





## Architecture des Systèmes d'Information



JSP

Java Server Pages (1.2)

**Introductions** 

<%(x) ..... %> X

Declaration : (!)

Enclosed Java expressions : (=)

Scriptlets : ( )

JSP Directives : (@)

Comments : (--)





## JSP concepts



- Servlet programming is a mixing of :
  - presentation (view),
  - data (model),
  - business logic (control).
- Separation of labor:
  - page authors
  - Java programmers



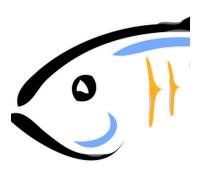
### Other features



Separation of static from dynamic content.

### JEE Presentation Frameworks

- JSP :Java Server Pages
- EL: Epression Language
- JSTL: Java Standard Tag Lib
- UEL: Unified Epression Language
- JSF : Java Serve Face, Facelets











## JEE versions and JSP versions

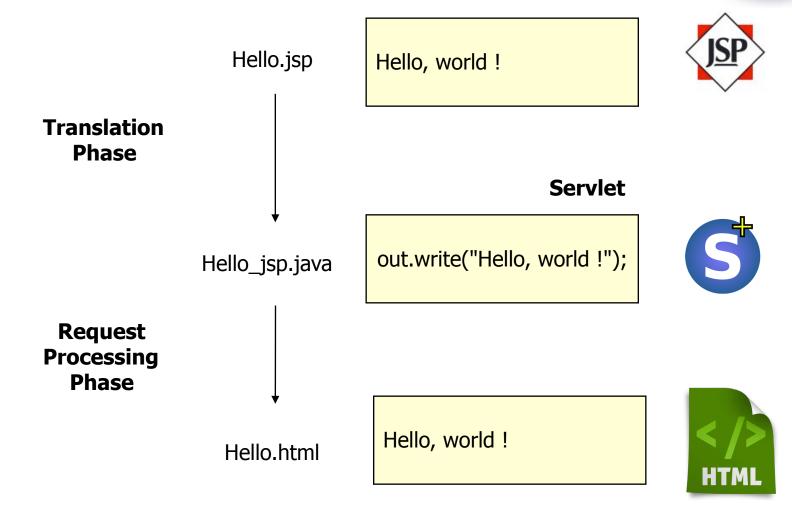


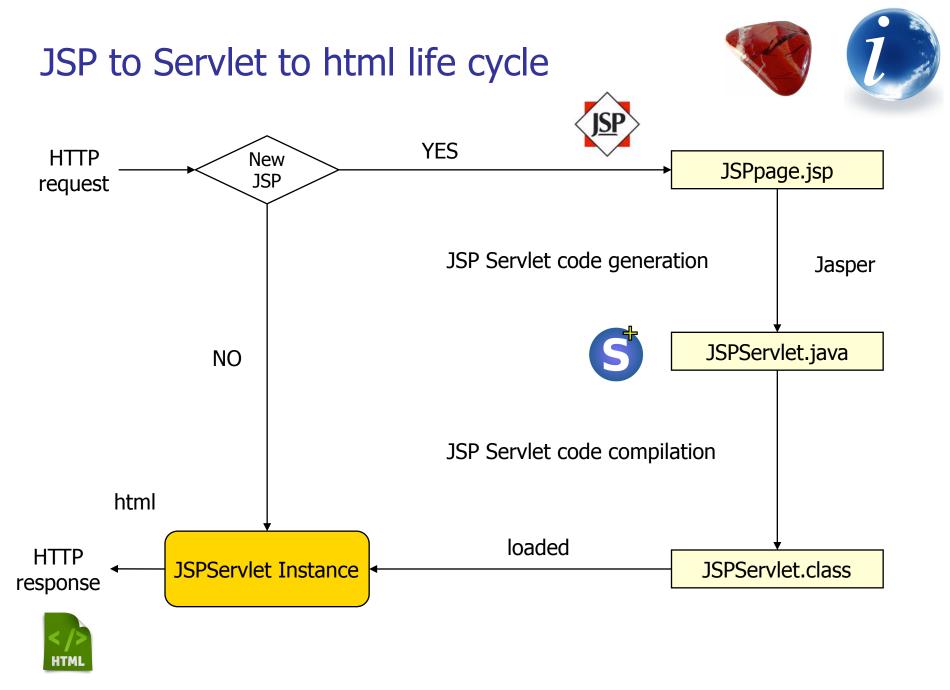
JEE	Date	Servlet	JSP	EL	JSTL	JSF
	janv-99	Servlet 1.1				
	juin-99		JSP 1.0			
JEE2 (J2EE 1.2)	12 Dec 1999	Servlet 2.2	JSP 1.1			
JEE3 (J2EE 1.3)	24-sept-01	Servlet 2.3	JSP 1.2			
JEE4 (J2EE 1.4)	11-nov-03	Servlet 2.4	JSP 2.0			
JEE5	11 May 2006	Servlet 2.5	JSP 2.1		JSTL 1.2	JSF 1.2
JEE6	10 Dec 2009	Servlet 3.0	JSP 2.2	EL 2.2	JSTL 1.2	JSF 2.0
JEE7	12-juin-13	Servlet 3.1	JSP 2.3	EL 3.0	JSTL 1.2	JSF 2.2



## From jsp to servlet to html (static) Jsp=Html







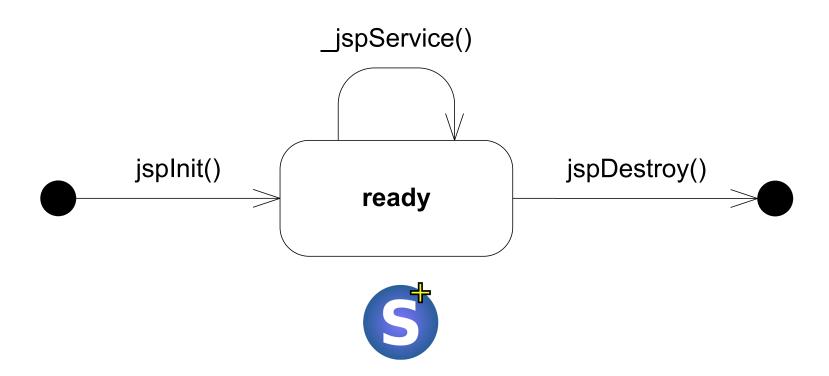
### javax.servlet.jsp



- javax.servlet.jsp package defines two interfaces:
  - JSPPage
  - HttpJspPage
- These interfaces defines the three methods for the compiled JSP page.
- These methods are:
  - *jspInit()*
  - jspDestroy()
  - \_jspService(HttpServletRequest request,HttpServletResponse response)

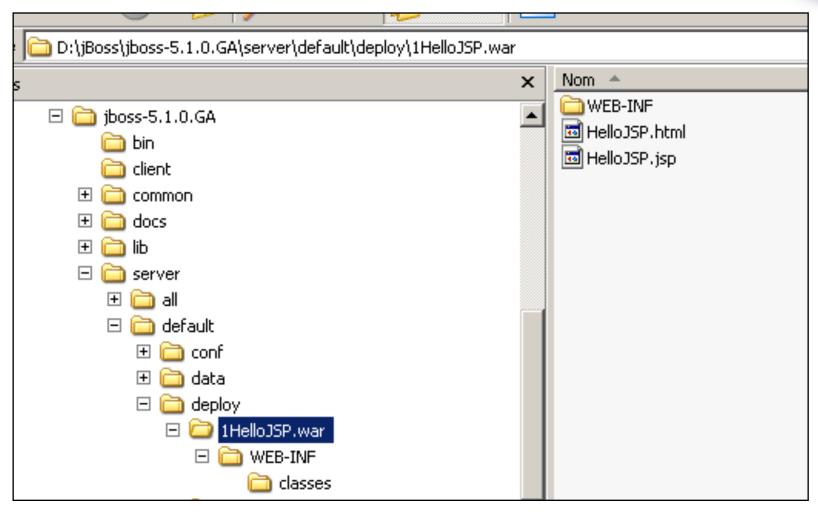
#### JSP states





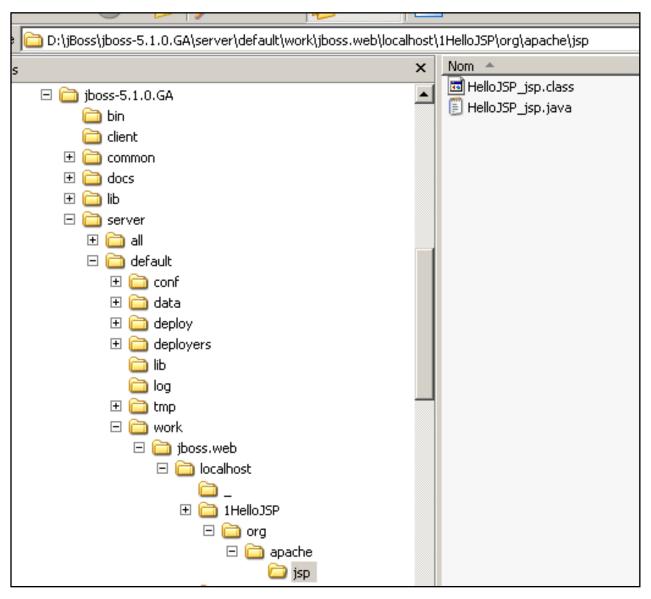
#### JSP location





### Compiled JSP servlet in JBoss work directory





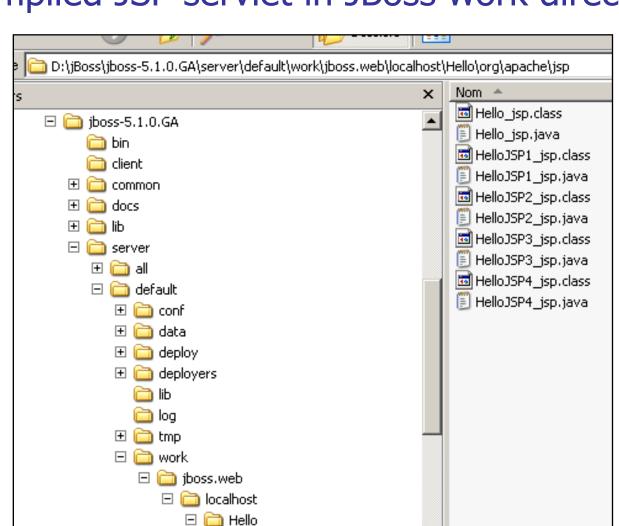
### HelloJSP compiled to servlet





```
ispService(HttpServletRequest request, HttpServletResponse response)
{
    out.write("<html>\n");
    out.write("\t<head>\n");
    out.write("\t\t<title>Hello JSP</title>\n");
    out.write("\t</head>\n");
    out.write("\t<body>\n");
    out.write("\t\t<h2>Hello</h2>\n");
    out.write("\t\t\World\n");
    out.write("\t</body>\n");
    out.write("</html> \n");
    out.write("\n");
```

# JSP Compiled JSP servlet in JBoss work directory



🖃 🧰 org

□ apache

įsp.

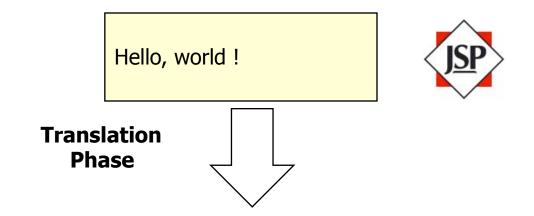
## Compiled JSP servlet in JBoss work directory



D:\jBoss\jboss-5.1.0.GA\server\default\work\jboss.web\localhost\Hello\org\apache\jsp					
	×	Nom A			
iboss-5.1.0.GA   ibin   icient   ici		Hello_jsp.class Hello_jsp.java HelloDate_jsp.class HelloDate_jsp.java HelloPrtln_jsp.class HelloPrtln_jsp.java			

## Simple html to JSP compiled servlet







## Static VS dynamic web page



Static:







**Dynamic:** 







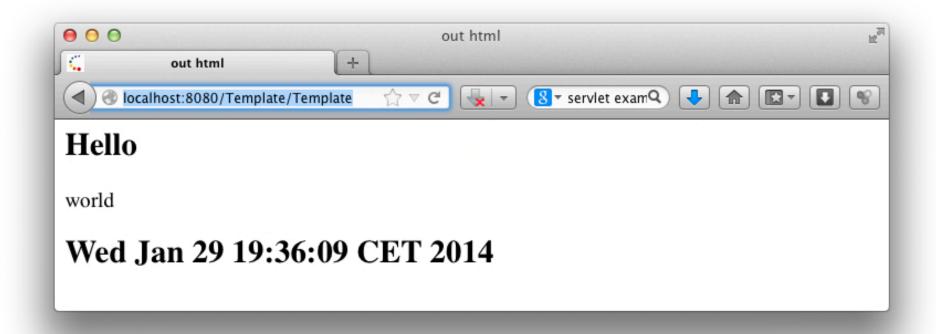


- <%(x) ..... %> X
  - Declaration : (!)
  - Enclosed Java expressions : (=)
  - Scriptlets : ( )
  - JSP Directives : (@)
  - Comments: (--)

## Dynamic Html page







## Html Hello world servlet: dynamic





```
import java.io.*;
import javax.servlet.http.*;
import javax.servlet.*;
public class HelloServlet extends HttpServlet {
 public void doGet (
                                  HttpServletRequest request,
                                  HttpServletResponse response)
  throws ServletException, IOException {
           response.setContentType ("text/plain");
           PrintWriter out = response.getWriter();
           out.println(
                                              "<html>"+
                                                         "<head>"+
                                                                     "<title>out html</title>"+
                                                         "</head>"+
                                                         "<body>"+
                                                                     "<h2>hello</h2>" +
                                                                     "world"+
                                                         "<h2>");
                       out.print(new java.util.Date() );
                                                         "</h2>" +
           out.println(
                                                          "</body>"+
                                              "</html>"
           );
```

## dynamic





```
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
                       throws ServletException, IOException {
               PrintWriter out = response.getWriter();
               out.println(
                               "<html>"+
                                       "<head>"+
                                              "<title>out html</title>"+
                                       "</head>"+
                                       "<body>"+
                                               "<h2>Hello</h2>" +
                                               "world"+
                                               "<h2>");
               out.print(new java.util.Date() );
               out.println(
                              "</body>"+
"</html>"
                                               "</h2>"+
                               );
       }
```

## HTML Template



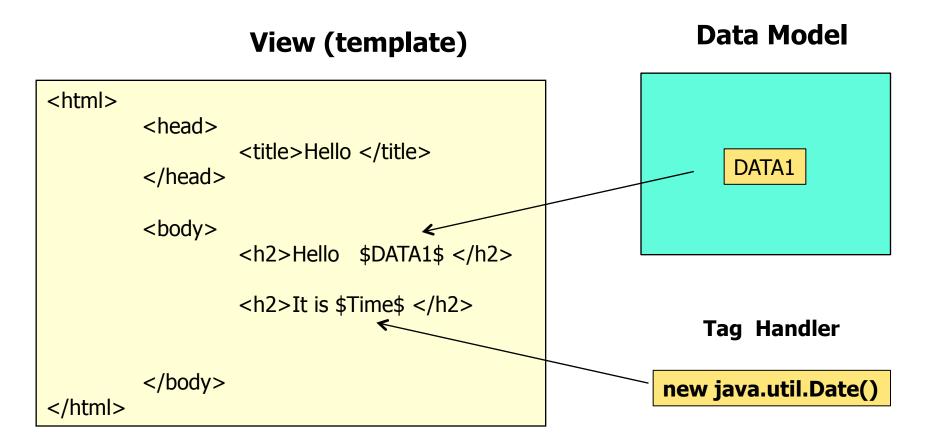
```
<html>
      <head>
            <title>Hello </title>
      </head>
      <body>
            <h2>Hello </h2>
            <h2>It is $Time$ </h2>
      </body>
</html>
```

Script Tag Label

**HelloTemplate** 

## HTML Template Engine: Dynamic





#### HelloTemplate

\$ : Label
</> : Tag

### JSP notation elements





- JSP directives are messages for the JSP engine :
  - Transformation : Compilation Directives
- JSP Script: Java executed during request processing time
  - Declarations : Servlet objects declaration
  - Expressions : One line embedded Java expressions
  - Scriptlets: Multiple lines Java block with control flow statements
  - Comments
- JSP Actions, request processing flow
  - HTML, XML extension expanded during request processing time

#### JSP notation



- JSP Directives (<%@)</li>
- Declaration (<%!)</p>
- Scriptlets (<%)</li>
- Java expressions (<%=)Enclosed</p>
- Comments (<%-- )</p>

JSP Action (<jsp:... />)

```
 <%(x) ..... %> X
   Declaration : (!)
   Enclosed Java expressions : (=)
   Scriptlets : ( )
   JSP Directives : (@)
   Comments : (--)
```



#### JSP notation

1

- JSP Directives (@)
  - <%@ page import="java.util.\*" %>
- Declaration (!)
  - <%! int ClassVariable = 0; %>
  - Inserted as class Variable
- Enclosed Java expressions (=)
  - <%= and %>
  - Inserted as parameter of an out.print() call.
- Scriptlets ( )
  - <% and %>
  - A scriptlet contains Java code that is executed every time the JSP servlet is invoked.
- Comments
  - <%-- a comment --%>



### JSP syntaxe <%(x) ..... %>

- <%(x) .... %> X
  - Declaration : (!)
  - Enclosed Java expressions : (=)
  - Scriptlets: ()
  - JSP Directives : (@)
  - Comments: (--)





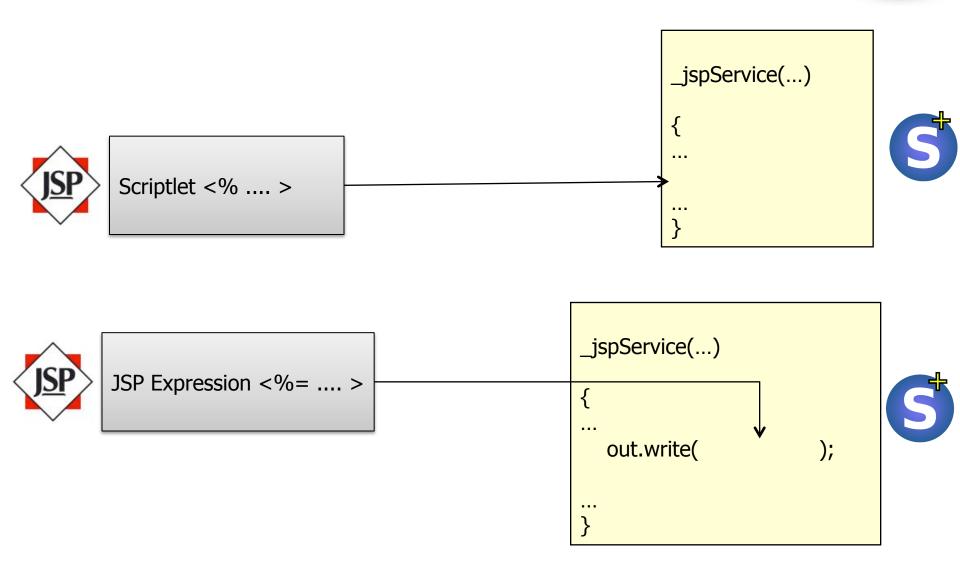
## Translation principles



- The HTML statements in the JSP file are written out as part of the response via out.write(...), without modification.
- JSP Scriptlet <% .... >: used to include Java statements, or part of Java statement.
  - The Java statements are placed inside the \_jspService() method of the translated servlet.
  - Scriptlets form the program logic, executed during request processing.
  - Java Source Code
- JSP Expression <%= .... >: used to evaluate a single Java expression to obtain a value.
  - The Java expression is placed inside a out.print(...).
  - The Java expression will be evaluated during request response processing, and the result of the evaluation written out as part of the response message.

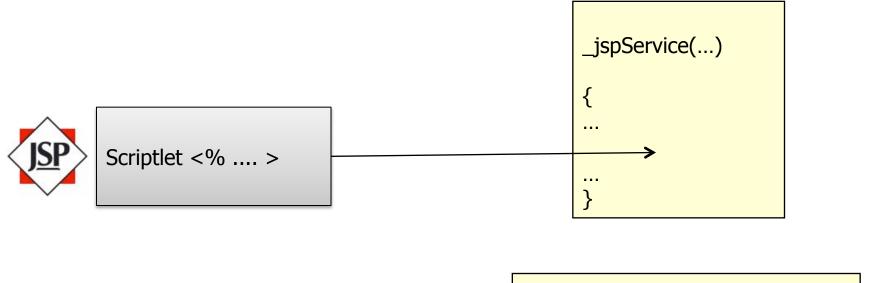
## JSP element translation principle



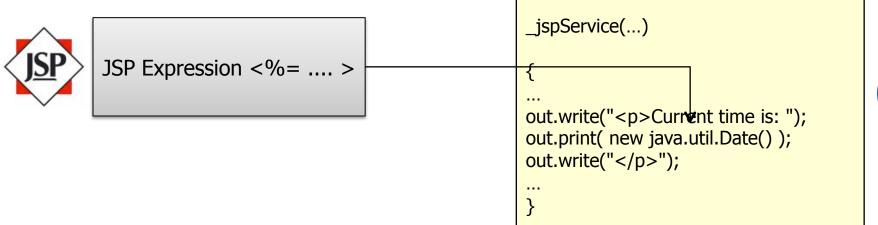


## JSP element translation principle









### JSP scriplets



```
<%(x) ..... %> X

Declaration : (!)

Enclosed Java expressions : (=)

Scriptlets : ()

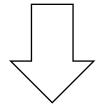
JSP Directives : (@)

Comments : (--)
```

System.out.println( "console JSP output 1" ); java.util.Date date = new java.util.Date();



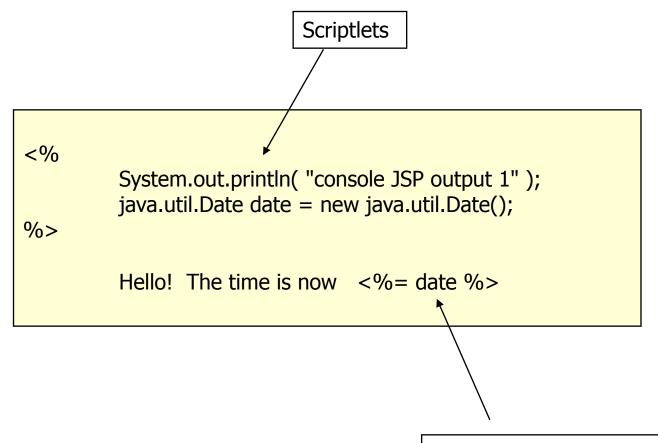
Translation Phase





### Simple JSP 1





- <%(x) ..... %> X
  - Declaration : (!)
  - Enclosed Java expressions : (=)
  - Scriptlets : ( )
  - JSP Directives : (@)
  - Comments: (--)

Enclosed Java expressions

## Simple JSP 1 compiled servlet



Scriptlets

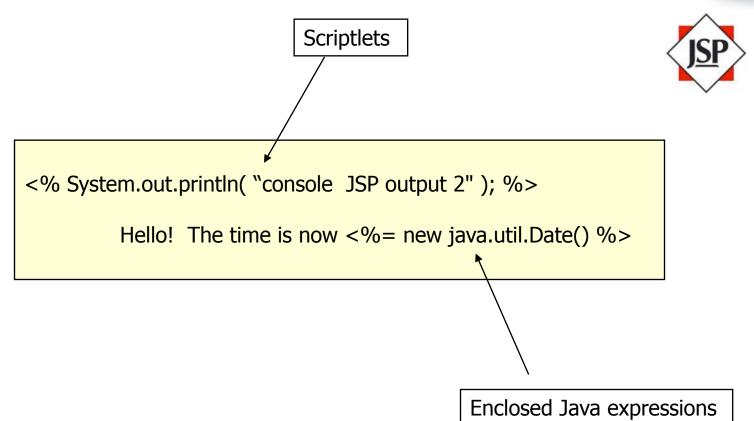
```
_jspService(HttpServletRequest request, HttpServletResponse response)
System.out.println( "console JSP output 1" );
          java.util.Date date = new java.util.Date();
    out.write("\n");
    out.write("\t\tHello! The time is now ");
    out.print( date );
```

Enclosed Java expressions



### Simple JSP 2





<%(x) .... %> X
Declaration : (!)
Enclosed Java expressions : (=)
Scriptlets : ()
JSP Directives : (@)
Comments : (--)

### Simple JSP 2 compiled servlet





```
_jspService(HttpServletRequest request, HttpServletResponse response)
System.out.println( "console JSP output 2" );
    out.write("\n");
    out.write("\n");
    out.write("Hello! The time is now");
    out.print( new java.util.Date() );
    out.write('\n');
```

**Enclosed Java expressions** 

## Simple JSP 3 scriptlets only





```
Scriptlets

<%
System.out.println( " console JSP output 3 " );
out.println( String.valueOf( new java.util.Date() ));
%>
```

```
<%(x) ..... %> X
Declaration : (!)
Enclosed Java expressions : (=)
Scriptlets : ( )
JSP Directives : (@)
Comments : (--)
```

### Simple JSP 3 compiled servlet

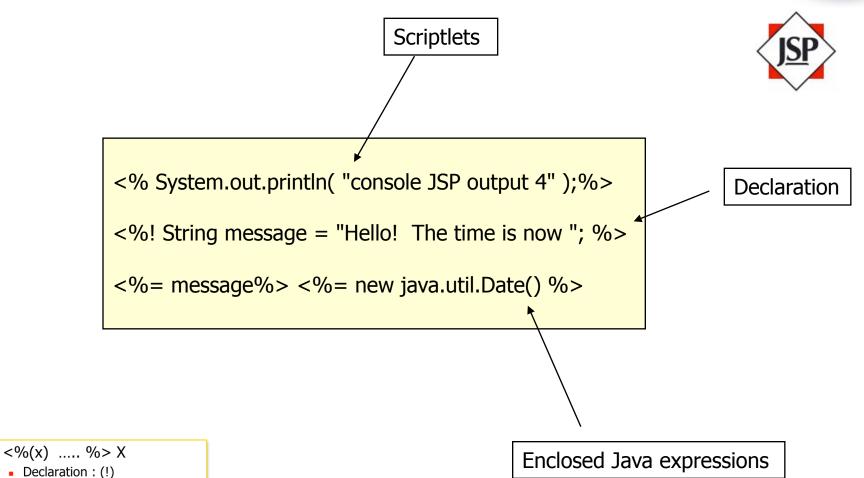




```
_jspService(HttpServletRequest request, HttpServletResponse response)
{
...
System.out.println( "console JSP output 3" );
out.println( String.valueOf( new java.util.Date() ));
...
}
```

### Simple JSP 4





Emmanuel fuchs Architecture des Systèmes d'Information

Scriptlets : ( )JSP Directives : (@)Comments : (--)

Enclosed Java expressions : (=)

## Simple JSP 4 compiled servlet

Class Variable



```
public final class HelloDeclare_jsp {
String message = "Hello! The time is now";
     _jspService(HttpServletRequest request, HttpServletResponse response) {
     System.out.println( "console JSP output 4" );
         out.write('\n');
         out.write('\n');
         out.print( message);
         out.print( new java.util.Date() );
         out.write(' ');
         out.write('\n');
```

## JSP Pre-Defined Objects

1

- request,
- response,
- out,
- session,
- application,
- config,
- page



## Object Scope



- Application
- Session
- Request
- Page

## JSP Pre-Defined Objects



 out: A Writer (JspWriter) object used to write response message to the client over the network socket, via methods print() or println().



### HelloJSP compiled to servlet





```
_jspService(HttpServletRequest request, HttpServletResponse response)
javax.servlet.http.HttpSession session = pageContext.getSession();
javax.servlet.ServletContext application = pageContext.getServletContext();
javax.servlet.ServletConfig config = pageContext.getServletConfig();
javax.servlet.jsp.JspWriter out = pageContext.getOut();
java.lang.Object page = this;
```





**Emmanuel fuchs Architecture des Systèmes d'Information**