

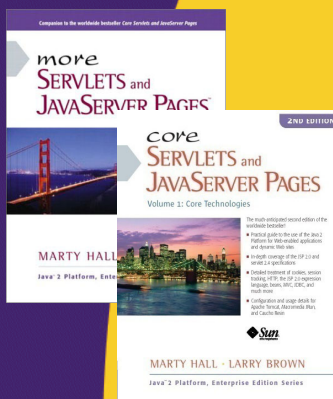


Including Files and Applets in JSP Pages

Originals of Slides and Source Code for Examples:
<http://courses.coreservlets.com/Course-Materials/csajsp2.html>

Customized Java EE Training: <http://courses.coreservlets.com/>
Servlets, JSP, JSF 2.0, Struts, Ajax, GWT 2.0, Spring, Hibernate, SOAP & RESTful Web Services, Java 6.
Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

2



**For live Java EE training, please see training courses
at <http://courses.coreservlets.com/>.**

Servlets, JSP, Struts, JSF 1.x, JSF 2.0, Ajax (with jQuery, Dojo, Prototype, Ext-JS, Google Closure, etc.), GWT 2.0 (with GXT), Java 5, Java 6, SOAP-based and RESTful Web Services, Spring, Hibernate/JPA, and customized combinations of topics.



Taught by the author of *Core Servlets and JSP*, *More Servlets and JSP*, and this tutorial. Available at public venues, or customized versions can be held on-site at your organization. Contact hall@coreservlets.com for details.

Agenda

- **<jsp:include page="..." />**
 - Using `jsp:include` to include pages at request time
- **<%@ include file="..." %>**
 - Using `<%@ include ... %>` (the include directive) to include files at page translation time
- **Usage**
 - Understanding why `jsp:include` is usually better than the include directive
- **Applets**
 - Using `jsp:plugin` to include applets for the Java Plug-in (rare!)

4

Including Pages at Request Time: `jsp:include`

- **Format**
 - `<jsp:include page="Relative address" />`
- **Purpose**
 - To reuse JSP, HTML, or plain text content
 - To permit updates to the included content without changing the main JSP page(s)
- **Notes**
 - JSP content cannot affect main page: only *output* of included JSP page is used
 - Don't forget that trailing slash
 - Relative URLs that starts with slashes are interpreted relative to the Web app, not relative to the server root.
 - You are permitted to include files from WEB-INF

5

jsp:include Example: A News Headline Page (Main Page)

```
...
<BODY>
<TABLE BORDER=5 ALIGN="CENTER">
  <TR><TH CLASS="TITLE">
    What's New at JspNews.com</TABLE>
<P>
Here is a summary of our three
most recent news stories:
<OL>
  <LI><jsp:include page="/WEB-INF/includes/Item1.jsp" />
  <LI><jsp:include page="/WEB-INF/includes/Item2.jsp" />
  <LI><jsp:include page="/WEB-INF/includes/Item3.jsp" />
</OL>
</BODY></HTML>
```

6

A News Headline Page, Continued (First Included Page)

```
<B>Bill Gates acts humble.</B> In a startling
and unexpected development, Microsoft big wig
Bill Gates put on an open act of humility
yesterday.
<A HREF="http://www.microsoft.com/Never.html">
More details...</A>
```

- Note that the page is *not* a complete HTML document; it has only the tags appropriate to the place that it will be inserted

7

A News Headline Page: Result



8

The jsp:param Element: Augmenting Request Parameters

- **Code**

```
<jsp:include page="/fragments/StandardHeading.jsp">  
  <jsp:param name="bgColor" value="YELLOW" />  
</jsp:include>
```
- **URL**
 - http://host/path/MainPage.jsp?fgColor=RED
- **Main page**
 - fgColor: RED
 - bgColor: null
 - Regardless of whether you check before or after inclusion
- **Included page**
 - fgColor: RED
 - bgColor: YELLOW

9

Including Files at Page Translation Time: `<%@ include ... %>`

- **Format**
 - `<%@ include file="Relative address" %>`
- **Purpose**
 - To reuse JSP content in multiple pages, *where JSP content affects main page*
- **Notes**
 - Servers are not required to detect changes to the included file, and in practice they don't.
 - Thus, you need to change the JSP files whenever the included file changes.
 - You can use OS-specific mechanisms such as the Unix “touch” command, or
 - `<%-- Navbar.jsp modified 4/1/09 --%>`
`<%@ include file="Navbar.jsp" %>`

10

`jsp:include` vs. `<%@ include ...>`

	<code>jsp:include</code>	<code><%@ include ...%></code>
Basic syntax	<code><jsp:include page="..." /></code>	<code><%@ include file="..." %></code>
When inclusion occurs	Request time	Page translation time
What is included	Output of page	Contents of file
Number of resulting servlets	Two	One
Can included page set response headers that affect the main page?	No	Yes
Can included page define fields or methods that main page uses?	No	Yes
Does main page need to be updated when included page changes?	No	Yes

11

Which Should You Use?

- **Use jsp:include whenever possible**
 - Changes to included page do not require any manual updates
 - Speed difference between jsp:include and the include directive (@include) is insignificant
- **The include directive (<%@ include ...%>) has additional power, however**
 - Main page
 - <%! int **accessCount** = 0; %>
 - Included page
 - <%@ include file="snippet.jsp" %>
 - <%= **accessCount++** %>

12

Include Directive Example: Reusable Footers

```
<%@ page import="java.util.Date" %>
<!-- The following become fields in each servlet that
      results from a JSP page that includes this file. --%>
<%!
private int accessCount = 0;
private Date accessDate = new Date();
private String accessHost = "<I>No previous access</I>";
%>
<P>
<HR>
This page &copy; 2008
<A HREF="http://www.my-company.com/">my-company.com</A>.
This page has been accessed <%= ++accessCount %>
times since server reboot. It was most recently
accessed from
<%= accessHost %> at <%= accessDate %>.
<% accessHost = request.getRemoteHost(); %>
<% accessDate = new Date(); %>
```

13

Reusing Footers: Typical Main Page

```
...
<BODY>
<TABLE BORDER=5 ALIGN="CENTER">
  <TR><TH CLASS="TITLE">
    Some Random Page</TABLE>
<P>
Information about our products and services.
<P>
Blah, blah, blah.
<P>
Yadda, yadda, yadda.
<%@ include file="/WEB-INF/includes/ContactSection.jsp" %>
</BODY></HTML>
```

14

Reusing Footers: Result



15

Understanding jsp:include vs. `<%@ include ... %>`

- Footer defined the `accessCount` field (instance variable)
- If main pages used `accessCount`, they would have to use `@include`
 - Otherwise `accessCount` would be undefined
- In this example, the main page did *not* use `accessCount`
 - So why did we use `@include`?

16

© 2010 Marty Hall



Applets and jsp:plugin

Customized Java EE Training: <http://courses.coreservlets.com/>

Servlets, JSP, JSF 2.0, Struts, Ajax, GWT 2.0, Spring, Hibernate, SOAP & RESTful Web Services, Java 6.

Developed and taught by well-known author and developer. At public venues or onsite at *your* location.

17

Why Applets?

- **When UI requirements demand it**
 - When HTML (even with Ajax) cannot support GUI
- **Major drawbacks**
 - Many browsers lack Java support
 - So, your Web page will have severely limited audience
 - Applet programming relatively tedious
- **Alternatives**
 - Consider Flash or even Silverlight instead
 - The fact that you are using Java on the server should not make you any more likely to use Java on the client!
 - `jsp:include` is very important part of JSP. `jsp:plugin` is not
 - Skip this entire tutorial section if you do not plan on using applets

18

Options for Deploying Applets

- **Support for very old browsers**
 - Develop the applets with JDK 1.1 or even 1.02
 - Works with almost any browser that has Java installed
 - Uses the simple `APPLET` tag
- **Support for moderately old browsers**
 - Have users install any version of the Java 2 Plug-in, then use Java 2 for the applets.
 - Works with almost any browser
 - Uses ugly `OBJECT` and `EMBED` tags
 - This second option simplified by the `jsp:plugin` tag
- **Support for recent browsers**
 - Have users install version 5 or 6 of the Java Runtime Environment (JRE), then use JDK 1.5/1.6 for the applets.
 - Requires IE 5.5, Netscape 6, or Firefox 2 or later
 - Uses the simple `APPLET` tag

19

Using jsp:plugin

- **Simple APPLET-like tag**
 - Expands into the real OBJECT and EMBED tags
- **APPLET Tag**
 - `<APPLET CODE="MyApplet.class" WIDTH=475 HEIGHT=350>`
`</APPLET>`
- **Equivalent jsp:plugin**
 - `<jsp:plugin type="applet" code="MyApplet.class" width="475" height="350">`
`</jsp:plugin>`
- **Reminder**
 - JSP element and attribute names are case sensitive
 - All attribute values must be in single or double quotes
 - This is like XML but unlike HTML

20

jsp:plugin: Source Code

```
<jsp:plugin type="applet"
            code="SomeApplet.class"
            width="300" height="200">
</jsp:plugin>
```

21

jsp:plugin: Resultant HTML

```
<object classid=
    "clsid:8AD9C840-044E-11D1-B3E9-00805F499D93"
    width="300" height="200"
    codebase="http://java.sun.com/products/plugin/1.2.
2/jinst all-1_2_2-win.cab#Version=1,2,2,0">
    <param name="java_code" value="SomeApplet.class">
    <param name="type" value="application/x-java-applet;">
    <COMMENT>
    <embed type="application/x-java-applet;" width="300"
        height="200"
        pluginspage="http://java.sun.com/products/plugin/"
        java_code="SomeApplet.class"
    >
    <noembed>
    </COMMENT>
    </noembed></embed>
</object>
```

22

jsp:plugin: Example (JSP Code)

```
...
<BODY>
<CENTER>
<TABLE BORDER=5>
    <TR><TH CLASS="TITLE">
        Using jsp:plugin</TH></TR></TABLE>
<P>
    <jsp:plugin type="applet"
        code="PluginApplet.class"
        width="370" height="420">
    </jsp:plugin>
</CENTER></BODY></HTML>
```

23

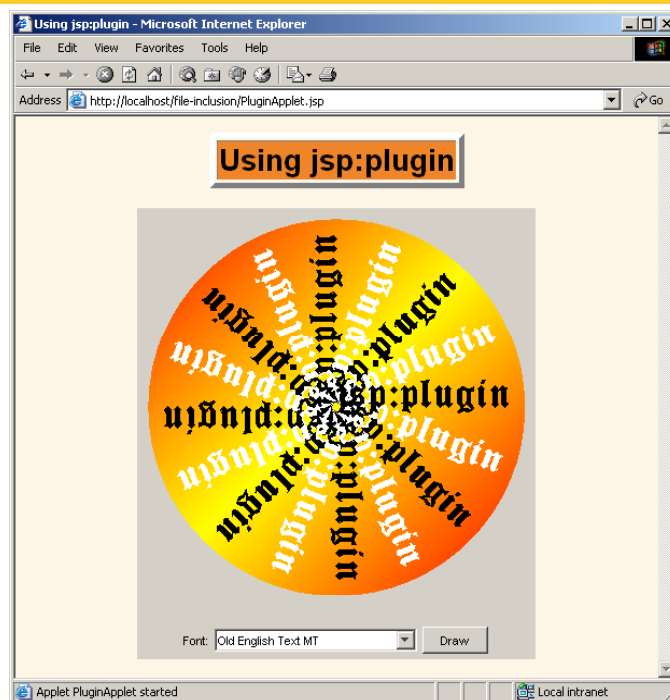
jsp:plugin: Example (Java Code)

```
import javax.swing.*;  
  
/** An applet that uses Swing and Java 2D  
 * and thus requires the Java Plug-in.  
 */  
  
public class PluginApplet extends JApplet {  
    public void init() {  
        WindowUtilities.setNativeLookAndFeel();  
        setContentPane(new JPanel());  
    }  
}
```

- Where are .class files installed?

24

jsp:plugin: Example (Result)



25

Attributes of the jsp:plugin Element

- **type**
 - For applets, this should be "applet".
Use "bean" to embed JavaBeans elements in Web pages.
- **code**
 - Used identically to CODE attribute of APPLET, specifying the top-level applet class file
- **width, height**
 - Used identically to WIDTH, HEIGHT in APPLET
- **codebase**
 - Used identically to CODEBASE attribute of APPLET
- **align**
 - Used identically to ALIGN in APPLET and IMG

26

Attributes of the jsp:plugin Element (Cont.)

- **hspace, vspace**
 - Used identically to HSPACE, VSPACE in APPLET,
- **archive**
 - Used identically to ARCHIVE attribute of APPLET, specifying a JAR file from which classes and images should be loaded
- **name**
 - Used identically to NAME attribute of APPLET, specifying a name to use for inter-applet communication or for identifying applet to scripting languages like JavaScript.
- **title**
 - Used identically to rarely used TITLE attribute

27

Attributes of the jsp:plugin Element (Cont.)

- **jreversion**
 - Identifies version of the Java Runtime Environment (JRE) that is required. Default is 1.2.
- **iepluginurl**
 - Designates a URL from which plug-in for Internet Explorer can be downloaded. Users who don't already have the plug-in installed will be prompted to download it from this location. Default value will direct user to Sun site, but for intranet use you might want to direct user to a local copy.
- **nspluginurl**
 - Designates a URL from which plug-in for Netscape can be downloaded. Default value will direct user to Sun site, but for intranet use you might want local copy.

28

The jsp:param and jsp:params Elements

- **PARAM Tags**
 - ```
<APPLET CODE="MyApplet.class"
 WIDTH=475 HEIGHT=350>
 <PARAM NAME="PARAM1" VALUE="VALUE1">
 <PARAM NAME="PARAM2" VALUE="VALUE2">
</APPLET>
```
- **Equivalent jsp:param**
  - ```
<jsp:plugin type="applet"
            code="MyApplet.class"
            width="475" height="350">
    <jsp:params>
        <jsp:param name="PARAM1" value="VALUE1" />
        <jsp:param name="PARAM2" value="VALUE2" />
    </jsp:params>
</jsp:plugin>
```

29

The jsp:fallback Element

- **APPLET Tag**

- `<APPLET CODE="MyApplet.class" WIDTH=475 HEIGHT=350>`
`Error: this example requires Java.`
`</APPLET>`

- **Equivalent jsp:plugin with jsp:fallback**

- `<jsp:plugin type="applet" code="MyApplet.class" width="475" height="350">`
`<jsp:fallback>`
`Error: this example requires Java.`
`</jsp:fallback>`
`</jsp:plugin>`

30

Summary

- **<jsp:include page="Relative URL" />**

- Output of URL inserted into JSP page at request time
 - Cannot contain JSP content that affects entire page
 - Changes to included file do not necessitate changes to pages that use it

- **<%@ include file="Relative URL" %>**

- File gets inserted into JSP page prior to page translation
 - Thus, file can contain JSP content that affects entire page (e.g., import statements, declarations)
 - Changes to included file require you to manually update pages that use it

- **<jsp:plugin ...>**

- Simplifies writing applets that use the Java Plug-In

31



Questions?

Customized Java EE Training: <http://courses.coreservlets.com/>

Servlets, JSP, JSF 2.0, Struts, Ajax, GWT 2.0, Spring, Hibernate, SOAP & RESTful Web Services, Java 6.

Developed and taught by well-known author and developer. At public venues or onsite at *your* location.