

Creating Custom JSP Tag Libraries: Advanced Topics

Core Servlets & JSP book: www.coreservlets.com
 More Servlets & JSP book: www.moreservlets.com
 Servlet and JSP Training Courses: courses.coreservlets.com

Slides © Marty Hall, <http://www.moreservlets.com>, book © Sun Microsystems Press

Agenda

- Manipulating the tag body
- Looping tags
 - JSP 1.1
 - JSP 1.2
- Nested tags
- Aside: parsing XML with SAX 2.0
- Using SAX and TagLibraryValidator to validate tag library syntax

www.moreservlets.com

Uses of JSP Constructs

Simple Application

↓

Complex Application

- Scripting elements calling servlet code directly
- Scripting elements calling servlet code indirectly (by means of utility classes)
- Beans
- Custom tags
- Servlet/JSP combo (MVC), with beans and possibly custom tags

www.moreservlets.com

Manipulating the Tag Body

- No tag body
 - `<prefix:tagName />`
 - `<prefix:tagName att1="val1" ... />`
- Previous uses of tag body
 - `<prefix:tagName>JSP Content</prefix:tagName>`
 - `<prefix:tagName att1="val1" ... />`
 JSP Content
 `</prefix:tagName>`
 - Content inserted verbatim
- Now
 - Same JSP syntax for using a tag body
 - Java code can read/modify/replace tag body

www.moreservlets.com

Manipulating Tag Body: The Tag Handler Class

- Extend `BodyTagSupport`
 - `BodyTagSupport` does not have enough infrastructure to support reading/manipulating the tag body
- New method to override: `doAfterBody`
 - Return `SKIP_BODY` when done
- `getBodyContent` returns object of type `BodyContent` that has three key methods
 - `getEnclosingWriter` -- returns the `JspWriter` being used by `doStartTag` and `doEndTag`
 - `getReader` -- returns a `Reader` that can read tag's body
 - `getString` -- returns a `String` containing entire tag body
- `getBodyContent` is expensive
 - Don't use it if `doStartTag` and `doEndTag` will do

www.moreservlets.com

Manipulating Tag Body: FilterTag.java

```
public class FilterTag extends BodyTagSupport {
    public int doAfterBody() {
        BodyContent body = getBodyContent();
        String filteredBody =
            // The filter method is defined in Section 3.6
            ServletUtilities.filter(body.getString());
        try {
            JspWriter out = body.getEnclosingWriter();
            out.print(filteredBody);
        } catch (IOException ioe) {
            System.out.println("Error in FilterTag: " + ioe);
        }
        // SKIP_BODY means I'm done. If I wanted to evaluate
        // and handle body again, I'd return EVAL_BODY_TAG.
        return(SKIP_BODY);
    }
}
```

www.moreservlets.com

Manipulating Tag Body: The Tag Library Descriptor File

- **No new capabilities needed**

```

<tag>
  <name>filter</name>
  <tagclass>coreservlets.tags.FilterTag</tagclass>
  <info>
    Replaces HTML-specific
    characters in body.
  </info>
</tag>
    
```

www.moreservlets.com

Using the filter Tag

```

...
<% taglib uri="csajsp-taglib.tld" prefix="csajsp" %>
<TABLE BORDER=1 ALIGN="CENTER">
<TR CLASS="COLORED"><TH>Example<TH>Result
<TR>
<TD><PRE><csajsp:filter>
<EM>Some emphasized text.</EM><BR>
<STRONG>Some strongly emphasized text.</STRONG><BR>
<CODE>Some code.</CODE><BR>
...
</csajsp:filter></PRE>
<TR>
<TD>
<EM>Some emphasized text.</EM><BR>
<STRONG>Some strongly emphasized text.</STRONG><BR>
<CODE>Some code.</CODE><BR>
...
</TABLE>
    
```

www.moreservlets.com

Using the filter Tag: Results

www.moreservlets.com

Looping Tags (JSP 1.1)

- **Return EVAL_BODY_TAG from doAfterBody until last repetition**
 - Return EVAL_BODY_TAG: keep going
 - Return SKIP_BODY: done
- **Use the TLD file to specify that JSP expressions are permitted as attribute values**
 - <rtexprvalue>>true</rtexprvalue>

www.moreservlets.com

Simple Looping Tag (JSP 1.1)

```

public class RepeatTag1 extends BodyTagSupport {
  private int reps;

  public void setReps(String repeats) {
    try {
      reps = Integer.parseInt(repeats);
    } catch (NumberFormatException nfe) {
      reps = 1;
    }
  }
}
    
```

www.moreservlets.com

Simple Looping Tag (JSP 1.1) (Continued)

```

public int doAfterBody() {
  if (reps-- >= 1) {
    BodyContent body = getBodyContent();
    try {
      JspWriter out = body.getEnclosingWriter();
      out.println(body.getString());
      body.clearBody(); // Clear for next evaluation
    } catch (IOException ioe) {
      System.out.println("Error in RepeatTag1: "
        + ioe);
    }
    return (EVAL_BODY_TAG);
  } else {
    return (SKIP_BODY);
  }
}
    
```

www.moreservlets.com

TLD File for Looping Tag

```

...
<tag>
  <name>repeat1</name>
  <tag-class>moreservlets.tags.RepeatTag1</tag-class>
  <attribute>
    <name>reps</name>
    <required>true</required>
    <!-- rtexprvalue indicates whether attribute
         can be a JSP expression. -->
    <rtexprvalue>true</rtexprvalue>
  </attribute>
</tag>

```

12 Advanced Custom Tags www.moreservlets.com

Looping Tag: Example

```

<%@ taglib uri="/WEB-INF/repeat-taglib.tld"
      prefix="msajsp" %>
<UL>
<msajsp:repeat1
  reps='<%= request.getParameter("repeats") %>\'
  <LI><%= Math.random() %>
</msajsp:repeat1>
</UL>

```

14 Advanced Custom Tags www.moreservlets.com

Looping Tag: Result



16 Advanced Custom Tags www.moreservlets.com

Looping Tags: JSP 1.2

- **Extend TagSupport instead of BodyTagSupport**
 - If you don't actually need to manipulate the tag body, it is wasteful to create BodyContent each time.
 - You have no choice in JSP 1.1.
- **Return EVAL_BODY_INCLUDE from doStartTag**
 - EVAL_BODY_INCLUDE: keep looping
 - SKIP_BODY: done
- **Return EVAL_BODY_AGAIN from doAfterBody for looping**
 - EVAL_BODY_AGAIN: keep looping
 - SKIP_BODY: done

16 Advanced Custom Tags www.moreservlets.com

Looping Tag: JSP 1.2 Version

```

public class RepeatTag2 extends TagSupport {
  private int reps;

  public void setReps(String repeats) {
    try {
      reps = Integer.parseInt(repeats);
    } catch (NumberFormatException nfe) {
      reps = 1;
    }
  }
}

```

17 Advanced Custom Tags www.moreservlets.com

Looping Tag: JSP 1.2 Version (Continued)

```

public int doStartTag() {
  if (reps >= 1) {
    return(EVAL_BODY_INCLUDE);
  } else {
    return(SKIP_BODY);
  }
}

public int doAfterBody() {
  if (reps-- > 1) {
    return(EVAL_BODY_AGAIN);
  } else {
    return(SKIP_BODY);
  }
}

```

18 Advanced Custom Tags www.moreservlets.com

Looping Tag: Result



www.moreservlets.com

Nested Tags

- **Finding enclosing tags**
 - The handler for the inner tag uses findAncestorWithClass to get a reference to the enclosing tag handler
- **Storing data**
 - If an enclosing tag wants to provide data for use by a nested tag, it should provide methods that let the nested tag store results in a field of the outer tag.

www.moreservlets.com

Nested Tags: Template

```

public class OuterTag extends TagSupport {
    public void setSomeValue(SomeClass arg) { ... }
    public SomeClass getSomeValue() { ... }
}

public class FirstInnerTag extends BodyTagSupport {
    public int doStartTag() throws JspTagException {
        OuterTag parent =
            (OuterTag)findAncestorWithClass(this,
                OuterTag.class);

        if (parent == null) {
            throw new JspTagException("nesting error");
        } else {
            parent.setSomeValue(...);
        }
        return(EVAL_BODY_TAG);
    }
    ...
}
    
```

www.moreservlets.com

Nested Tags: Template (Continued)

```

public class SecondInnerTag extends BodyTagSupport {
    public int doStartTag() throws JspTagException {
        OuterTag parent =
            (OuterTag)findAncestorWithClass(this, OuterTag.class);
        if (parent == null) {
            throw new JspTagException("nesting error");
        } else {
            SomeClass value = parent.getSomeValue();
            doSomethingWith(value);
        }
        return(EVAL_BODY_TAG);
    }
    ...
}
    
```

www.moreservlets.com

Nested Tags: the if Tag

```

<%@ taglib uri="csajsp-taglib.tld" prefix="csajsp" %>
<csajsp:if>
  <csajsp:condition>true</csajsp:condition>
  <csajsp:then>Condition was true</csajsp:then>
  <csajsp:else>Condition was false</csajsp:else>
</csajsp:if>
...
Some coin tosses:<BR>
<csajsp:repeat reps="20">
  <csajsp:if>
    <csajsp:condition>
      <%= Math.random() > 0.5 %>
    </csajsp:condition>
    <csajsp:then><B>Heads</B><BR></csajsp:then>
    <csajsp:else><B>Tails</B><BR></csajsp:else>
  </csajsp:if>
</csajsp:repeat>
    
```

www.moreservlets.com

Using the if Tag: Results



www.moreservlets.com

Issue: Semantics of Custom Tag Usage

- System already uses the JSP DTD to verify that the *standard* tags are used properly.
- System will already verify basic custom tag syntax
 - Tags are well formed
 - All tag and attribute names spelled properly
 - Required attributes supplied
 - No undeclared attributes used
- **But, what about deeper issues?**
 - Certain custom tags must be nested in certain patterns
 - A custom tag has two attributes: both must appear or neither must appear.

26 Advanced Custom Tags www.moreservlets.com

Idea: Use Ordinary XML Tools

Original Page	Internal Representation
<pre><DOCTYPE ...> ... <%= Math.random() %> <myTags:doSomething> Blah </myTags:doSomething> </BODY></HTML></pre>	<pre><?xml version="1.0" ?> <jsp:root ...> <jsp:text> <![CDATA[<DOCTYPE...>...]]> </jsp:text> <jsp:expression> Math.random() </jsp:expression> <myTags:doSomething> Blah </myTags:doSomething> <jsp:text> <![CDATA[<BODY>...]]> </jsp:text> </jsp:root></pre>

26 Advanced Custom Tags www.moreservlets.com

Aside: Parsing XML with the Simple API for XML (SAX) -- Setup

- Download a SAX-compliant parser.
 - Parser provides the classes that follow the SAX 2 API. For XML parsers in Java, see http://www.xml.com/pub/rg/Java_Parsers. I use the Apache Xerces-J -- <http://xml.apache.org/xerces-j/>.
- Download the Java API for XML Processing (JAXP).
 - This API provides a small layer on top of SAX that lets you plug in different parsers without making any changes to your basic code. See <http://java.sun.com/xml/>.
- **Tell your development environment and the server about the SAX and JAXP classes.**
 - Tomcat already comes with Xerces, but you need to set your CLASSPATH to include xerces.jar and jaxp.jar.

27 Advanced Custom Tags www.moreservlets.com

Aside: Parsing XML with the Simple API for XML (SAX) – Parsing

- **Tell the system which parser you want.**

```
String jaxpPropertyName =
    "javax.xml.parsers.SAXParserFactory";
if (System.getProperty(jaxpPropertyName) == null) {
    String apacheXercesPropertyValue =
        "org.apache.xerces.jaxp.SAXParserFactoryImpl";
    System.setProperty(jaxpPropertyName,
        apacheXercesPropertyValue);
}
...

```
- **Create a parser instance.**

```
SAXParserFactory factory =
    SAXParserFactory.newInstance();
SAXParser parser = factory.newSAXParser();

```

28 Advanced Custom Tags www.moreservlets.com

Aside: Parsing XML with the Simple API for XML (SAX) – Parsing Continued

- **Create a content handler to respond to parsing events. Subclass DefaultHandler and override:**
 - startDocument, endDocument
 - startElement, endElement
 - characters, ignoreableWhitespace
- **Invoke the parser with the designated content handler.**

```
MyHandler handler = new MyHandler();
InputSource source =
    new InputSource(someInputStream);
parser.parse(source, handler);

```

29 Advanced Custom Tags www.moreservlets.com

Checking Tag Library Syntax with TagLibraryValidator

- **Create a subclass of TagLibraryValidator.**
- **Override the validate method.**

```
public ValidationMessage[] validate(String prefix,
    String uri,
    PageData page) {
    InputStream stream = page.getInputStream();
    // Pass stream to SAX parser; return null if valid
    • Note that the InputStream reads a pure-XML version of the JSP page. E.g. <%= foo %> will be read as <jsp:expression>foo</jsp:expression>.

```
- **Declare the validator in the TLD file.**

```
<taglib>...
<validator>
<validator-class>
    somePackage.SomeValidatorClass
</validator-class>
</validator>
```

30 Advanced Custom Tags www.moreservlets.com

Example: Enforcing Nesting Order

- outerTag cannot be nested
- innerTag can only appear within outerTag
 - Directly or indirectly
- innerTag can be nested arbitrarily

Legal: <pre><test:outerTag> <test:innerTag> <test:innerTag/> </test:innerTag> </test:outerTag> <test:innerTag> <test:innerTag/> </test:innerTag> </test:outerTag> <test:outerTag/></pre>	Illegal: <pre><test:innerTag/></pre> Also Illegal: <pre><test:outerTag> <test:outerTag/> </test:outerTag></pre>
--	--

www.moreservlets.com

Enforcing Nesting Order: SAX Handler

```
public void startElement(String namespaceUri,
    String localName,
    String qualifiedName,
    Attributes attributes)
    throws SAXException {
    String tagName = mainTagName(qualifiedName);
    if (tagName.equals(outerTagName)) {
        if (inOuterTag) {
            throw new SAXException("\nCannot nest "
                + outerTagName);
        }
        inOuterTag = true;
    } else if (tagName.equals(innerTagName)
        && !inOuterTag) {
        throw new SAXException("\n" + innerTagName +
            " can only appear within " +
            outerTagName);
    }
}
```

www.moreservlets.com

Enforcing Nesting Order: SAX Handler (Continued)

```
public void endElement(String namespaceUri,
    String localName,
    String qualifiedName)
    throws SAXException {
    String tagName = mainTagName(qualifiedName);
    if (tagName.equals(outerTagName)) {
        inOuterTag = false;
    }
}
```

www.moreservlets.com

Summary

- Manipulating the tag body**
 - Extend BodyTagSupport, implement doAfterBody
 - Get tag body by calling getBodyContent then getString
 - Return results with getEnclosingWriter
- Looping tags**
 - JSP 1.1: Return EVAL_BODY_TAG from doAfterBody until last repetition
 - JSP 1.2: Return EVAL_BODY_INCLUDE from doStartTag; EVAL_BODY_AGAIN from doAfterBody
- Nested tags**
 - Find outer tag with findAncestorWithClass
- Using TagLibraryValidator**
 - Extend TagLibraryValidator, override validate
 - Get InputStream to read XML representation of page

www.moreservlets.com




Questions?

Core Servlets & JSP book: www.coreservlets.com
 More Servlets & JSP book: www.moreservlets.com
 Servlet and JSP Training Courses: courses.coreservlets.com

Slides © Marty Hall, <http://www.moreservlets.com>, book © Sun Microsystems Press

More Information

- Source code for examples**
 - <http://www.moreservlets.com>
- More Servlets & JSP**
 - <http://www.moreservlets.com>
 - Site includes info on servlet and JSP training courses
- Core Servlets & JSP**
 - Prequel to More Servlets & JSP
 - <http://www.coreservlets.com>
- Servlet home page**
 - <http://java.sun.com/products/servlet/>
- JavaServer Pages home page**
 - <http://java.sun.com/products/jsp/>



www.moreservlets.com