</form-beans>

<action-mappings>

//action中path是请求的地址，其中必须是…….do类型，而且必须需要“/”，//type是指定请求的分配到哪action上，类名

//forword可以指定其中跳转的地址，根据传过来的name来动态跳转（path的值必须是有“/”）

<action path=*"/login"*

type=*"com.wuliang.action.LoginAction"*

name=*"LoginForm"*>

<forward name=*"succee"* path=*"/login.jsp"*></forward>

<forward name=*"no"* path=*"/no.jsp"*></forward>

</action>

</action-mappings>//mappings说明有多个action，根据

</struts-config>

1. struts自动接收参数ActionForm

一： 继承ActionForm//自动接收URL参数

二：写入PO

**public** **class** LoginActionForm **extends** ActionForm {

**private** String username;

**private** String password;

**public** String getUsername() {

**return** username;

}

**public** **void** setUsername(String username) {

**this**.username = username;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

}

1. 问题1 ：No action config found for the specified url.

说明写错了<struts-config>没有path名称

问题2: Path ./no.jsp does not start with a "/" character

解决方法是：<struts-config>中<froword 上path必须”/地址”

问题3：

总是没有拿到值，当报值为null，或者为初始值时，则是由于

没有使用AutoActionForm aaf = (AutoActionForm)form;

而是新new AutoActionForm();就得不到表单上面的值

解决struts中的Action与ActionForm的设计膨胀的问题

1. 解决ActionForm设计膨胀：

**1.把原来的<struts-config>中<form-bean>改为**

<form-beans>

<form-bean name=*"LoginForm"*

type=*"org.apache.struts.action.DynaActionForm"*>

<form-property name=*"username"* type=*"java.lang.String"*/>

<form-property name=*"password"* type=*"java.lang.String"*/>

</form-bean>

</form-beans>

**2. 删除原来的ActionForm**

**3. 修改Action，结果**

DynaActionForm daf = (DynaActionForm)form;

String username = daf.getString("username");

String password = daf.getString("password");

1. 解决Action设计膨胀：
2. 其中Action全部删除，继承dispatchAction
3. 把每类只写一个种Action，例如用户中就全部写在UserAction，每个操作用方法代替
4. 在<struts-config>中action写入parameter=*"command"，修改其中的类名，并且还需要使用在跳转前的页面加入*form action=*"./login.do?command=login"* method=*"post"*>就可以了
5. UserAction

**public** **class** UserAction **extends** DispatchAction {

**public** ActionForward login(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) **throws** Exception {

DynaActionForm daf = (DynaActionForm)form;

String username = daf.getString("username");

String password = daf.getString("password");

String path = "no";

**try**{

UserDao.*login*(username, password);

path = "succee";

}**catch**(UsernameNotFoundException e) {

}**catch**(PasswordNotFoundException e) {

}

**return** mapping.findForward(path);

}

**public** ActionForward add(ActionMapping mapping, ActionForm form,

HttpServletRequest request, HttpServletResponse response) **throws** Exception {

String path = "succee";

System.*out*.println("---------add----------------");

**return** mapping.findForward(path);

}

}

配置文件：

<struts-config>

<form-beans>

<form-bean name=*"LoginForm"*

type=*"org.apache.struts.action.DynaActionForm"*>

<form-property name=*"username"* type=*"java.lang.String"*/>

<form-property name=*"password"* type=*"java.lang.String"*/>

</form-bean>

</form-beans>

<action-mappings>

<action parameter=*"command"*

path=*"/login"*

type=*"com.wuliang.action.UserAction"*

name=*"LoginForm"*>

<forward name=*"succee"* path=*"/login.jsp"*></forward>

<forward name=*"no"* path=*"/no.jsp"*></forward>

</action>

</action-mappings>

</struts-config>

1. 解决ActionForm 动态跳转问题

在方法中写入

ActionForward af = **new** ActionForward();

af.setPath("/ok.jsp+page"+page);

**return** af;

这样就动态的改变了跳转的方向，使用在分页上是最好的

ActionForward af = mapping.findForward("succee");

af.setPath("/ok.jsp");

**return** af;

这样会报500错误，由于mapping。findForward是<struts-config>中的确定的地址栏，不能setPath修改已确定的文件，在读取配置文件是在初始化中，服务器一开启的时候配置文件被读取，编译成对象，所以一旦修改配置文件的时候必须重新开启服务器，但不容许在编译过程中修改配置文件

4.一些注意的地方:

1.如果设置找不到原来的界面为一个固定的界面，就用使用unknown的，当然，其中也必须使用“…….do”,才可以

<action path=*"/unknown"*

unknown=*"true"*

scope=*"request"*

forward=*"/404.jsp"*>

</action>

2.可以不经过action就可以这直接到固定的界面

<action path=*"/welcome"*

forward=*"/index.jsp"*>

</action>

1. 在配置文件中

**是默认request.getSSession传值，所有最好先设置好**scope=*"request"*

<exception type=*"com.i18n.exception.PasswordErrorEcxeption"* key=*"login.error"*></exception>

<exception type=*"com.i18n.exception.UsernameNotFormException"* key=*"login.found"*></exception>