

Core Servlets and JavaServer Pages / 2e

Volume 1: Core Technologies

Marty Hall • Larry Brown

**Including Files
and Applets
in JSP Pages**

Agenda

- Using `jsp:include` to include pages at request time
- Using `<%@ include ... %>` (the include directive) to include files at page translation time
- Understanding why `jsp:include` is usually better than the include directive
- Using `jsp:plugin` to include applets for the Java Plug-in

Including Files at Request Time: `jsp:include`

- **Format**

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

- **Purpose**

- To enable you to insert the contents of an HTML page, plain text document, output of a JSP page or the output of a servlet
 - 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

jsp:include Relative URLs

If the JSP page is located in

`http://host/headlines/sports/table-tennis.jsp`

```
<jsp:include page="bios/cheng-yinghua.jsp" />
```

the application looks in the sports/bios subdirectory

```
<jsp:include page="/templates/footer.jsp" />
```

the application looks in the templates subdirectory of the headlines application (and not of the server root)

- **URLs that begin with slashes are interpreted**
 - relative to the current Web application whenever the server handles them
 - relative to the server root only when the client (browser) handles them
 - Recommended to put included pages in WEB-INF as it will prevent the client from accidentally accessing them

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/Item1.html" />
  <LI><jsp:include page="/WEB-INF/Item2.html" />
  <LI><jsp:include page="/WEB-INF/Item3.html" />
</OL>
</BODY></HTML>
```

A News Headline Page, Continued (First Included Page)

[/WEB-INF/Item1.html](#)

```
<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

A News Headline Page: Result



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

- **Format**

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

- **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 update the modification dates of each JSP page that uses the file.
 - Some OSs have commands that update the modification date without your actually editing the file (UNIX `touch` command)

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**
 - It can contain JSP code that affects the main page
 - `snippet.jsp`
 - `<%! int accessCount = 0; %>`
 - Main page
 - `<%@ include file="snippet.jsp" %>`
 - `<%= accessCount++ %>`

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

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

Include Directive Example: Reusable Footers

- You have a JSP page that generates an HTML snippet containing a small footer that includes access counts and information about the most recent accesses to the current pages (see `ContactSession.jsp` on next slide)
- If several pages want to have footers of that type, put the footer in `WEB-INF` and each page would use it via

```
<%@ include file="/WEB-INF/ContactSection.jsp" %>
```

(see `SomeRandompage.jsp` slide)

Include Directive Example: Reusable Footers

ContactSession.jsp

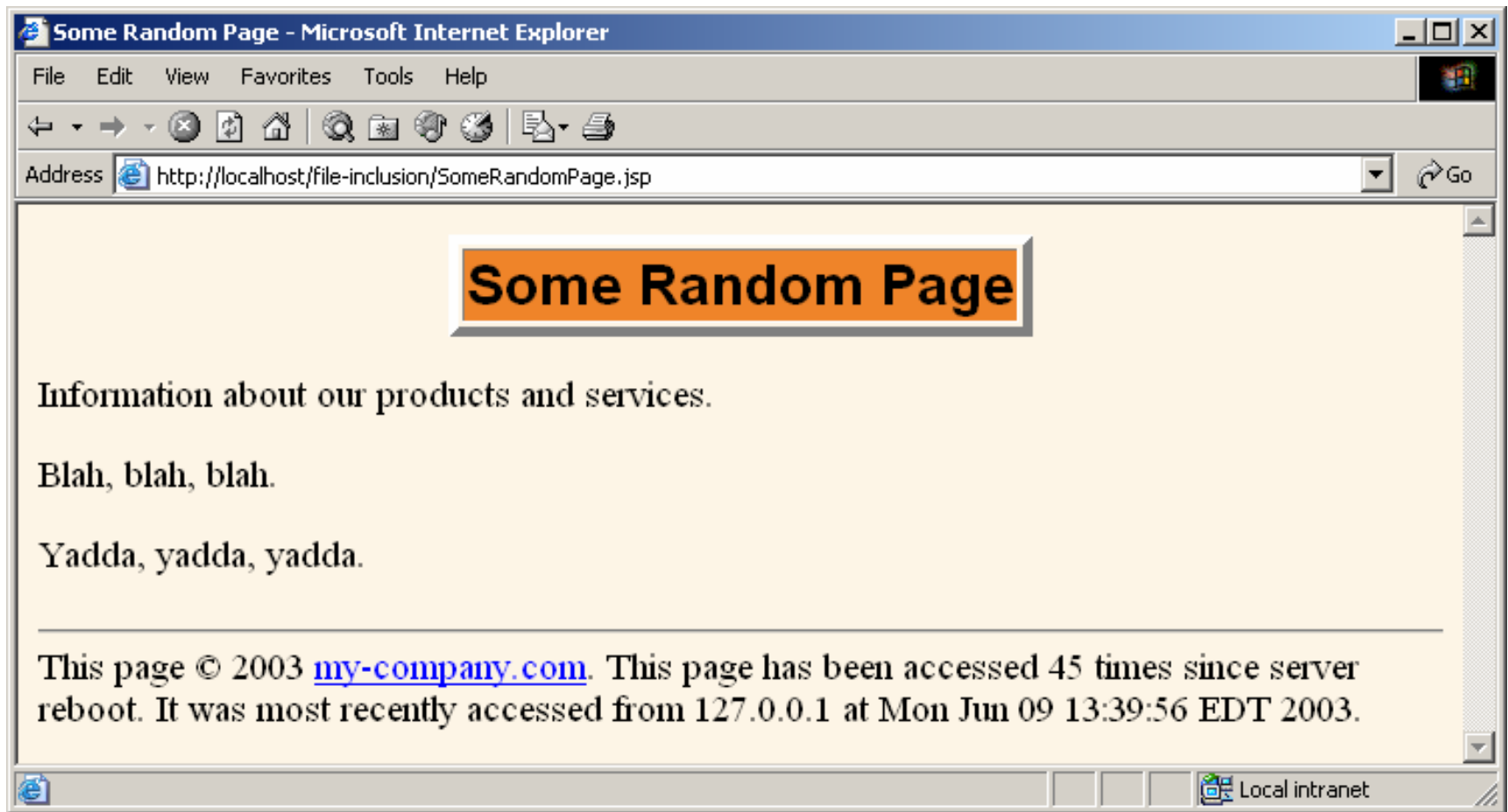
```
<%@ page import="java.util.Date" %>
<!-- The following become fields in each servlet that
      results from a JSP page that includes this file. -->
<%!
private int accessCount = 0;
private Date accessDate = new Date();
private String accessHost = "<I>No previous access</I>";
%>
<P>
<HR>
This page &copy; 2003
<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(); %>
```

Reusing Footers: Typical Main Page

SomeRandomPage.jsp

```
...
<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/ContactSection.jsp" %>
</BODY></HTML>
```

Reusing Footers: Result



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`?**
 - `jsp:include` would give the same access count across all pages and not a separate count per page
 - Don't want `ContactSession.jsp` to be its own servlet, rather have it included in each separate servlet that results from a JSP page that uses `ContactSession.jsp`

Options for Deploying Applets

- **Develop the applets with JDK 1.1 or even 1.02 (to support *really* old browsers).**
 - Works with almost any browser
 - Uses the simple APPLET tag
 - Lack of support for Java 2 Platform restricts applications
- **Have users install version 1.4 of the Java Runtime Environment (JRE), then use JDK 1.4 for the applets.**
 - Requires IE 5 or later or Netscape 6 or later
 - Uses the simple APPLET tag
- **Have users install any version of the Java 2 Plugin, then use Java 2 for the applets.**
 - Works with almost any browser
 - Uses ugly OBJECT and EMBED tags
 - This third option simplified by the `jsp:plugin` tag

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

jsp:plugin: Source Code

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

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>
```

jsp:plugin: Example (JSP Code)

```
...  
<BODY>  
<CENTER>  
<TABLE BORDER=5>  
    <TR><TH CLASS="TITLE">  
        Using jsp:plugin</TABLE>  
<P>  
// 4 main attributes for a plugin  
<jsp:plugin type="applet"  
            code="PluginApplet.class"  
            width="370" height="420">  
</jsp:plugin>  
</CENTER></BODY></HTML>
```

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());  
    }  
}
```

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

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

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.

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>`

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