



Jakarta EE

ENTERPRISE JAVA

Hany ahmed



Set of servers on the Internet that support HTTP

Resources are identified by URLs and usually accessed through hyper linked documents called HTML

HTTP Protocol is being used by a web client (Usually a web browser) to request documents from a web server



HTML

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
```

```
    <title>Hello to my page</title>
```

```
    <link href="styles.css" rel="stylesheet" type="text/css">
```

```
    <script src="script.js" type="application/javascript" />
```

```
  </head>
```

```
  <body>
```

```
    <h1>Hello!</h1>
```

```
    
```

```
    <!-- Need to add more content here -->
```

```
    <p style="background-color: yellow">Welcome to my site</p>
```

```
    <a href="https://www.google.org">Search on google</a>
```

```
  </body>
```

```
</html>
```

HTML forms

```
<form action="registerAction" method="POST">
  name      <input type="text" name="name"><br>
  Password  <input type="password" name="pass">
  Gender
    <input type="radio" name="gender" value="male">Male
    <input type="radio" name="gender" value="female">Female
  Languages
    <input type="checkbox" name="language" value="ar">Arabic
    <input type="checkbox" name="language" value="en">English
  City
    <select name="city">
      <option value="1">Cairo</option>
      <option value="2">Giza</option>
    </select>
  <textarea name="details"></textarea><br />
  <input type="submit" value="Register">
</form>
```



Coding an enterprise application

Listening for incoming requests

Request routing

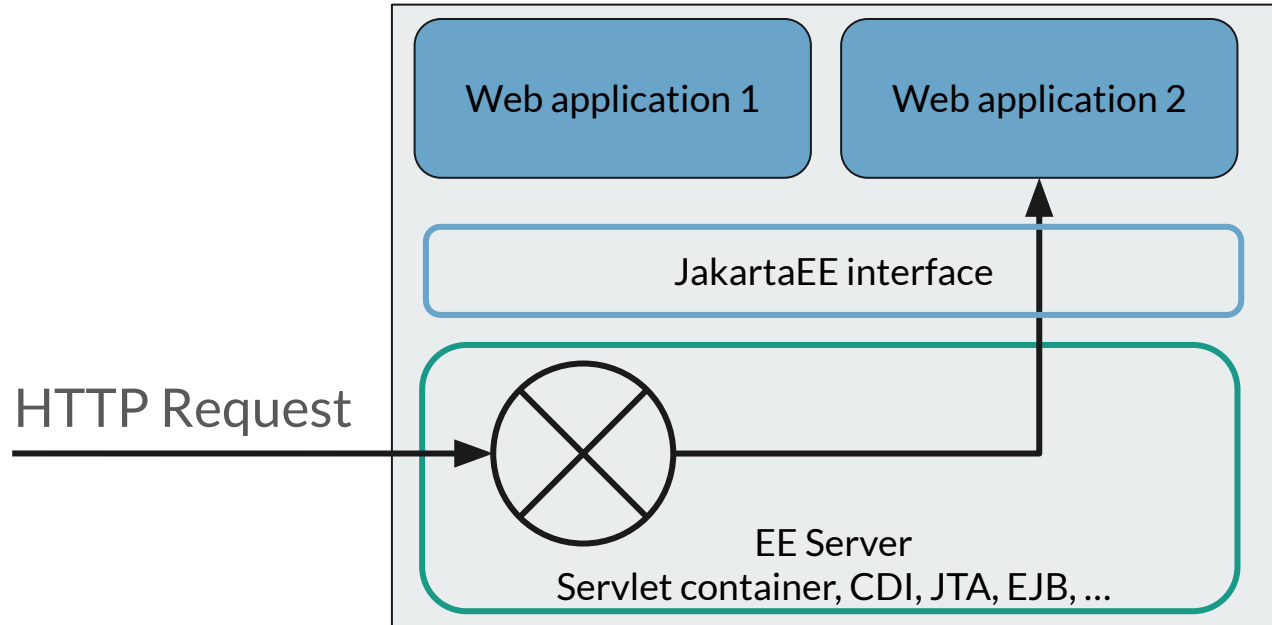
Request/Response processing

Multithreading

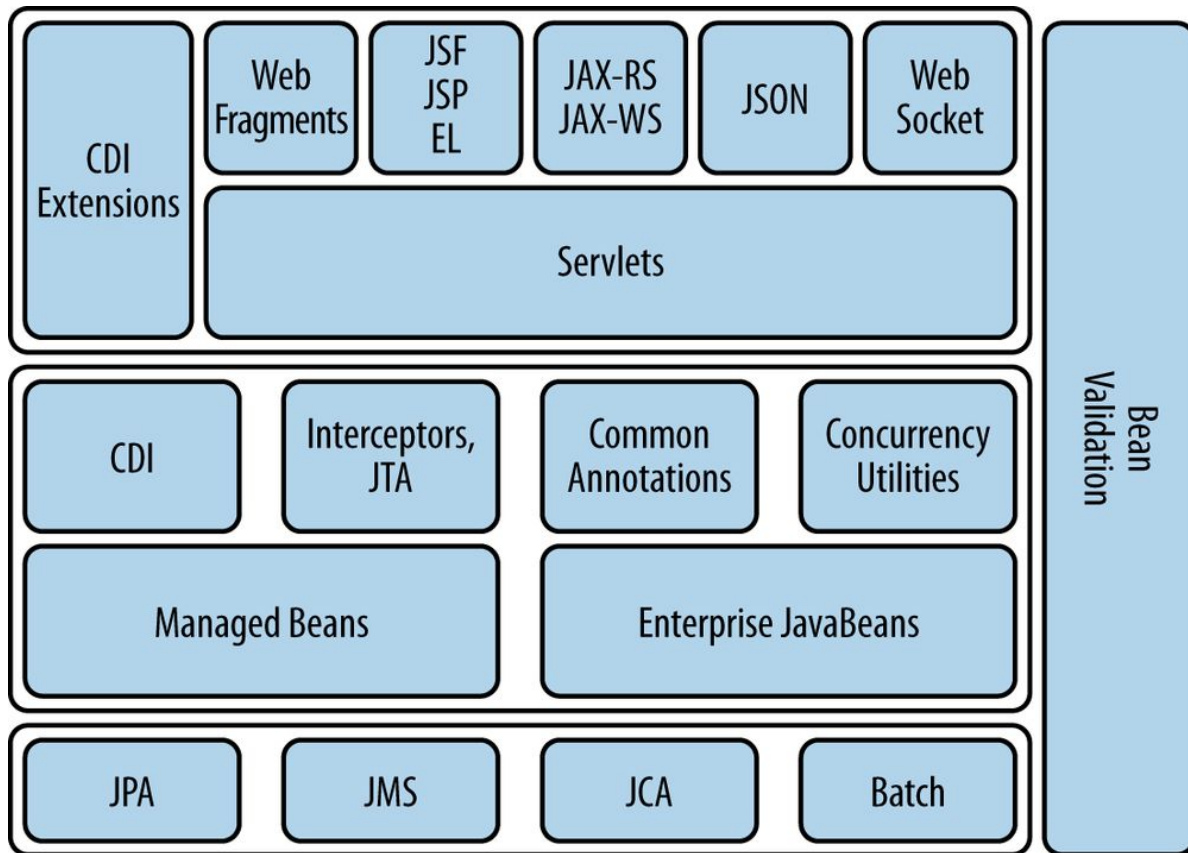
State management

Resource pooling

EE Model



EE Components





Workshop: Running the server

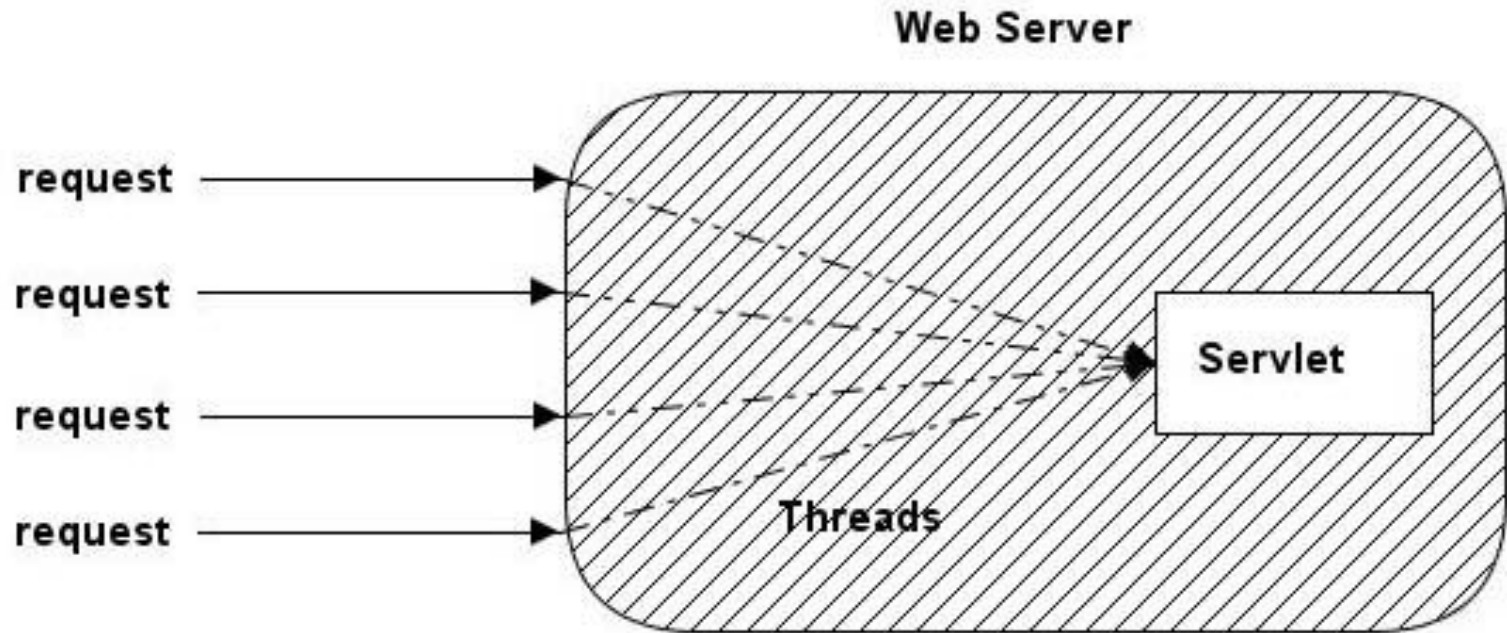
Download tomcat

Start it using bin/startup.sh

Stopping it using bin/shutdown.sh

View logs: logs/catalina.out

Servlet





HttpServlet Example

```
doGet(HttpServletRequest request, HttpServletResponse response){  
  
    response.setContentType("text/html");  
    PrintWriter out = response.getWriter();  
    out.println("<html>");  
    out.println("<head>");  
    out.println("<title>Hello</title>");  
    out.println("</head>");  
    out.println("<body>");  
    out.println("<h1>Hello world</h1>");  
    out.println("</body>");  
    out.println("</html>");  
  
}
```



Workshop: Servlet

Create new maven project

Add dependency for jakartaee-api

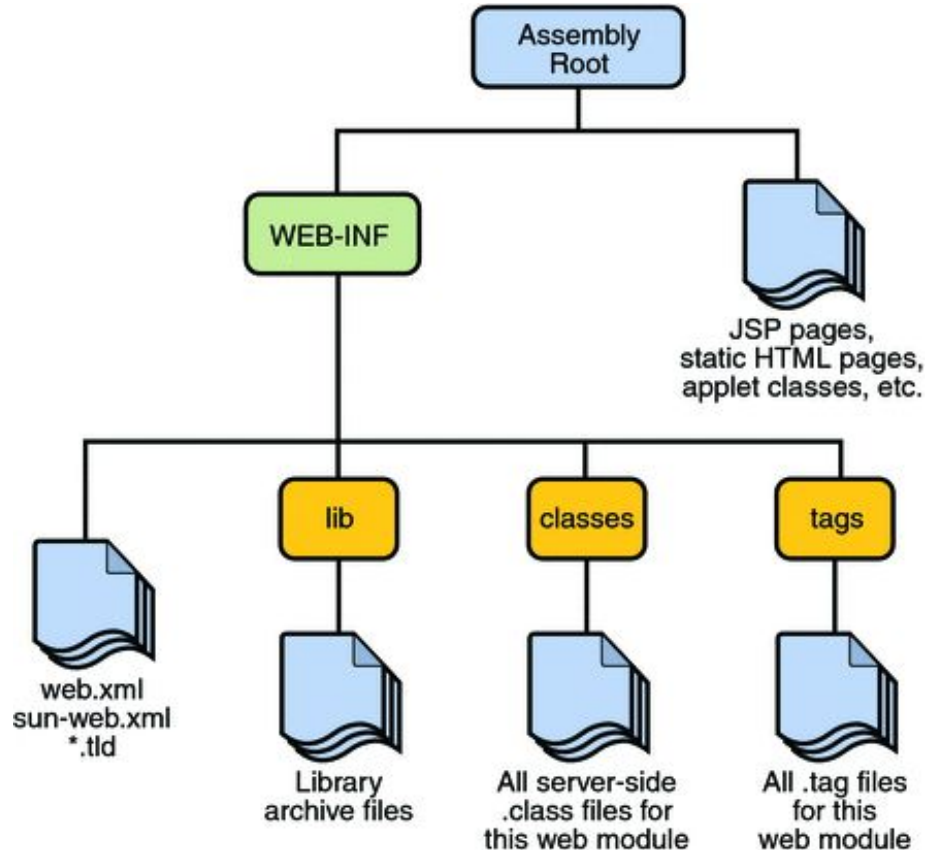
```
<dependency>
  <groupId>jakarta.platform</groupId>
  <artifactId>jakarta.jakartaee-api</artifactId>
  <version>8.0.0</version>
  <scope>provided</scope>
</dependency>
```

Create your first Servlet

package the project into war

Deploy it to tomcat

Web directory structure





Handle request data

URL query parameters or form data

action?**name**=ali&**interest**=reading&**interest**=football

Get parameter value

```
String name = request.getParameter("name");  
String[] arr = request.getParameterValues("interest");
```

Get all parameters

```
Map<String, String[]> map = request.getParameterMap();  
Enumeration<String> en = request.getParameterNames();
```



Request Dispatcher

From request

// absolute

```
request.getRequestDispatcher("/welcome");
```

// relative

```
request.getRequestDispatcher("welcome");
```

forward: to forward the request to another servlet and discard the current

include: to pass request and include response from another servlet



Request attributes

Data held by HttpServletRequest object

```
// set  
request.setAttribute("user", "ahmed");  
  
// get  
String user = (String)  
request.getAttribute("user");  
  
// remove  
request.removeAttribute("user");  
  
// get all attribute names  
Enumeration<String> en =  
request.getAttributeNames();
```



Cookies

// add cookie

```
Cookie cookie = new Cookie("user", "ahmed");  
cookie.setMaxAge(60*60*24*30);  
response.addCookie(cookie);
```

// Get cookie

```
Cookie[] cookies = request.getCookies();  
    if (cookies != null)  
        for (int i = 0; i < cookies.length; i++)  
            Cookie cookie = cookies[i];  
            if(cookieName.equals(cookie.getName()))
```




Cookies notes

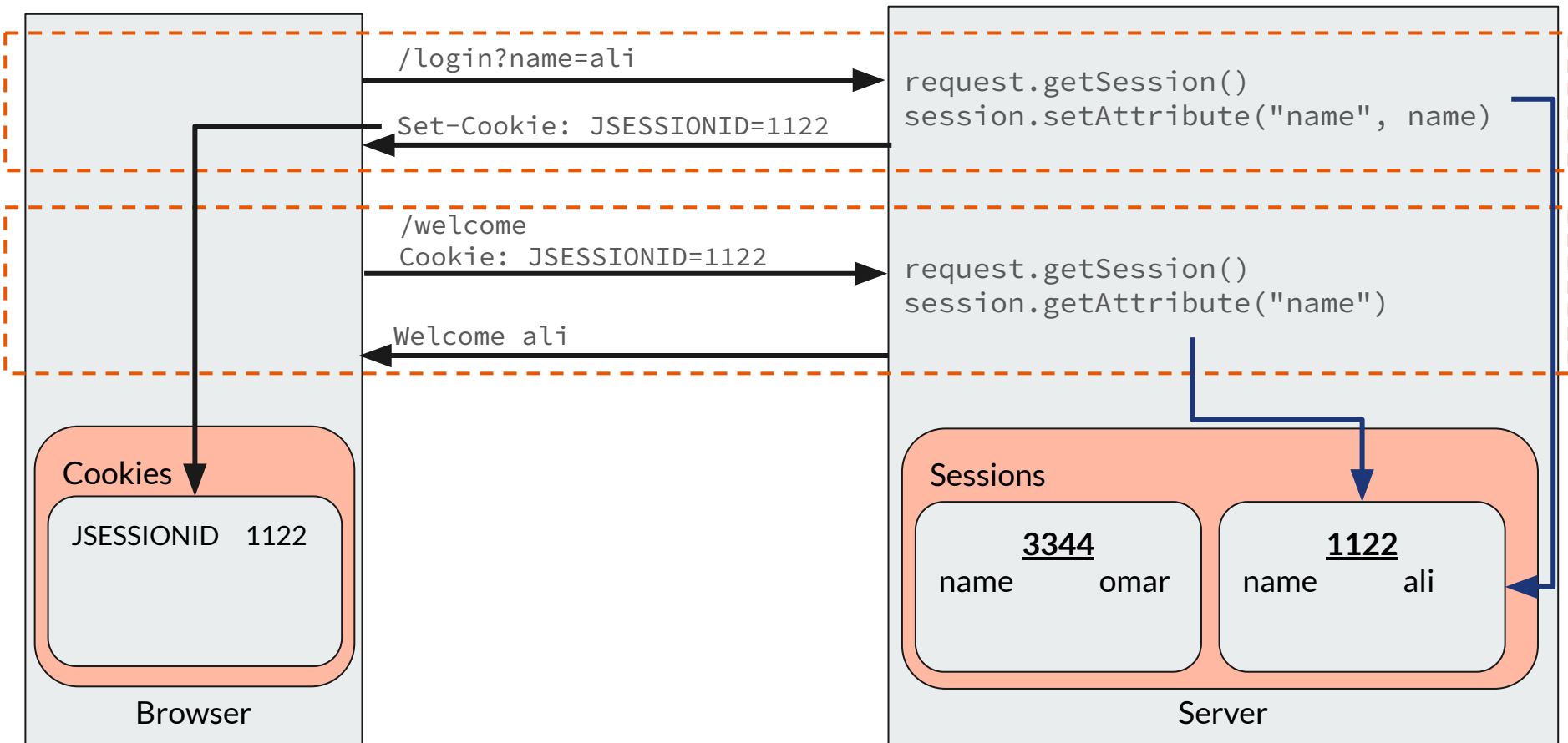
All cookies are sent by default by the browser at every request

Sending a cookie with the same name with a new value would override the old cookie.

Updating a cookie doesn't take any effect until It's sent with the headers.

Setting the maximum age to zero or negative would result in invalidating the cookie.

Session





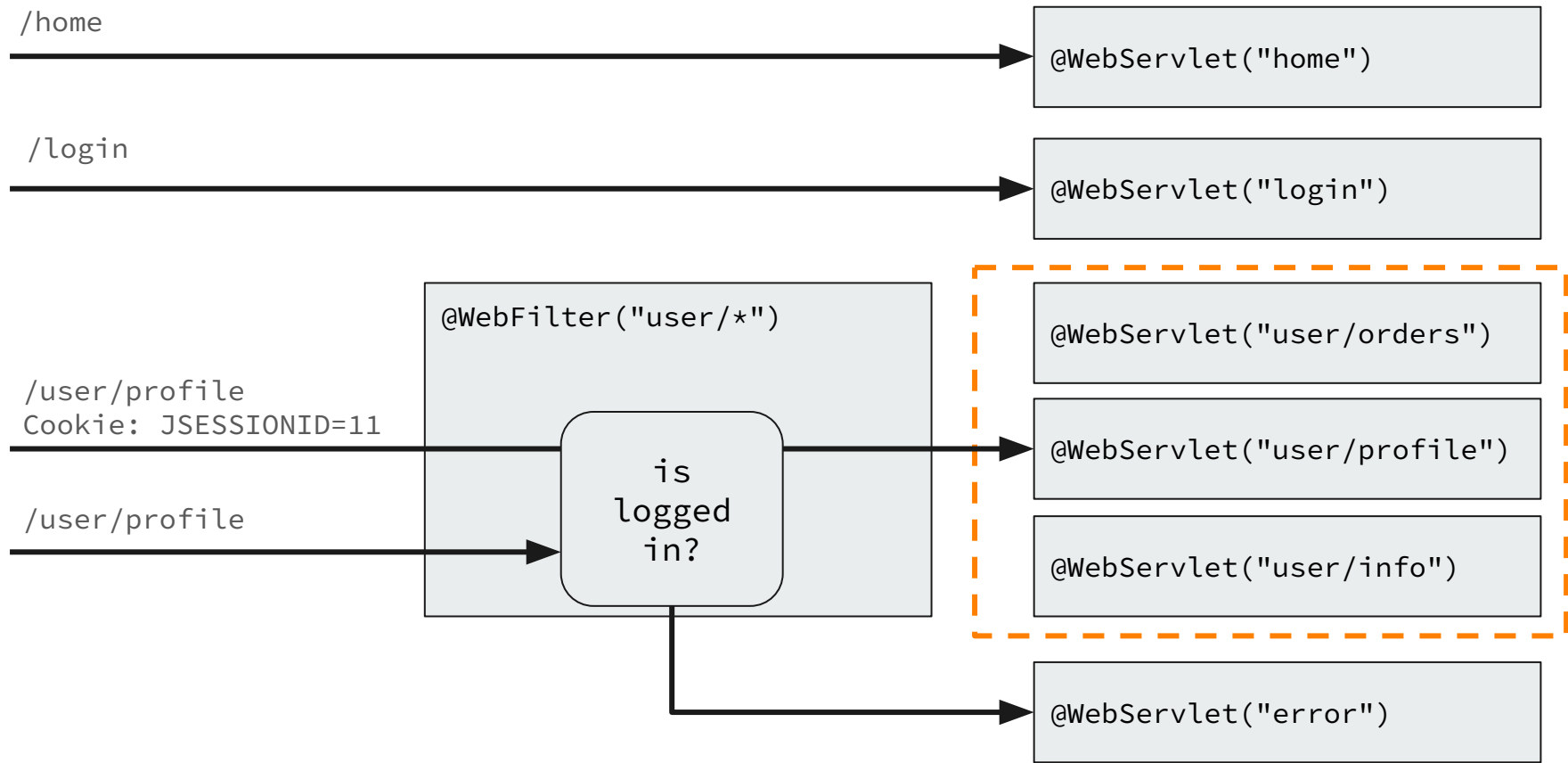
Scopes

request

session

application (ServletContext)

Filter





Documents

under /webapp

- static content (html, js, cs, jpg ...)
- dynamic content (jsp)



JSP

Document files like HTML

Accepts dynamic parts like scriptlets, expressions

Today's date: `<%= new java.util.Date()%>`
with access to implicit objects like: request, response,
session, application, out

Accepts placeholders with custom expressions (EL)

<code>\${param.name}</code>	<code>\${param["name"]}</code>
<code>\${header.Host}</code>	<code>\${requestScope.attr}</code>
<code>\${sessionScope.attr}</code>	<code>\${applicationScope.attr}</code>



JSP vs Servlet

```
<html>
<head><title>First JSP</title></head>
<body>
  <%
    double num = Math.random();
    if (num > 0.95) {
  %>
    <h2>You'll have a luck day!</h2><p>(<%= num %>)</p>
  <%
    } else {
  %>
    <h2>Well, life goes on ... </h2><p>(<%= num %>)</p>
  <%
    }
  %>
  <a href="<%= request.getRequestURI() %>"><h3>Try
Again</h3></a>
</body>
</html>
```

```
out.write("<html>\r\n ");
double num = Math.random();
if (num > 0.95) {
    out.write("<h2>You will have a luck day!");
    out.write("</h2><p>(");
    out.print( num );
    out.write(")</p>\r\n");
} else {
    out.write("\r\n ");
    out.write("<h2>Well, life goes on ... ");
    out.write("</h2><p>(");
    out.print( num );
    out.write(")</p>\r\n ");
}
out.write("<a href=\"");
out.print( request.getRequestURI() );
out.write("\">");
out.write("<h3>Try Again</h3></a>\r\n");
out.write("</html>\r\n");
```

MVC

