Introduction to Struts 2.x

Agenda

1 **Introduction to Struts2.x**

Introduction to Struts2.x

Objectives

At the end of this module, you will be able to :

- Clearly understand the advantages of using the Struts 2.x framework for building web applications
- Understand the usage of the Model, View and Controller components of Struts 2.x framework
- Build web applications that use the features of Struts 2.x framework

Demo

- In this section we would try and understand how the Struts framework works by building a simple application.

Problem Statement:

The following Struts Application provides a textbox where the user is asked to enter a number from 1 to 10.

When the submit button is clicked the application displays the number that is entered by the user if it is within the range

If the number is not within the range, then an appropriate error message will be displayed to the user

The Prerequisites

- Though just a text editor and the console can be used to create Struts applications, yet the whole process tends to become far simpler through the usage of an IDE like Eclipse or NetBeans. We would use Eclipse IDE for this module

- Create a Dynamic Web Project in Eclipse IDE

- Add the jar files of the struts library to the project's WEB-INF/lib folder

- Create a file named struts.xml in the WEB-INF folder

What is Struts?

- Implementation of the MVC pattern done by Apache Software Foundation
- Web application development framework
- Based on Java Technologies like Servlets, JSP, Java Beans
- Present in two major flavors:
  - Version 1.x
  - Version 2.x
- Provides rich libraries that automate several tasks involved in developing web applications like AJAX, validation etc.

Struts – History

- Developed by Craig McClanaghan
- Donated to the Apache Software Foundation in the year 2000
- Ever since had been used in numerous projects across the world and has been accepted as the web development framework of choice
- Early 2005, WebWork project spun off from the Struts framework
- Webwork and Struts merge together in Feb 2007 and reemerged as struts 2
- Current version 2.3.1

The Struts way of doing things

- Struts is a Java based MVC implementation

- Java technologies have been used in realizing the model, the view and the controller

- These components are implemented through the following:

  - Model– Action Classes(Simple classes also known as POJOs)
  - View – JSP Pages
  - Controller – Servlet Filter

Model – Action classes

- Models in Struts are normal Java classes
- These classes usually have properties to hold data. For instance the data posted by a form is held in an Action class via the member variables
- These classes also have methods that contain the business logic of the application like validation, saving data to database or retrieve data from the database, perform some calculation from the perspective of the application etc.
- These methods may themselves do the task or call methods present in other classes, just to introduce modularity
- The methods return a string that represents the result of the activity performed by the method

Building an action class

- Action classes are simple Java classes. So they need not extend or implement any class or interface as such
- There are classes and interfaces however in the Struts library that make the task of creating action classes a little simpler.
- Often action classes may implement

  Action interface

  or extend

  ActionSupport class(which in turn implements the Action interface and a few more)
- Implementing the Action interface or extending the ActionSupport class provides utility functions for doing stock activities like validation etc.

Building the Action class (Contd.).

- In our example, there is only one data we need to work with
- Let us call this attribute as number
- Let us create an Action Class which encapsulates the attribute called number
- We are going to create an action class called ‘NumberAction’
- This class can be called as POJO (Plain Old Java Object) as it doesn’t extend from any class nor implements any interface

The Action class

```
package com.wipro.action;

public class NumberAction {

int number;

public int getNumber() {
return number;
}

public void setNumber(int number) {
this.number = number;
}

public String execute()
{
if ((number>=1)&&(number<=10))
return "success";
else
return "failure";
}
}
```

Getter Setter methods

Action method

Building the View

- Views are one of the most beautiful aspects of Struts framework
- Views are JSP pages
- Views may be forms or reports
- Struts provides a rich set of tags(struts-tags) that make creating forms and incorporating validation into them easier than plain JSP
- So lets go ahead and create a file – Number.jsp in our project

Building the View (Contd.).

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib uri="/struts-tags" prefix="s" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-
1">
<title>Insert title here</title>
</head>
<body>
<s:form action="firstaction">
<s:textfield label="Enter any number from 1 to 10"
name="number"></s:textfield>
<s:submit label="Submit"></s:submit>
</s:form>
</body>
</html>
```

Struts tags

Action name

Building the Controller

- Finally comes the controller
- Well very less often do we build controllers. The struts libraries provide us with a controller that does almost all that needs to be done
- The controller is a Servlet filter by the name : FilterDispatcher
- All that we need to do is configure the web app in such a manner that, all requests are routed through the FilterDispatcher
- The FilterDispatcher is configured using an xml configuration file that goes by the name struts.xml. Its contained in WEB-INF
- Primarily it contains the various possible outcomes of actions and the targets the outcomes need to be mapped to

Configuring the Controller

- In order to configure the controller add the following code to the web.xml file.

```
<web-app>
  <filter>
    <filter-name>Struts2Filter</filter-name>
    <filter-class>
    org.apache.struts2.dispatcher.FilterDispatcher
    </filter-class>
  </filter>
  <filter-mapping>
    <filter-name>Struts2Filter</filter-name>
    <url-pattern>/*</url-pattern>
  </filter-mapping>
</web-app>
```

- This would make sure all requests that are received by the web server are routed through the filter(Controller)

The Struts configuration file

- We are almost done but there is one last thing that needs to be done i.e. wiring the action class up to the rest of the application

- Forms can only submit data to JSP pages or servlets. In case we want the data to be submitted to an Action class, we need to route it through the FilterDispatcher

- The FilterDispatcher looks up the struts configuration file and decides which Action class has to be instantiated and whose method has to be called

- Configuration information related to Action classes are stored in the struts.xml file

- This file should be present in the same place where other classes are stored, i.e. /WEB-INF/classes.

The struts configuration file (Contd.).

- Type the following code into the struts.xml file:

<action name=*"firstaction"*
class="com.wipro.action.NumberAction" method="execute">
<result name=*"success">Success.jsp</result>*
<result name=*"failure">Failure.jsp</result>*
</action>

- The above code snippet indicates that when “firstaction” method is called, the control will go to the “execute” method found in “NumberAction” class.

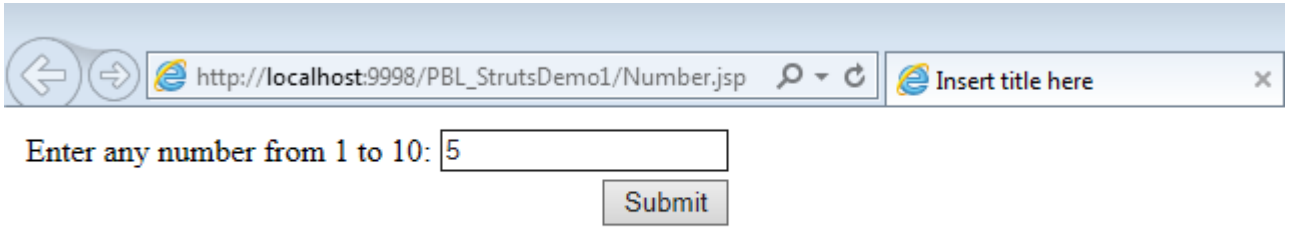
- If the “execute” method returns “success”, the control will go to “Success.jsp”

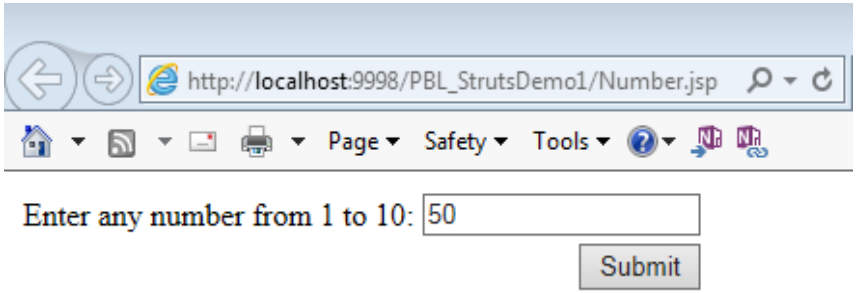
- If the “execute” method returns “failure”, the control will go to “Failure.jsp”

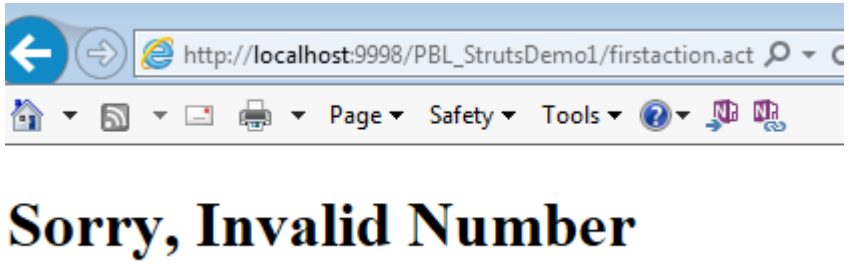
Deploy and Run

- Finally the application can be deployed an executed
- Some error scenarios:
  - Tomcat startup errors may be due to:
    - Wrong data in struts.xml or web.xml
    - Incompatible jar files in the lib folder(Please use the versions specified)
  - Resource not found errors may be again due to the same reasons. So please check web.xml and struts.xml files

The Form







Quiz Time

1. The role of controller in strust2 applications is performed by a _____ .

2. The _____ method of the model is called by default by the controller.

3. Configuration information in struts 2 applications is maintained in the _____ file.

4. Models in struts 2 applications are also known as _____ classes.

Summary

In this module, you were able to :

- Clearly understand the advantages of using the Struts 2.x framework for building web applications
- Understand the usage of the Model, View and Controller components of Struts 2.x framework
- Build web applications that use the features of Struts 2.x framework

Thank You