

学生应知信息知识

JAVA 知识手册 (六)

狄登峰 主编

# 目 录

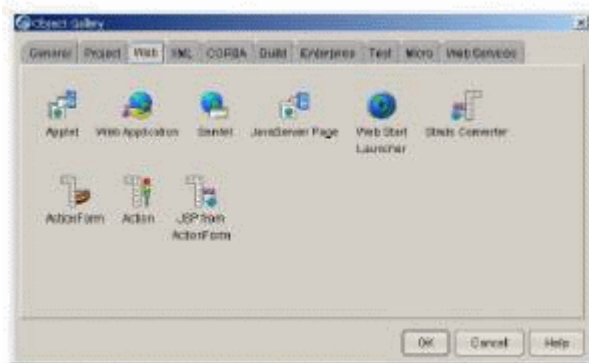
用 JBuilder9 开发 Struts 实例 .....	1
Struts 框架之构建 Model 组件 .....	15
Java 中 Decorate 的三种实现方法 .....	29
Struts 框架之构建 Controller 组件 .....	48
触摸 Java 中的包和文档 .....	63
用 EJB 开发在线课堂 .....	72
Java 多线程编程初步 .....	118
Java 模式设计之数据访问对象模式 .....	128

## 用 JBuilder9 开发 Struts 实例

本文将给大家介绍在 JBuilder9 集成开发环境下,如何运用 StrutsFramework 来开发 Web 应用程序,我们这里所说的 JBuilder9 是指 JBuilder9Developer 与 Enterprise 两个版本。

### 一、JBuilder 对于 Struts 之支持

JBuilder 提供图形使用者接口以简化组件文件 struts-config.xml 之编辑与设定,此外, JBuilder 于 ObjectGallery 对话框内的 Web 页次,提供四种精灵以简化 Struts 开发流程。



StrutsConverterWizard :

可指定 HTML 或一般 JSP 档案自动产生使用 StrutsTagLibraries 之 JSP 档案。

ActionFormWizard :

可导入 JSP 档案产生继承自 ActionForm 之类 ,或自定义属性(property)产生继承自 ActionForm 之类 ,类产生后可选择是否自动完成 struts-config.xml 对应设定。

ActionWizard :

可产生继承自 Action 之类 , 设定搭配之 JSP 档案、有效范围(scope)并自动完成 struts-config.xml 对应设定。

JSPfromActionFormWizard :

可由指定之 ActionForm 自动产生使用 StrutsTagLibraries 之 JSP 档案。

除了上述精灵外 , JBuilder 可协助开发人员自动设定 web.xml 部署描述档 , 以便 JSP 使用 Struts 自定义 TagLibraries。

## 二、使用 Jbuilder 开发 struts 实例

我们在这里举开发一个简易的 Web 应用程序的例子来帮助大家了解 StrutsFramework。此 Web 应用程序允许使用者输入两独立字符串 , 并将两字符串连接后显示于网页。由于处理逻辑过于简单 , 此实例中我们打算直接在扮演 Controller 角色的 Action 中完成两字符串连接动作 , 因此未使用扮演 Model 角色之 JavaBean 或类来处理字符串连接。

### 步骤一:建立 Web 应用程序

首先点击选择 JBuilder 集成开发环境主菜单

File|NewProject... 建立一个名为 StrutsDemo 之项目。接着由菜单 File|New... 激活 ObjectGallery 对话框，并点击选择 Web 页次中的 WebApplication 精灵准备建立 Web 应用程序。

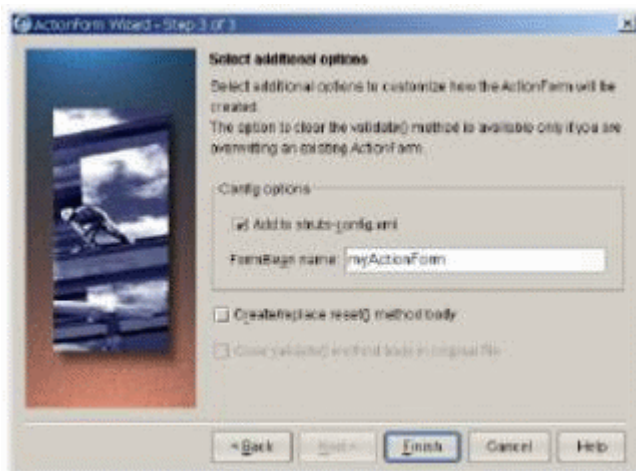


WebApplication 精灵会为此 Web 应用程序准备妥产生 WAR 所需之部署描述档 web.xml，WAR 是 J2EE 规格中用以简化 Web 程序部署复杂度之标准档案格式。我们为此 Web 应用程序命名为 demo，往后将会在编译此项目档时产生名为 demo.war 档案以便于部署，在此 Web 应用程序中由于将会用到 Struts 之 TagLibraries，因此在 JSP/Servletframeworks 中勾选 Struts1.0，同时 JBuilder 会自动设定妥 web.xml。

## 步骤二:建立自定义之 ActionForm 类

StrutsFramework 中并未限制 MVC 三部份开发顺序，由于此时我们并未准备妥可输入两字符串之 HTML 或 JSP，因此先建立储存信息之 ActionForm 类，再产生

扮演 Controller 角色之 Action 类，最后再由 ActionForm 类产生 JSP 档案，大家可依状况调整顺序以适应实际案例。点击选择 ObjectGallery 对话框 Web 页次中 ActionFormWizard 可产生自定义之 ActionForm 类，ActionFormWizard 又可分为三步：第一步精灵自动侦测出之前已建妥之 Web 应用程序 demo，我们将自定义 ActionForm 名称由预设 Untitled1ActionForm 修改为 myActionForm，接着按下 Next 按钮进入第二步。第二步依照需求将未来 JSP 页面上所需显示与输入之字段加入 Fielddefinition 清单方块中，倘若负责网页美工专职人员已备受输入用 JSP 网页，亦可按下 AddfromJSP... 按钮直接导入，JBuilder 将解析此 JSP 档案产生 Fielddefinition 表格。本例中由于未准备输入之 JSP 档案，因此必须自行加入所需之字段。我们为此 Web 应用程序预先准备三个字段，fieldA 与 fieldB 可储存输入之字符串，fieldResult 可置放两字符串连接后结果。设定完成后按下 Next 按钮进入第三步。

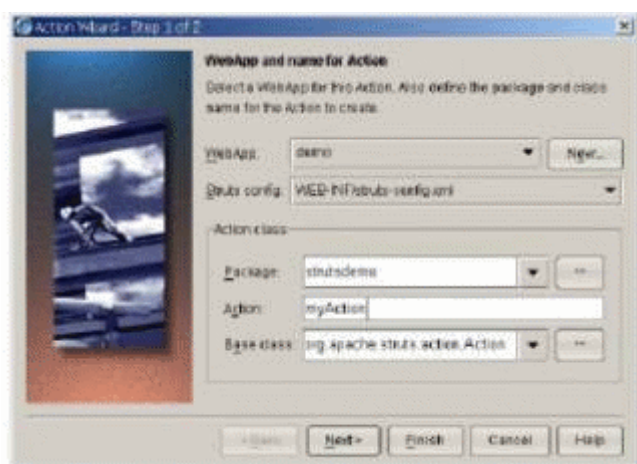


步骤三对话框提示是否将 myActionForm 加入 struts-config.xml 组件档中？是否需要产生 reset 方法？可依照默认值按下 Finish 按钮。

此时 ActionFormWizard 已经自动产生了继承自 ActionForm 之 myActionForm.java 档案，如果我们不打算实现资料核验之 validate 方法，我们完全无须撰写任何程序代码。

### 步骤三:建立自定义之 Action 类

点击选择 ObjectGallery 对话框 Web 页次中 ActionWizard 可产生自定义之 Action 类，ActionWizard 分为两步：第一步精灵自动侦测出之前已建立之 Web 应用程序 demo 与 struts-config.xml 组件档，我们将自定义 Action 名称由预设 Untitled1Action 修改为 myAction，接着按下 Next 按钮进入第二步。



在第二步对话框中，我们可以设定 myAction 的

perform 方法会传入哪一个 ActionForm，在 FormBeanName 组合方块中，所有记录于 struts-config.xml 中的 ActionForm 都会列出以供选用，此外我们设定 myAction 之有效范围(Scope)为 Session，并指明/input.jsp 所有输入动作，将交由 myAction 处理。

按下 Finish 按钮后即产生 myAction.java 档案。在 myAction.java 中，JBuilder 已经自动产生了 perform 方法与尚未实现之例外程序代码，我们将 perform 内的程序代码做以下之修改，已完成字符串连接的动作：

```
myActionFormmyActionForm=(myActionForm)ActionForm;
```

```
StringsA=myActionForm.getFieldA();
```

```
StringsB=myActionForm.getFieldB();
```

```
myActionForm.setFieldResult(sA+sB);
```

```
return(actionMapping.findForward("input"));
```

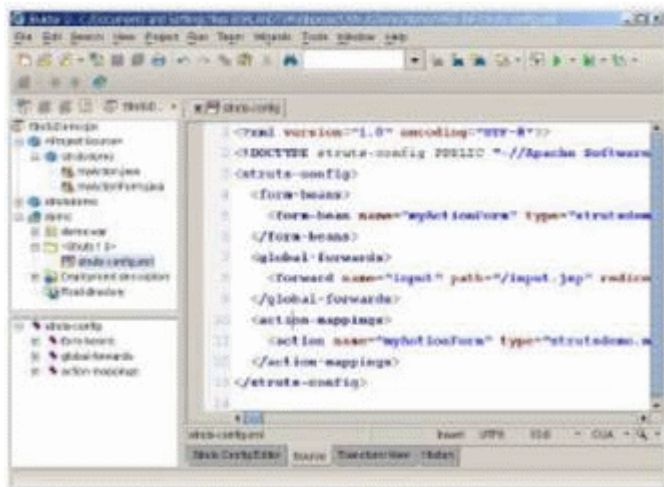
由于 perform 方法会传入参数 ActionForm 以便获知 Web 浏览器内使用者输入之字符串，在此 JBuilder 已将 ActionForm 转型(TypeCast)为 myActionForm 之程序代码产生完成，开发人员仅需以 getFieldA 与 getFieldB 方法即可得知使用者输入之字符串，并将连接后结果以 setFieldResult 方法回存 fieldResult 属性。return 将决定负责显示之 JSP 网页，利用 perform 方法传入参数 actionMapping 之 findForward 方法，可用以指定显示之

JSP 网页。之前精灵对话框中已预设/input.jsp 为负责输入之 JSP 网页，Struts 中允许输入与显示同为一个 JSP 网页，因此我们需在 struts-config.xml 中定妥名为 input 之 Forward 标记，以便 findForward 方法能够利用此标记得知真正 JSP 网页名称/input.jsp。

#### 步骤四:设定 Forward 标记

点击选择 JBuilder 左上方之项目窗格(ProjectPane) 内之 struts-config.xml，此时集成开发环境便可以图形使用者接口方式编辑 struts-config.xml 内容，左下方结构窗格(StructurePane)会依照 struts-config.xml 结构，分为 DataSources、FormBeans、GlobalForwards 与 ActionMappings 四类标记项目。

当点击选择 GlobalForwards 项目后 JBuilder 右方内容窗格(ContentPane)将出现如图所示之编辑画面，开发人员可按下 Add 按钮加入名为 input 之标记，并指明此标记对应路径为/input.jsp。



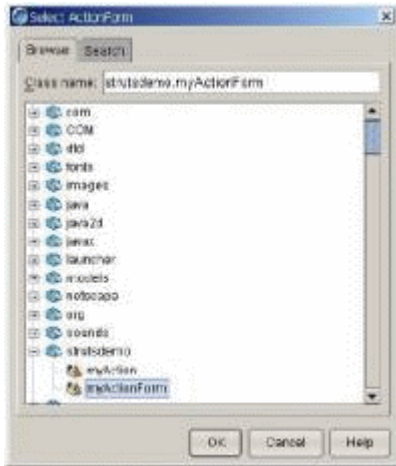
对于已经熟悉 StrutsFramework 的开发人员,若欲直接修改 struts-config.xml 内容,可以点击选择内容窗格中的 Source 页次,此时内容窗格如图所示将以文字编辑器显示 XML 内容,我们可自由选择习惯之方式维护 struts-config.xml 内容。

### 步骤五:产生 JSP

接下来我们将利用现有 myActionForm.java 产生 /input.jsp,点击选择 ObjectGallery 对话框 Web 页次中 JSPfromActionFormWizard,将出现以三步产生 JSP 之精灵:

在第一步对话框中,我们可将欲产生的 JSP 档名由预设之 UntitledJsp1.jsp 改为/input.jsp,并可按下[...]按钮以选取 ActionForm。ActionForm 选取对话框有两种查找模式,我们可以浏览 JavaPackage 树状结构直接选取 myActionForm。

或是键入 Java 类前缀名称,JBUILDER 将会自动查找相符之类。例如我们键入 myActionForm 的前缀 my,将可见到对话框查找到两个相符之类。此一便捷选取类别之对话框,可在 JBUILDER 编辑器中以热键 Ctrl-Alt-H 随时激活使用。

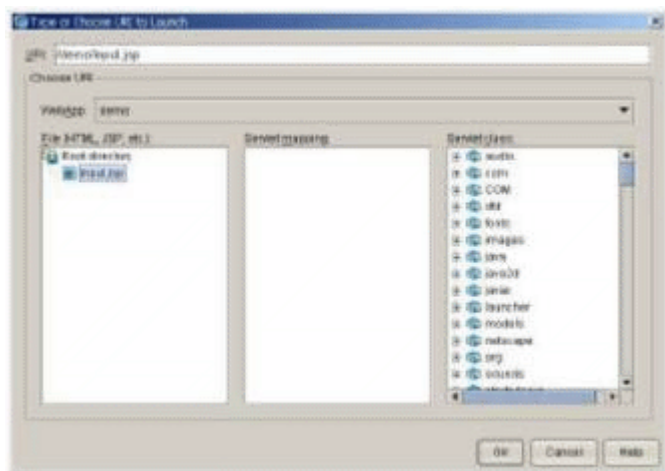


选取 myActionForm 完毕后，按下 Next 按钮可进入至第二步对话框。JBuilder 将分析 myActionForm.java 内容显示其字段，我们可以组合方块决定该字段将以何种 StrutsTag 搭配显示，本例中三个字段皆使用可输入文字、预设之 text 型态，按下 Next 按钮即进入第三步。第三步对话框针对 StrutsTagLibraries 特性，提供多种选项，并且询问是否将全部之 StrutsTagLibraries 宣告导入至 JSP 档中。本例我们采用默认值，按下 Finish 按钮后 JBuilder 即产生/input.jsp。

#### 步骤六:设定执行组件

接着我们可以设定执行组件以测试本程序。点击选择 JBuilder 集成开发环境主菜单 Project|ProjectProsperities... 将出现项目属性设定之对话框，点击选择 Run 页次准备设定执行组件。JBuilder 允许在单一项目中拥有多个执行组件，可执行 JavaApplication、JavaApplet、J2MEMIDlet 及 J2EE 各式 Server 端应用程序。

本例中将建立一个 Server 类型之执行组件。按下对话框右侧 New... 按钮将会出现执行组件属性设定对话框，以新增一个执行组件。



执行组件属性设定对话框中，我们可以键入一识别用途之名称于 Name 字段，本例中我们键入中文识别名称"执行 input.jsp"，在 Run 页次 Type 组合方块中可选取 Server 类型应用程序，由于本例我们采用 JBuilder 内置之 Tomcat4.0 为 JSP/ServletContainer，因此 Server 下拉盒选取 Tomcat4.0。倘若我们的 Server 组合方块中无适当选项可供选取，可于 JBuilder 主菜单 Tools|ConfigureServer... 中进行相关设定。接着我们必须选取此 Web 应用程序激活 URI，我们可按下 LaunchURI 字段右侧[...]按钮，即可出现选取 URI 之对话框，我们可直接以鼠标点击选择左侧 input.jsp，按下 OK 按钮完成选取，并返回执行组件属性设定对话框。

此时执行组件属性设定对话框中主要项目皆已设定完毕，按下 OK 按钮返回项目属性设定对话框。项目属

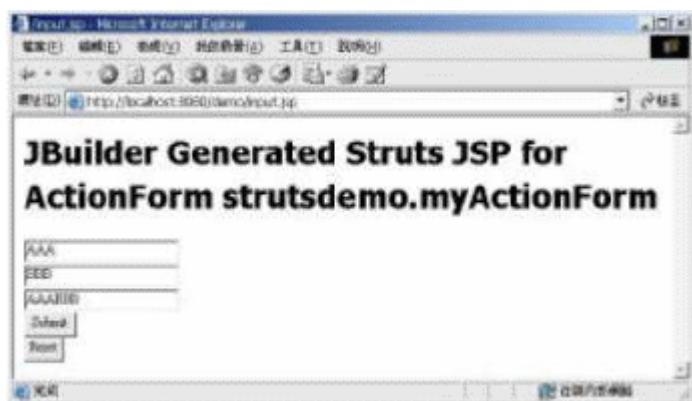
性对话框中将把前述步骤所设定之执行组件列于 RuntimeConfigurations 清单方块中，按下 OK 按钮即完成执行组件设定。

### 步骤七:执行 Web 应用程序

接下来可点击选择 JBuilder 菜单 Run 或工具列中绿色三角形符号执行此 Web 应用程序。JBuilder 将于集成开发环境内激活 Tomcat4.0Server，我们可以通过 JBuilder 内置之 Web 浏览器进行测试，或于 MicrosoftInternetExplorer 网址字段中键入：

`http://localhost:8080/demo/input.jsp`

以进行测试。在 Web 浏览器中我们键入 AAA 与 BBB 按下 Submit 按钮，得到预期之运算结果 AAABBB。



倘若将第一个字段改以繁体中文键入，却发现产生乱码，字符串连接结果也无法正确辨认。StrutsFramework 并无中文兼容性上之问题，但是 Tomcat4.0 在使用 JSP 处理内含中文请求时，并需设定请求编码。繁体中文环

境常用编码为 Big5 与 MS950 , MS950 编码包括 Big5 编码 , 并额外增加暮、恒、里、妆、嫫、锈、墙等七字。

例如 , 在 Microsoft Windows 平台下 , 欲使 Tomcat4.0 正确处理繁体中文 , 通常在 JSP 起始处加上 :

```
<
%@pagecontentType="text/html;charset=MS950"% >

< %request.setCharacterEncoding("MS950");% >
```

以设定网页内容编码与请求编码为 MS950 编码 ; 类似地 , 对于简体中文 ( 常用编码为 GB2312 ) , 以上的代码应为 :

```
<
%@pagecontentType="text/html;charset=GB2312"% >

< %request.setCharacterEncoding("GB2312");% >
```

但是 StrutsFramework 中实际上是由 ActionServlet 收到 Web 浏览器端请求 , 再转交特定之 Action 类加以处理 , 因此直接于 JSP 中设定请求编码并无效果。

### 三、处理中文编码问题 ( 以繁体中文为例 )

对于处理 StrutsFramework 中文编码问题有多种方式 , 在此我们继承 ActionServlet 类并覆盖 process 方法 , 于子类 process 方法中设定请求编码。由此展现 ActionServlet 扮演前端 Controller 之特质。

首先，点击选择 JBuilder 主菜单 File|NewClasses 准备产生一新的类。并继承自 org.apache.struts.action.ActionServlet 并将预设的类名称由 Untitled1 改为 myActionServlet。按下 OK 按钮后即产生程序代码。点击选择 JBuilder 主菜单 Wizards|OverrideMethods... 激活覆盖方法精灵，覆盖方法精灵会自动列出父类 ActionServlet 所有可供覆盖之方法。我们可于 Availablemethods 清单方块中选取 process(HttpServletRequestrequest,HttpServletResponse response)方法，完成后按下 OK 按钮。

此时我们可修改覆盖后 process 方法程序代码如下：

```
request.setCharacterEncoding("MS950");
```

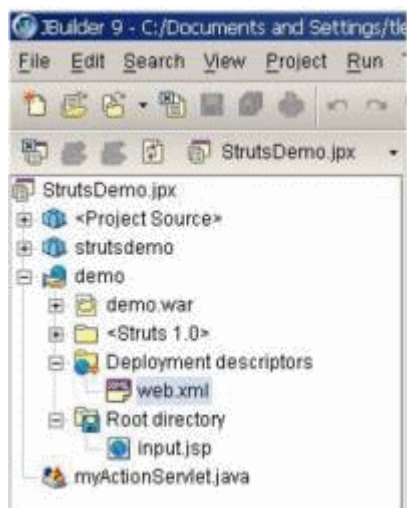
```
super.process(request,response);
```

备注：此处以繁体中文为例，在中国国内一般我们用的是简体中文，因此可以将上述代码在简体中文环境下编写为：

```
request.setCharacterEncoding("GB2312");
```

```
super.process(request,response);
```

首行程序代码将 process 传入之参数 request 设定为 MS950 编码，其余行为同父类。接下来我们将要修改 web.xml 部署描述档，以指定 Servlet 之进入点，首先点击选择左上方项目窗格中的 web.xml。

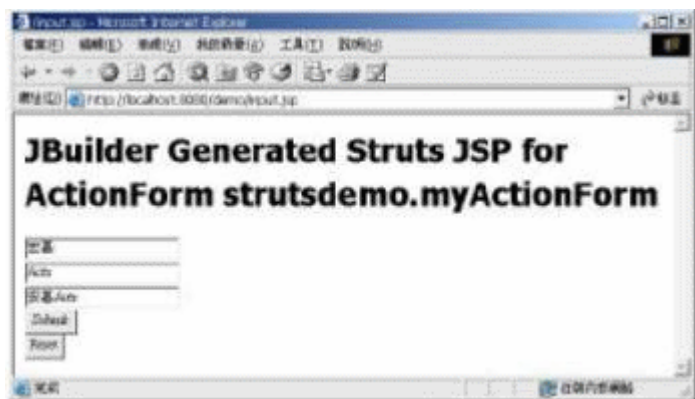


此时，左下方结构窗格将出现 web.xml 内 XML 标记之树状结构。我们可以点击选择 Servlets 内的 Action 项目。

此时，便可运用右方内容窗格之图形使用者接口编辑器修改 Servlet 进入点，目前预设之进入点是 StrutsFramework 的 ActionServlet。

我们可以按下 Servletclass 项目右方[...]按钮选取新的进入点替换预设之 ActionServlet，也就是之我们撰写之 strutsdemo.myActionServlet。倘若我们非常熟悉 web.xml 结构，同样地可以直接点击选择内容窗格内的 Sources 页次，直接以文字编辑器修改 < servlet-class > 标记的内容。

修改完毕后，内容窗格内应如下图：



完成后重新编译此项目并执行之。再次测试后我们将发现繁体中文已经可以正确处理。

## Struts 框架之构建 Model 组件

### 一、概述

Model 就是在对用户请求的整个控制过程中，真正处理用户请求并保存处理结果的对象，在整个过程中，我们一般利用 JavaBean 来把一些信息保存起来以便在各个对象之间传递。因为在框架中，Model 对象是真正处理商业逻辑功能的对象，因此也就是框架中应用需求实现相关性最大的部分。在 Struts 的实现里，Model 的具体表现形式就是 ActionForm 对象和与其对应的 Action 对象了。对用户提交表单的数据进行校验，甚至对数据进行预处理都能在 ActionForm 中完成。通常的应用中，一般是一个 Model 对象和一个请求页面对应的关系，但也可以一个 Model 对象对应多个页面请求。如果