



# Java Application Programming

## JavaServer Pages

2008-11-27

P-O Östberg  
Department of Computing Science, Umeå University




Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

- Static HTML
- Directives
- Scripting Elements
- Actions
- Comments
- Tag Libraries
- Implicit Objects
- Cookies
- Sessions
- JSP Processing
- Client-Side Components

Java Application Programming  
JavaServer Pages

P-O Östberg  
Department of Computing Science, Umeå University




WEB TECHNOLOGIES

## Web Technologies

- ▶ Markup & presentation (HTML, XHTML, CSS etc)
- ▶ Data storage & access (JDBC, XML etc)
- ▶ Network & application protocols (TCP/IP, HTTP etc)
- ▶ Programming & scripting languages (Java, JSP etc)

Java Application Programming  
JavaServer Pages

P-O Östberg  
Department of Computing Science, Umeå University




WEB TECHNOLOGIES

## World Wide Web (WWW)

- ▶ Web clients and servers
- ▶ More than 1 billion web pages on
- ▶ More than 170 million TLDs (May 2008)
- ▶ HTML, graphics, and components
- ▶ Transferred via HTTP (over TCP/IP)

Java Application Programming  
JavaServer Pages

P-O Östberg  
Department of Computing Science, Umeå University




WEB TECHNOLOGIES

## Hypertext Markup Language (HTML)

- ▶ Text-based
- ▶ Content + Markup
- ▶ Interpreted and visualized by browsers
- ▶ Constitutes a small part of the resulting network traffic

Java Application Programming  
JavaServer Pages

P-O Östberg  
Department of Computing Science, Umeå University



WEB TECHNOLOGIES

## Extensible Hypertext Markup Language (XHTML)

- ▶ XML version of HTML
- ▶ Well-formed XML document
- ▶ Can be parsed by any XML tool
- ▶ Treated by browsers as a new version of HTML

Java Application Programming  
JavaServer Pages

P-O Östberg  
Department of Computing Science, Umeå University



## Uniform Resource Locator (URL)

- Today
- Web Technologies
- Server-Side Web Development
- JavaServer Pages (JSP)
- Basic HTML
- Directives
- Scripting Elements
- Actions
- Comments
- Tag Libraries
- Insertion Points
- Custom Tags
- Web Services
- XML
- Web Services
- XML
- Web Services
- XML

<http://www.cs.umu.se:80/kurser/5DV085/HT-08/index.html>

Contains

- ▶ Protocol (http)
- ▶ Address (www.cs.umu.se)
- ▶ Port (80)
- ▶ Path (/kurser/5DV085/HT-08/index.html)



## Web Client

- Today
- Web Technologies
- Server-Side Web Development
- JavaServer Pages (JSP)
- Basic HTML
- Directives
- Scripting Elements
- Actions
- Comments
- Tag Libraries
- Insertion Points
- Custom Tags
- Web Services
- XML
- Web Services
- XML
- Web Services
- XML

- ▶ Requests resources via HTTP
- ▶ Interprets and presents the data retrieved
- ▶ Can be anything that requests data via HTTP
- ▶ Usually a web browser
- ▶ Uses URLs to reference resources



## Web Server

- Today
- Web Technologies
- Server-Side Web Development
- JavaServer Pages (JSP)
- Basic HTML
- Directives
- Scripting Elements
- Actions
- Comments
- Tag Libraries
- Insertion Points
- Custom Tags
- Web Services
- XML
- Web Services
- XML
- Web Services
- XML

- ▶ Serves resources via HTTP
- ▶ Can be anything that serves data via HTTP
- ▶ Usually a dedicated server software
- ▶ May contain several pieces of software working together
- ▶ One server may serve several sites (and vice versa)
- ▶ Large sites often clustered for performance



## Protocols

- Today
- Web Technologies
- Server-Side Web Development
- JavaServer Pages (JSP)
- Basic HTML
- Directives
- Scripting Elements
- Actions
- Comments
- Tag Libraries
- Insertion Points
- Custom Tags
- Web Services
- XML
- Web Services
- XML
- Web Services
- XML

Specifies how information is exchanged in networks

- ▶ In which order
- ▶ In what format
- ▶ Transport level and application level protocols
- ▶ Built in protocol stacks



## Hypertext Transfer Protocol (HTTP)

- Today
- Web Technologies
- Server-Side Web Development
- JavaServer Pages (JSP)
- Basic HTML
- Directives
- Scripting Elements
- Actions
- Comments
- Tag Libraries
- Insertion Points
- Custom Tags
- Web Services
- XML
- Web Services
- XML
- Web Services
- XML

- ▶ Text-based
- ▶ Application-level protocol (mostly used over TCP)
- ▶ Client-driven (requests and responses)
- ▶ Stateless (sessions are stored in cookies or rewritten URLs)
- ▶ Can handle text as well as binary data (encoded as text)



## HTTP Request

- Today
- Web Technologies
- Server-Side Web Development
- JavaServer Pages (JSP)
- Basic HTML
- Directives
- Scripting Elements
- Actions
- Comments
- Tag Libraries
- Insertion Points
- Custom Tags
- Web Services
- XML
- Web Services
- XML
- Web Services
- XML

- ▶ Request line (method + URI + protocol)
- ▶ Headers (request information)
- ▶ Body (optional)

Umeå University

WEB TECHNOLOGIES

## HTTP GET Request Example

---

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Objects  
Annotations  
XML Processing  
Web Services  
Miscellaneous

```
GET /kurser/5DV085/HT-08/test.html HTTP/1.1
Host: www.cs.umu.se
```

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

WEB TECHNOLOGIES

## HTTP Response Example

---

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Objects  
Annotations  
XML Processing  
Web Services  
Miscellaneous

```
HTTP/1.1 200 OK
Date: Tue, 15 May 2007 14:25:27 GMT
Server: Apache/2.0.54 (Unix)
Accept-Ranges: bytes
Content-Length: 50
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=ISO-8859-1
```

```
<html>
<head>
</head>
<body>
test
</body>
```

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

WEB TECHNOLOGIES

## HTTP Request Methods

---

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Objects  
Annotations  
XML Processing  
Web Services  
Miscellaneous

- ▶ **HEAD** - simulate a get request
- ▶ **GET** - retrieve resource
- ▶ **POST** - submit data to resource
- ▶ **PUT** - upload resource
- ▶ **DELETE** - delete resource
- ▶ **TRACE** - echo request
- ▶ **OPTIONS** - query server for supported methods
- ▶ **CONNECT** - create TCP/IP tunnel

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

WEB TECHNOLOGIES

## HTTP Response Status Codes

---

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Objects  
Annotations  
XML Processing  
Web Services  
Miscellaneous

Directs client behavior

- ▶ 100-199 - informational
- ▶ 200-299 - request successful
- ▶ 300-399 - resource unavailable / moved
- ▶ 400-499 - client-side error
- ▶ 500-599 - server-side error

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

SERVER-SIDE WEB DEVELOPMENT

## Server-Side Web Development

---

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Objects  
Annotations  
XML Processing  
Web Services  
Miscellaneous

- ▶ Data access and logic rather than interfaces
- ▶ More technical than HTML/CSS design
- ▶ Several alternative technology platforms exists
- ▶ Dynamic content vs static content
- ▶ Content-driven solutions
- ▶ Focus on web applications
- ▶ Need for a structured programming model
- ▶ Need for a simpler way to do programming

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

SERVER-SIDE WEB DEVELOPMENT

## Web Applications

---

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Objects  
Annotations  
XML Processing  
Web Services  
Miscellaneous

- ▶ Applications with a web interface
- ▶ Ideal for thin-client solutions
- ▶ Suffers from limitations of the web media
- ▶ Usually combines techniques (e.g., JSP + AJAX)
- ▶ Several development frameworks available
- ▶ Clear trend towards generated web interfaces
- ▶ Usually session-oriented
- ▶ Usually deployed in WAR files

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

SERVER-SIDE WEB DEVELOPMENT

## Web Archives (WAR)

- ▶ ZIP-file = compressed archive
- ▶ JAR-file = Java Archive (ZIP file with a manifest)
- ▶ WAR-file = JAR-file with web application information

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

SERVER-SIDE WEB DEVELOPMENT

## Sessions

- ▶ Used to store data for a series of HTTP requests
- ▶ E.g., a shopping cart, user preferences, site history
- ▶ A session identifier is sent with each request
- ▶ The session identifier is used to locate the session
- ▶ Data is stored in the session context

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

SERVER-SIDE WEB DEVELOPMENT

## Three Layer Architectures

1. Interface - web pages
2. Logic - software components (JavaBeans, EJB)
3. Data - databases

- ▶ Clean separation of concerns
- ▶ Scalable
- ▶ Support role-based development cycles
- ▶ Well suited for large sites and business logic integration

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

SERVER-SIDE WEB DEVELOPMENT

## Two Layer Architectures

1. Interface (+ Logic)
2. Data + Logic

- ▶ Requires fewer software components
- ▶ Shorter development cycles for small development teams
- ▶ Better suited for smaller web sites

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

SERVER-SIDE WEB DEVELOPMENT

## Web Servers

- ▶ Serves resources via HTTP
- ▶ Can be anything that serves data via HTTP
- ▶ Usually a dedicated machine running web server software
- ▶ Can contain modules that processes requests

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

SERVER-SIDE WEB DEVELOPMENT

## Request Processing

1. A Connection is established
2. A HTTP request is received
3. The (logical) path in the request is translated
4. The requested resource is identified (via the path)
5. A server module handling that resource is invoked
6. The module processes the request and generates a reply
7. A mime-type is provided and a HTTP response is created
8. The HTTP response is sent (possibly in increments)

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

SERVER-SIDE WEB DEVELOPMENT

## JSP Processing

1. A JSP page is requested
2. Server checks if a Java Servlet for the page exists
3. If no Servlet is found (or newer JSP is detected), the JSP is translated to Java (a Servlet class is created)
4. The Java Servlet is compiled
5. The Java Servlet is invoked and processes the request

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

SERVER-SIDE WEB DEVELOPMENT

## Java Servlets

- ▶ Java classes
- ▶ Implements the Java Servlet API interfaces (predates JSP)
- ▶ Receives a request and generates a response
- ▶ Can be written manually
- ▶ Must be thread-safe
- ▶ Usually generated automatically from JSP
- ▶ Hosted in a Servlet container

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

SERVER-SIDE WEB DEVELOPMENT

## Java Servlets

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

SERVER-SIDE WEB DEVELOPMENT

## The Servlet Lifecycle

1. `init()` - called on Servlet instantiation
2. `service()` - called for each request
3. `destroy()` - called on container shutdown

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

SERVER-SIDE WEB DEVELOPMENT

## The Servlet `service()` method

- ▶ Part of a service-pattern
- ▶ Should not be implemented directly
- ▶ Inherit base class and implement handler methods
- ▶ Distinct handlers for each HTTP method (e.g., `doGet()`)

1. `service()` parses request and determines HTTP method
2. `service()` calls appropriate handler method
3. Handler method processes request

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

JAVASERVER PAGES (JSP)

## JavaServer Pages (JSP)

- ▶ HTML with extra XML-tags
- ▶ (Scripted server-side) Java for the web
- ▶ A way to provide dynamic content in web pages
- ▶ XML-tags act as front-ends for Java classes
- ▶ May include other pages dynamically
- ▶ JSPs are compiled into Java Servlets
- ▶ JSP code is never visible to clients
- ▶ Generates response (HTML) dynamically
- ▶ Model-View-Controller pattern recommended

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects

JAVA SERVER PAGES (JSP)  
**JSP Syntax**

JSP may contain

- ▶ Static HTML
- ▶ Directives
- ▶ Scripting Elements
- ▶ Actions
- ▶ Comments
- ▶ Tag Libraries
- ▶ Implicit Objects

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects

JAVA SERVER PAGES (JSP)  
**JSP Script Tags**

- ▶ Directives  
`<%@ ... %>`
- ▶ Declarations  
`<%! ... %>`
- ▶ Scriptlets  
`<% ... %>`
- ▶ Expressions  
`<%= ... %>`
- ▶ Comments  
`<%-- ... --%>`

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects

JAVA SERVER PAGES (JSP) – Static HTML  
**Static HTML**

- ▶ All HTML is treated by JSP as static text
- ▶ HTML may be mixed with JSP in any way
- ▶ All non-JSP tags are treated as HTML

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects  
Web Tools

JAVA SERVER PAGES (JSP) – Directives  
**Directives**

`<%@ ... %>`

Types of JSP directives

- ▶ Page
- ▶ Include
- ▶ Tag Libraries (aka Custom Tags)

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects

JAVA SERVER PAGES (JSP) – Directives  
**Page Directives**

`<%@ page attribute="..." %>`

- ▶ Instructs the JSP engine how to process the JSP
- ▶ Attributes determine directive content

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects  
Web Tools

JAVA SERVER PAGES (JSP) – Directives  
**Page Directive Attributes**

- ▶ **language** - selects scripting language (Java)
- ▶ **extends** - base class for generated Servlet
- ▶ **import** - Java class / package import
- ▶ **session** - enable session tracking (default: true)
- ▶ **buffer** - set output buffer size
- ▶ **autoFlush** - enable auto flushing of output buffer
- ▶ **isThreadSafe** - thread safe marker
- ▶ **info** - page information (author, version, copyright etc)
- ▶ **errorPage** - set default error page
- ▶ **isErrorPage** - enable exception tracking on page
- ▶ **contentType** - set response mime type

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Tags  
Custom Tags  
JSP Elements  
JSP Pages  
JSP Directives

JAVA SERVER PAGES (JSP) – Directives

## Include Directive

```
<%@ include file="page.jsp" %>
```

- ▶ Includes another page at translation time
- ▶ Included page becomes part of Servlet
- ▶ Generates error if page not found

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Tags  
Custom Tags  
JSP Elements  
JSP Pages  
JSP Directives

JAVA SERVER PAGES (JSP) – Directives

## Tag Libraries

```
<%@ taglib uri="taglib.tld" prefix="prefix" %>
```

- ▶ Loads a tag library
- ▶ Tags are usable via the specified tag prefix
- ▶ The JSP version of language extension

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Tags  
Custom Tags  
JSP Elements  
JSP Pages  
JSP Directives

JAVA SERVER PAGES (JSP) – Directives

## Directive Examples

```
<%@ page import="examples.*, examples.tags.*" %>
<%@ taglib uri="/WEB-INF/examples-taglib.tld"
    prefix="examples" %>

<%@ include file="/includes/head.jsp" %>
...
<examples:ValidateParameters
    parameters="name,age"/>
...
<%@ include file="/includes/foot.jsp" %>
```

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Tags  
Custom Tags  
JSP Elements  
JSP Pages  
JSP Directives

JAVA SERVER PAGES (JSP) – Scripting Elements

## Scripting Elements

- ▶ Declarations
 

```
<%! ... %>
```
- ▶ Scriptlets
 

```
<% ... %>
```
- ▶ Expressions
 

```
<%= ... %>
```

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Tags  
Custom Tags  
JSP Elements  
JSP Pages  
JSP Directives

JAVA SERVER PAGES (JSP) – Scripting Elements

## Declarations

- ▶ Content placed in Servlet body (members)
- ▶ Used to declare members and methods
- ▶ Does (usually) not produce HTML output
- ▶ Declared content is later used by Scriptlets or expressions
- ▶ Lines must be terminated with a semicolon

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Predefined Tags  
Custom Tags  
JSP Elements  
JSP Pages  
JSP Directives

JAVA SERVER PAGES (JSP) – Scripting Elements

## Declaration examples

```
<%!
    public int getSum (int x, int y)
    {
        return x + y;
    }
%>

<jsp:declaration>
    public int getSum (int x, int y)
    {
        return x + y;
    }
</jsp:declaration>
```

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University



Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions

JAVA SERVER PAGES (JSP) – Scripting Elements

## Scriptlets

- ▶ Content placed in Servlet `_jspService()` method (local variables and in-line code)
- ▶ Used to embed Java code directly in the page
- ▶ May produce HTML output (via `out.println()`)
- ▶ Lines must be terminated with a semicolon

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions

JAVA SERVER PAGES (JSP) – Scripting Elements

## Scriptlet examples

```
<%
    int x = 1;
    int y = 2;
    int sum = x + y;
%>
```

```
<jsp:scriptlet>
    int x = 1;
    int y = 2;
    int sum = x + y;
</jsp:scriptlet>
```

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions

JAVA SERVER PAGES (JSP) – Scripting Elements

## Expressions

- ▶ Content output directly to HTML
- ▶ Used as an alias for `out.println()`
- ▶ Code must evaluate to an expression
- ▶ Lines must **not** be terminated with a semicolon

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions

JAVA SERVER PAGES (JSP) – Scripting Elements

## Expression examples

```
1 + 2 = <%= 1 + 2 %>
```

```
1 + 2 =
<jsp:expression>
    1 + 2
</jsp:expression>
```

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions

JAVA SERVER PAGES (JSP) – Actions

## Actions

- ▶ **include** - include another page
- ▶ **forward** - forward request to another resource
- ▶ **param** - specify parameters when calling or forwarding
- ▶ **plug-in** - generates browser-specific code for applets
- ▶ **fallback** - content if browser does not support applets
- ▶ **getProperty** - get a property from a JavaBean
- ▶ **setProperty** - set a property on a JavaBean
- ▶ **useBean** - use a JavaBean

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)  
Basic HTML  
Directives  
Scripting Elements  
Actions

JAVA SERVER PAGES (JSP) – Actions

## Include Action

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

- ▶ Includes another page at request time
- ▶ Generates a request to included page Servlet
- ▶ Ignored if page not found
- ▶ Usually used to call declared methods
- ▶ Control is returned

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University



Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions

Comments  
Tag Libraries  
Form Elements  
Annotations  
JSP Annotations  
JSP Annotations  
JSP Annotations

Java Application Programming  
JavaServer Pages  
Department of Computing Science, Umeå University

P-O Östberg

JAVASERVER PAGES (JSP) – Actions

## Forward Action

```
<jsp:forward page="page.jsp"/>
```

- ▶ Forwards request to another resource
- ▶ Control is not returned

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions

Comments  
Tag Libraries  
Form Elements  
Annotations  
JSP Annotations  
JSP Annotations  
JSP Annotations

Java Application Programming  
JavaServer Pages  
Department of Computing Science, Umeå University

P-O Östberg

JAVASERVER PAGES (JSP) – Actions

## Param Action

```
<jsp:forward page="page.jsp">
  <jsp:param name="name" value="value"/>
</jsp:forward>
```

- ▶ Used to specify parameters when including / forwarding

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions

Comments  
Tag Libraries  
Form Elements  
Annotations  
JSP Annotations  
JSP Annotations  
JSP Annotations

Java Application Programming  
JavaServer Pages  
Department of Computing Science, Umeå University

P-O Östberg

JAVASERVER PAGES (JSP) – Comments

## HTML Comments

```
<!-- HTML comment -->
```

- ▶ Evaluated by server (as HTML)
- ▶ Part of the server response / web page
- ▶ Visible in page source from web browser

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions

Comments  
Tag Libraries  
Form Elements  
Annotations  
JSP Annotations  
JSP Annotations  
JSP Annotations

Java Application Programming  
JavaServer Pages  
Department of Computing Science, Umeå University

P-O Östberg

JAVASERVER PAGES (JSP) – Comments

## JSP Comments

```
<%-- JSP comment --%>
```

- ▶ Not evaluated by server
- ▶ Not part of the server response / web page
- ▶ Not visible in page source from web browser

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions

Comments  
Tag Libraries  
Form Elements  
Annotations  
JSP Annotations  
JSP Annotations  
JSP Annotations

Java Application Programming  
JavaServer Pages  
Department of Computing Science, Umeå University

P-O Östberg

JAVASERVER PAGES (JSP) – Tag Libraries

## Tag Libraries

- ▶ Java classes
- ▶ Implements the JSP Tag Extension API interfaces
- ▶ Usable as JSP tags in JSPs
- ▶ Java class coupled to a tag using a XML-descriptor
- ▶ Tags can control JSP processing
- ▶ A very convenient way to encapsulate complex code
- ▶ Supports role-based web development

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions

Comments  
Tag Libraries  
Form Elements  
Annotations  
JSP Annotations  
JSP Annotations  
JSP Annotations

Java Application Programming  
JavaServer Pages  
Department of Computing Science, Umeå University

P-O Östberg

JAVASERVER PAGES (JSP) – Implicit Objects

## Implicit Objects

- ▶ **request** - HTTP request
- ▶ **response** - HTTP response
- ▶ **out** - response stream
- ▶ **session** - web application session
- ▶ **pageContext** - page context data
- ▶ **application** - Servlet context data
- ▶ **config** - Servlet configuration data
- ▶ **page** - Servlet object
- ▶ **exception** - exception data

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects

## JAVASERVER PAGES (JSP) – Implicit Objects

# The Request Object

- ▶ Represents the HTTP request
- ▶ Contains all info in the request
- ▶ Exposes an API for traversing request data
- ▶ Performs URL-decoding of data
- ▶ Provides uniform ways to read data regardless of method

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects

## JAVASERVER PAGES (JSP) – Implicit Objects

# Reading Parameters

- ▶ `request.getParameter("name")`
- ▶ `request.getParameterValues("name")`
- ▶ `request.getParameterNames()`
- ▶ `request.getParameterMap()`

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects

## JAVASERVER PAGES (JSP) – Implicit Objects

# Reading Parameters

- ▶ Always check for null
- ▶ Always provide a default value (as appropriate)

Parameter	getParameter() returns
not present in request	null
contains empty string	empty string

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects  
Cookies

## JAVASERVER PAGES (JSP) – Cookies

# Cookies

- ▶ HTTP connections may be closed at any time
- ▶ Need a way to identify requests from the same client
- ▶ Cookies are small text fragments sent in headers
- ▶ Cookies are stored on client computer file systems and are used to identify site visitors (track users)

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects  
Cookies

## JAVASERVER PAGES (JSP) – Cookies

# Browsers & Cookies

- ▶ Maximal cookie size 4096 bytes
- ▶ Max 20 cookies per site
- ▶ Max 300 cookies in total
- ▶ Stored on client computer
- ▶ Cookies can be blocked by browsers
- ▶ Cookies can be removed from client computer
- ▶ Cookies can be altered
- ▶ Don't store sensitive data in cookies
- ▶ Use cookies, but don't depend on them

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Static HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Implicit Objects  
Cookies  
Sessions

## JAVASERVER PAGES (JSP) – Sessions

# Sessions

- ▶ Data contexts stored on the web server and shared between requests from the same user
- ▶ Can store anything (POJO)
- ▶ Can be serialized to databases or short-lived
- ▶ Need a way to identify session for new requests
  - Cookies
  - URL rewrites
  - hidden form fields
- ▶ The JSP session API hides the session identification

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Internationalization  
Localization  
JSP Processing

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

JAVASERVER PAGES (JSP) – JSP Processing

## JSP Processing

1. A JSP page is requested
2. Server checks if a Java Servlet for the page exists
3. If no Servlet is found (or newer JSP is detected), the JSP is translated to Java (a Servlet class is created)
4. The Java Servlet is compiled
5. The Java Servlet is invoked and processes the request

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Internationalization  
Localization  
JSP Processing

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

JAVASERVER PAGES (JSP) – JSP Processing

## JSP Translation

- ▶ JSP declarations become
  - Servlet class members
  - Servlet class methods
- ▶ JSP scriptlets become
  - local variables in the Servlet `service()` method
- ▶ JSP expressions become
  - `out.print()` calls in - " -

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Internationalization  
Localization  
Client-Side Components

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

CLIENT-SIDE COMPONENTS

## Client-Side Components

- ▶ Compiled programs that execute on the client
- ▶ Downloaded from the server when needed
- ▶ Reduces client installation needs
- ▶ Description of the component part of the HTML
- ▶ Useful for providing advanced user interfaces
- ▶ Plenty of third party components available
- ▶ Commercial markets for special purpose components

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Internationalization  
Localization  
Client-Side Components

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

CLIENT-SIDE COMPONENTS

## Java Applets

- ▶ Java objects extending the Applet class
- ▶ Can be used in any HTML environment
- ▶ Developed in Java
- ▶ Contains a sandbox security model
- ▶ Supports code signing
- ▶ Distributed as standalone classes or JARs

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Internationalization  
Localization  
Client-Side Components

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

CLIENT-SIDE COMPONENTS

## Java Applets

- ▶ Extend `Applet`
  - creates an AWT container
  - functions as a `Frame` for your applet
- ▶ Extend `JApplet`
  - creates a heavy-weight Java Swing container
  - functions as a `JFrame` for your applet
- ▶ Assure thread safe implementations of applets (invoke worker threads via `SwingUtilities.invokeLaterAndWait()`)

Umeå University

Today  
Web Technologies  
Server-Side Web Development  
JavaServer Pages (JSP)

Basic HTML  
Directives  
Scripting Elements  
Actions  
Comments  
Tag Libraries  
Internationalization  
Localization  
Client-Side Components

Java Application Programming  
JavaServer Pages  
P-O Östberg  
Department of Computing Science, Umeå University

CLIENT-SIDE COMPONENTS

## Applet Lifecycle

1. `init()`
  - performs applet initialization
  - called after the applet has received its parameters
2. `start()`
  - activates the applet
  - called after `init()` and whenever an applet page receives focus
3. `stop()`
  - deactivates the applet
  - called when an applet page loses focus
4. `destroy()`
  - deinitializes the applet
  - called when the browser is shut down



## Applet Restrictions

- ▶ Applets cannot load libraries or define native methods
- ▶ An applet cannot ordinarily read or write files on the host that is executing it
- ▶ An applet cannot make network connections except to the host that it came from
- ▶ An applet cannot start any program on the host that is executing it
- ▶ An applet cannot read certain system properties
- ▶ Windows that an applet brings up look different than windows that an application brings up

Each browser has a SecurityManager object that implements its security policies. When a SecurityManager detects a violation, it throws a SecurityException. Your applet can catch this SecurityException and react appropriately.



## Applet Capabilities

- ▶ Applets provide a way to create non-web GUIs to web applications
- ▶ Applets can make network connections to the host they came from  

```
String host = getCodeBase().getHost();
```
- ▶ Applets running within a Web browser can easily cause HTML documents to be displayed  

```
getAppletContext().showDocument(url,browsertitle)
```
- ▶ Applets can invoke public methods of other applets on the same page
- ▶ Applets that are loaded from the local file system (from a directory in the user's CLASSPATH) have none of the restrictions that applets loaded over the network have



## Applets as Applications

```
PokerClockApplet applet = new PokerClockApplet();

JFrame frame = new JFrame("Poker Clock Applet");
frame.setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
frame.getContentPane().add(applet, BorderLayout.CENTER);
applet.init();
applet.start();
frame.pack();
frame.setVisible(true);
```



## Next Time

- ▶ Web Services