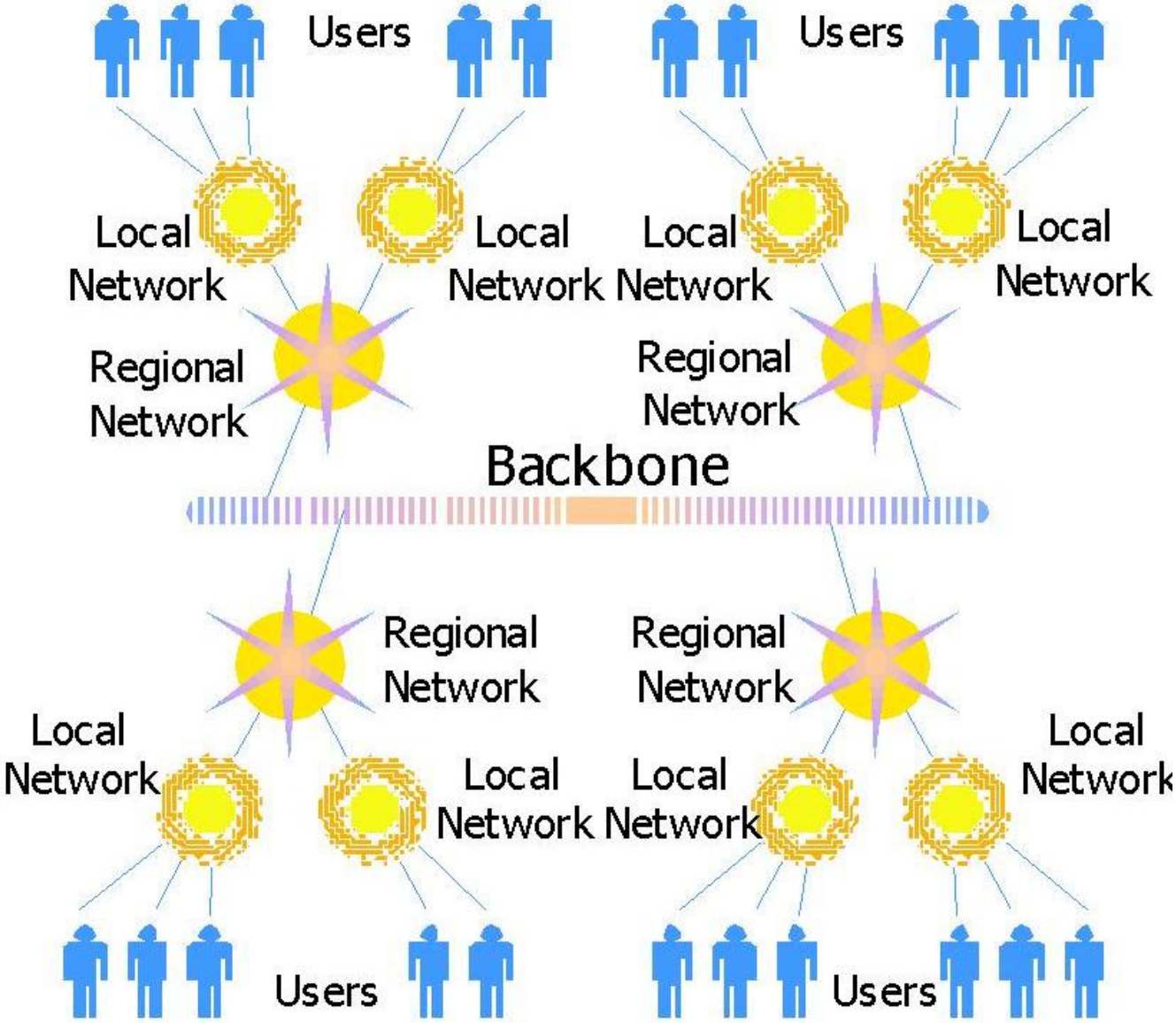
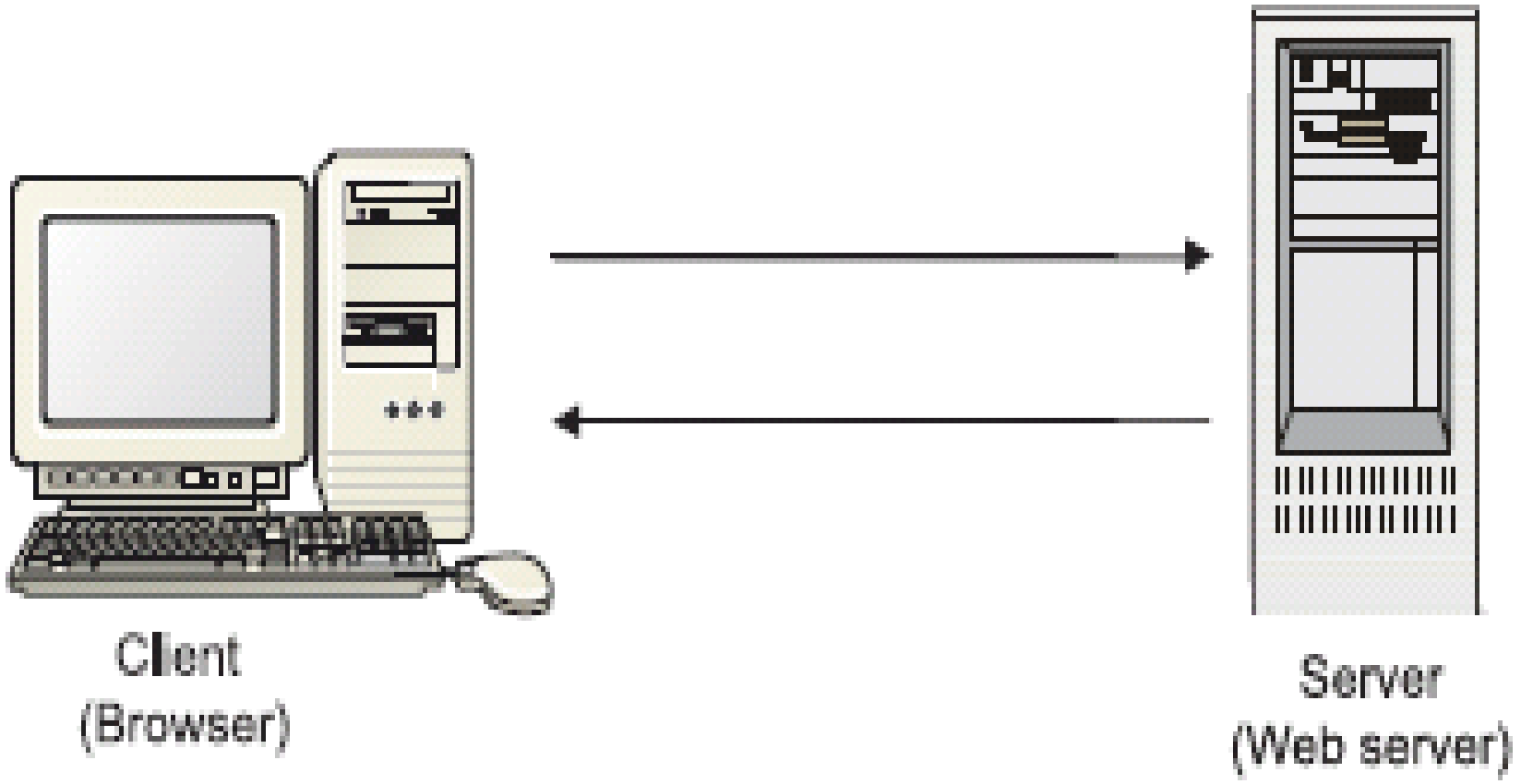
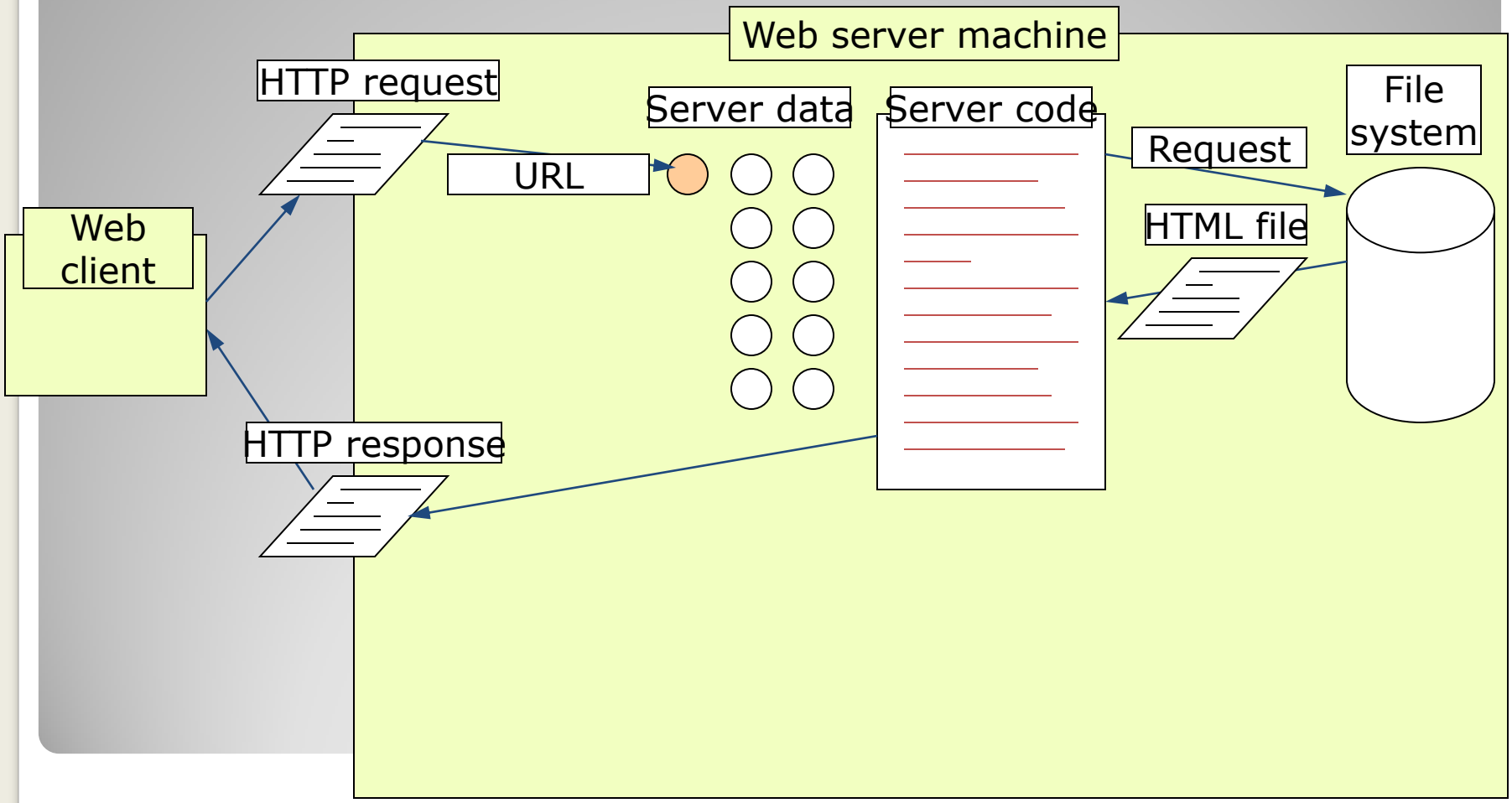
Web Programming


Users
Users
Local Network
Local Network
Local Network
Local Network
Regional Network
Regional Network
Backbone
Regional Network
Regional Network
Local Network
Local Network
Local Network
Local Network
Users
Users


Client
(Browser)
Server
(Web server)



"Lifecycle" of static web page

Bucky Badger's web page - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://www.cs.wisc.edu/~estan/examples/Bucky.html

Go

Welcome to Bucky's web page

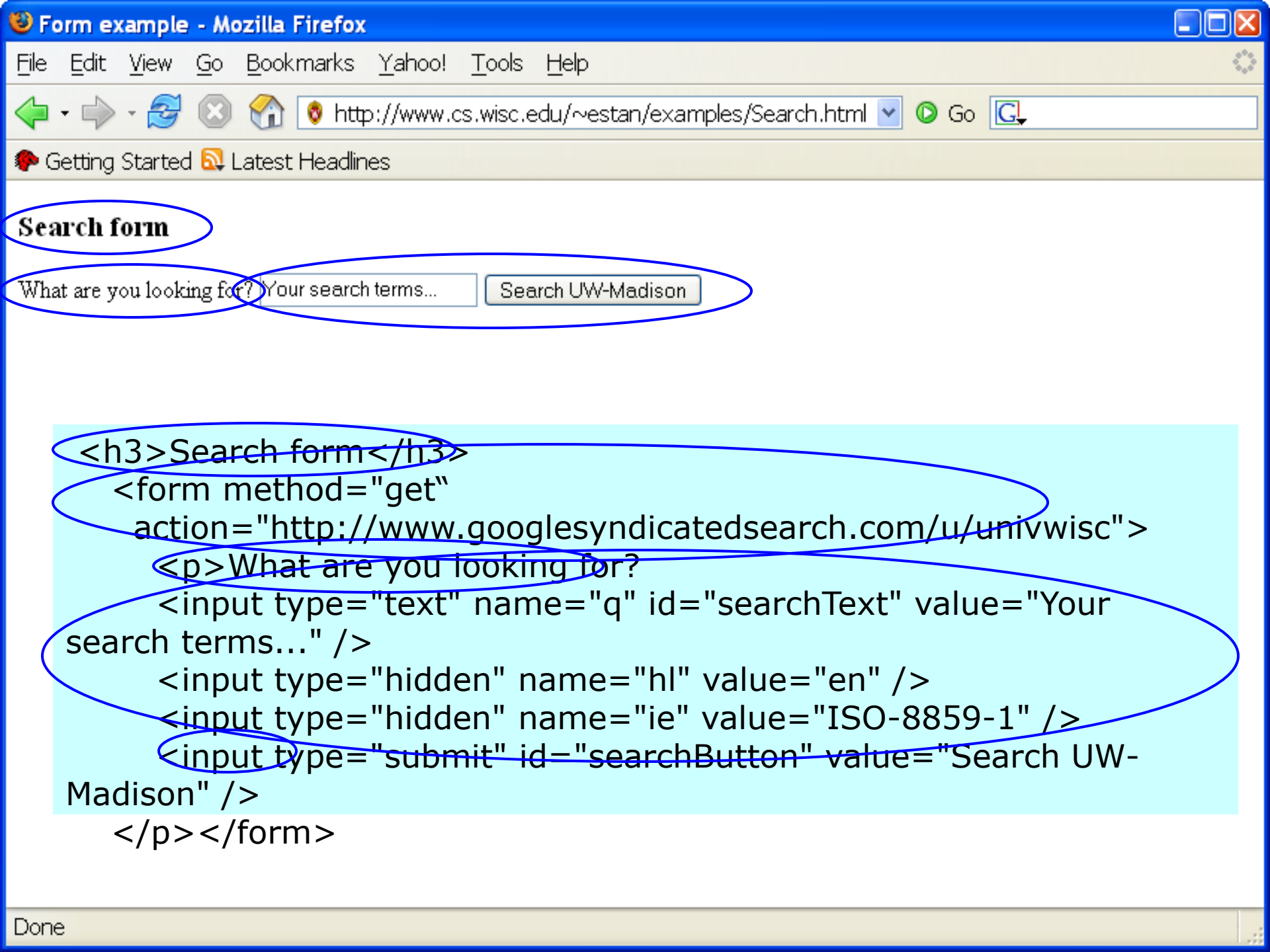
```
<TITLE>Bucky Badger's web page</TITLE>
<BODY>
<H1>Welcome to Bucky's web page</H1>
<IMG SRC="bucky.gif">
<P>I am Bucky, the mascot for University of
Wisconsin athletics. Please visit
<A
HREF="http://www.uwbadgers.com/football/index.ht
ml"> the web page of our football team</A>
and  <A
HREF="http://www.uwbadgers.com/basketball/index.
html"> the web page of our basketball team</A>.
</BODY>
```

I am Bucky, the mascot for University of Wisconsin athletics. Please visit the web page of our football team and the web page of our basketball team.

Done

About forms

- Forms are the traditional way for users to send information to a web server
  - The user fills out fields in the browser
  - The user submits the form
  - http carries the user input to the web server
  - A server side program processes the user data
  - The server sends a reply document to the client


Form example - Mozilla Firefox
File Edit View Go Bookmarks Yahoo! Tools Help
http://www.cs.wisc.edu/~estan/examples/Search.html
Go
Getting Started
Latest Headlines
Search form
What are you looking for?
Your search terms...
Search UW-Madison
<h3>Search form</h3>
<form method="get"
action="http://www.googlesyndicatedsearch.com/u/univwisc">
What are you looking for?
<input type="text" name="q" id="searchText" value="Your search terms..." />
<input type="hidden" name="hl" value="en" />
<input type="hidden" name="ie" value="ISO-8859-1" />
<input type="submit" id="searchButton" value="Search UW-Madison" />
</form>
Done

About tables

- Their original role was to display tables
- Their most prevalent use is for controlling the placement of visual elements on the page
  - http://www.cs.wisc.edu, http://www.google.com
  - Frames control placement too – don't use them
- The table is a collection of rows
- The rows are collections of cells
- Cells on the same row/column are aligned
- Cells can contain anything (even other tables)

The <table></table> tag

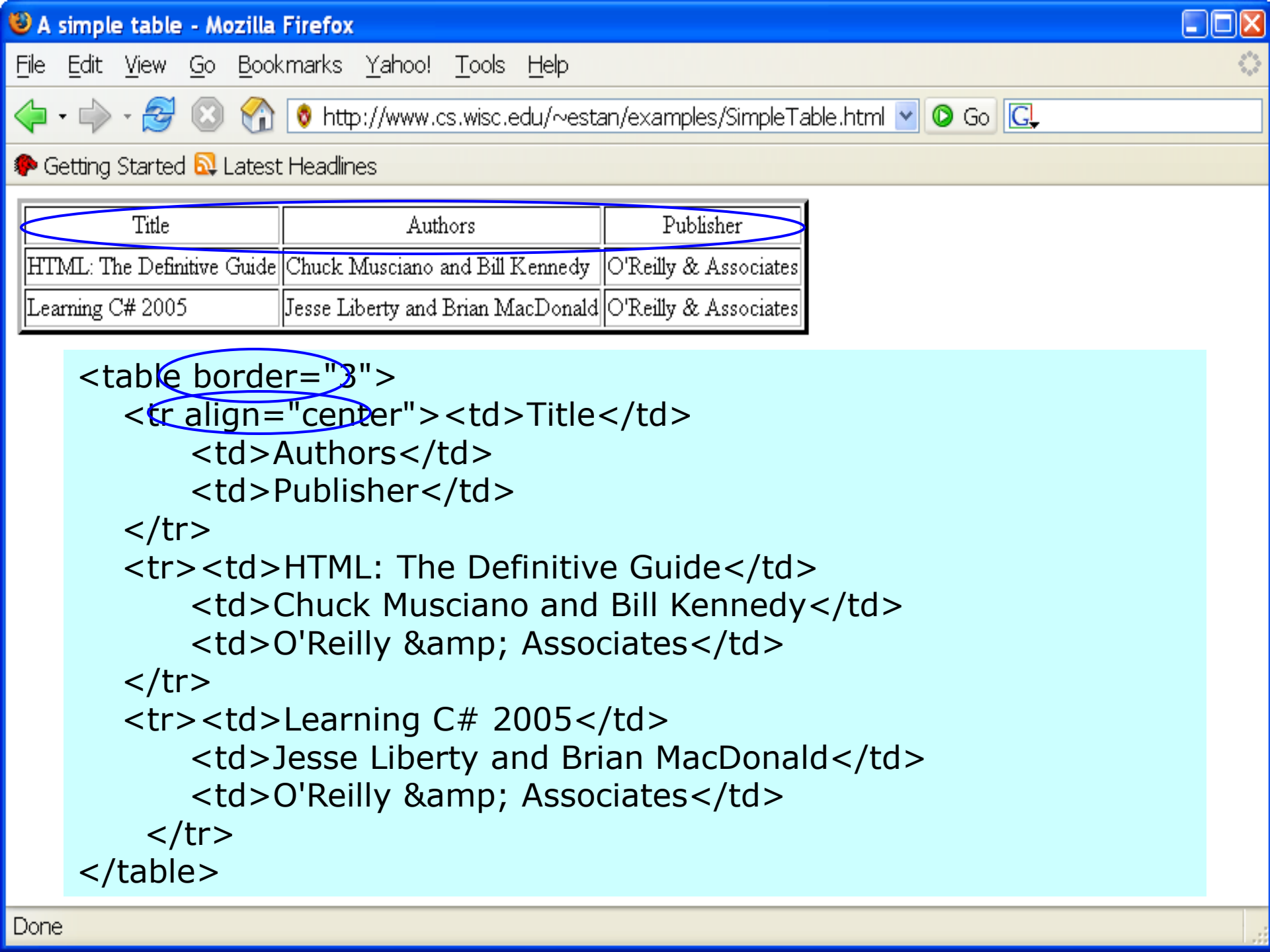
- Defines a table
- The “border” attribute defines the width of the lines used to draw the table (in pixels)
  - Defaults to 0 which means no lines are drawn
- The “width” attribute controls table width
  - By default it is in pixels
  - It can be given as a percentage of the window
  - If not specified, the table is only as wide as needed to display cell contents

The <tr></tr> tag

- Defines a table row
  - The “align” attribute controls horizontal alignment of text in cells – can be “left”, “right”, “center”
  - The “valign” attribute controls vertical alignment of text in cells – can be “top”, “bottom”, “middle”
  - The “nowrap” attribute instructs the browser not to wrap the text from within the cells

The <td></td> tag

- Defines a table cell
  - Has “align”, “valign” and “nowrap” attributes
  - “width” can be given as percentage of table width
  - “height” gives minimum height for cell
  - “colspan” allows a cell to span multiple columns
  - “rowspan” allows a cell to span multiple rows


A simple table - Mozilla Firefox
File Edit View Go Bookmarks Yahoo! Tools Help
http://www.cs.wisc.edu/~estan/examples/SimpleTable.html
Go
Getting Started
Latest Headlines
Title
Authors
Publisher
HTML: The Definitive Guide
Chuck Musciano and Bill Kennedy
O'Reilly & Associates
Learning C# 2005
Jesse Liberty and Brian MacDonald
O'Reilly & Associates
<table border="3">
<tr align="center"><td>Title</td>
<td>Authors</td>
<td>Publisher</td>
</tr>
<tr><td>HTML: The Definitive Guide</td>
<td>Chuck Musciano and Bill Kennedy</td>
<td>O'Reilly & Associates</td>
</tr>
<tr><td>Learning C# 2005</td>
<td>Jesse Liberty and Brian MacDonald</td>
<td>O'Reilly & Associates</td>
</tr>
</table>
Done

Example of a complex table - Mozilla Firefox

File Edit View Go Bookmarks Yahoo! Tools Help

http://www.cs.wisc.edu/~estan/examples/ComplexTable.html

Getting Started Latest Headlines

Key Department Contacts

You can contact the people listed here.

Prof. Gurindar Sohi
Chair

Prof. Susan Horwitz
Associate Chair

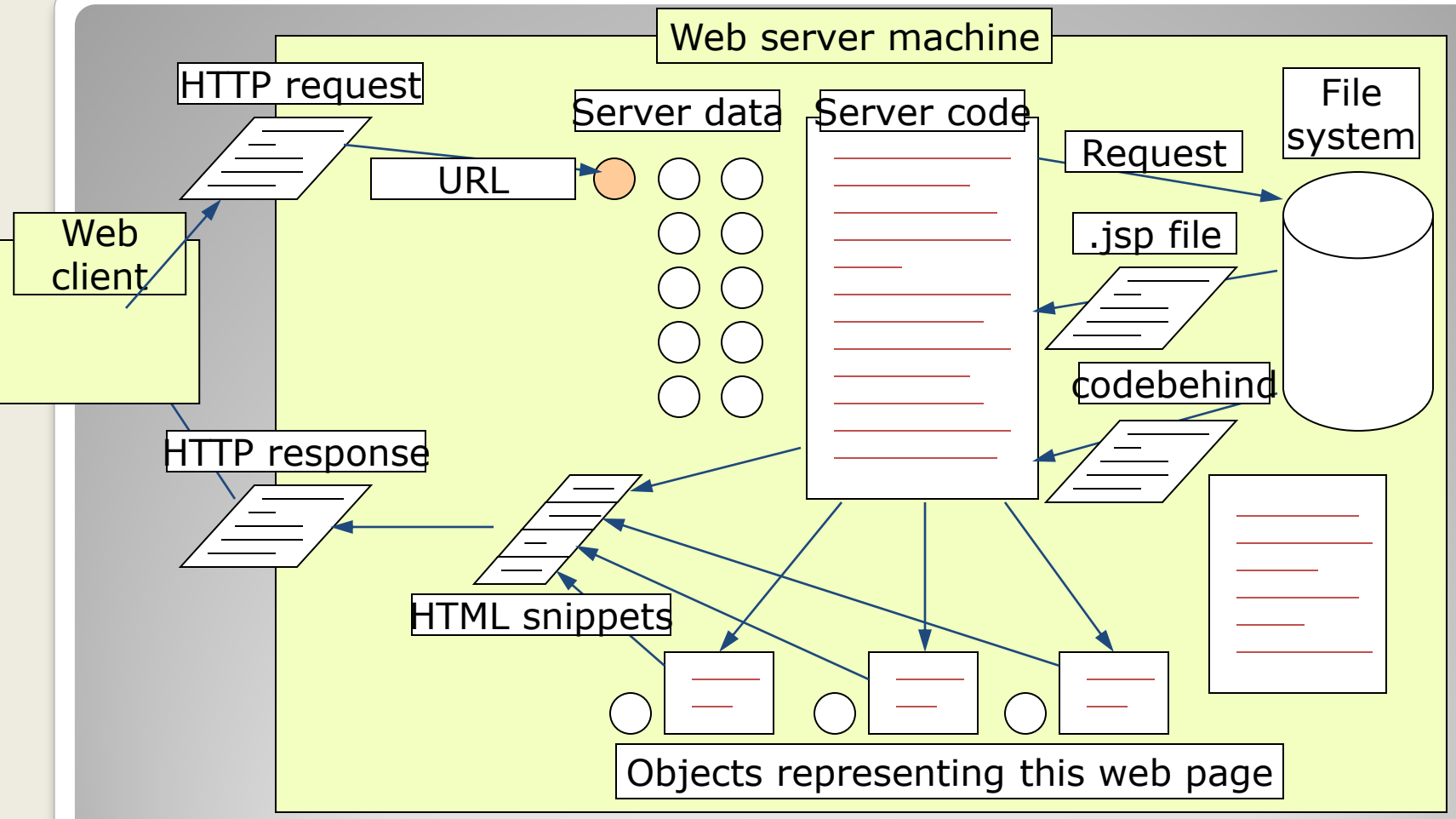
There are a few others we should have listed.

Done

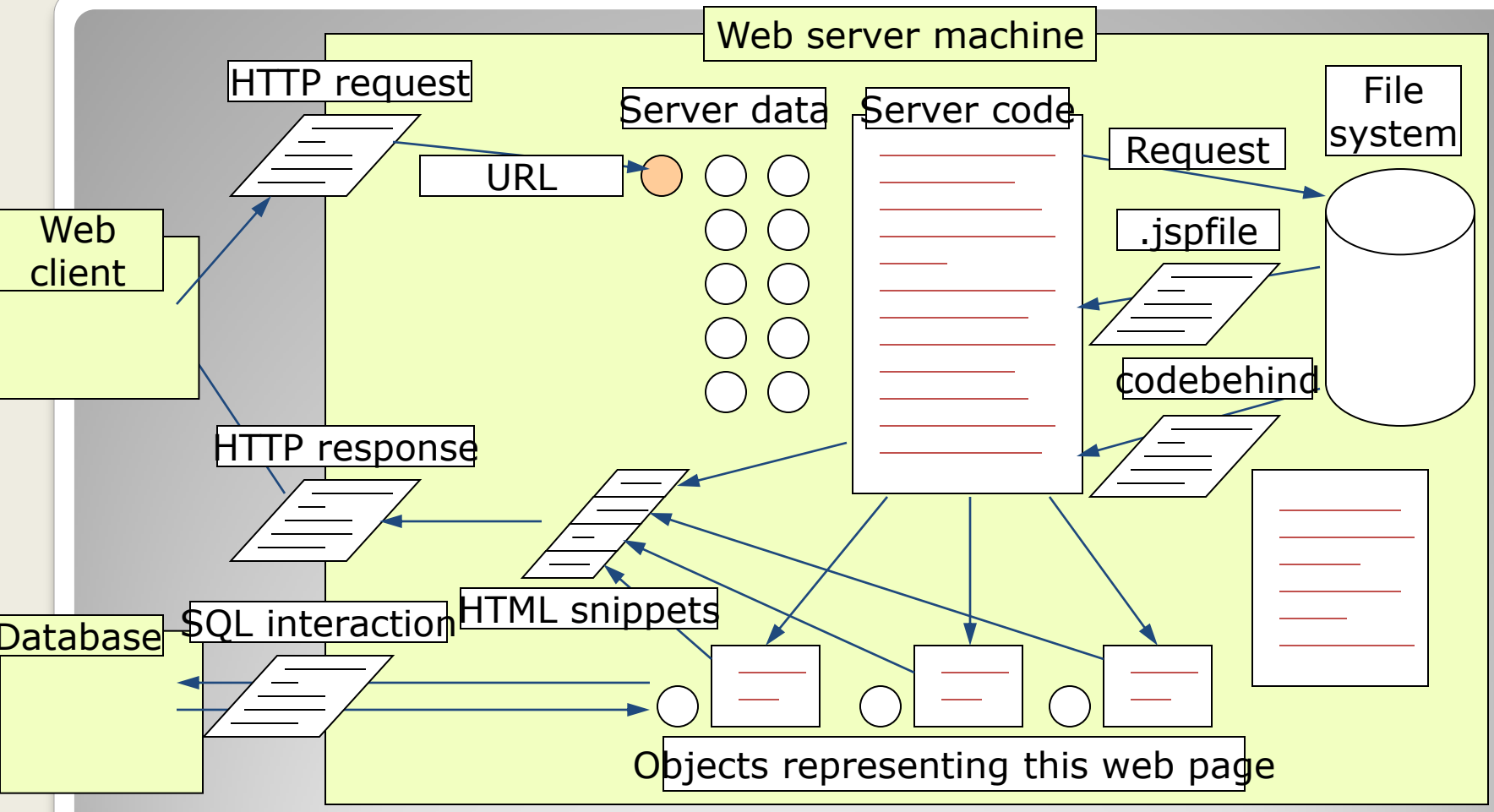
```
<table width="300">
   <tr valign="top">
      <td rowspan="2" valign="middle">
         <i>You can contact the people listed
here.</i></td>
      <td width="140">
         <img src="sohi-mini.jpg" />
         <br /><b><a href="/~sohi/">Prof. Gurindar Sohi
            </a></b>
         <br />Chair</td>
      <td width="140">
         <img src="horwitz-mini.jpg" />
         <br /><b><a href="/~horwitz/">Prof. Susan
Horwitz
            </a></b>
         <br />Associate Chair</td>
   </tr>
   <tr><td colspan="2" align="center">
         <i>There are a few others we should have
listed.</i>
         </td>
   </tr>
</table>
```

HTML Tutorial

- Basics of HTML
  - http://www.w3schools.com/html/html_intro.asp
  - Do all but skip: **Stylesheets OR CSS, Frames and Iframes**
- More on forms
  - http://www.tizag.com/htmlT/forms.php



Lifecycle of a dynamic webpage - Server-side programming



Page with database interaction

Java Server Pages

- JavaServer Pages (JSP) is a technology based on the Java language and enables the development of dynamic web sites
- JSP was developed by Sun Microsystems to allow server side development
- JSP files are HTML files with special Tags containing Java source code that provide the dynamic content.

History

- Static web pages: first experience with making web pages
- CGI server side applications: Common Gateway Interface
  - Return either files or method output
- A Servlet is a Java class that provides special server side service
  - In Servlets you need to have lots of println statements to generate HTML

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class HelloClientServlet extends HttpServlet
{
  protected void doGet(HttpServletRequest req,
                       HttpServletResponse res)
            throws ServletException, IOException
  {
    res.setContentType("text/html");
    PrintWriter out = res.getWriter();
    out.println("<HTML><HEAD><TITLE>Hello Client!</TITLE>"+
                "</HEAD><BODY>Hello Client!</BODY></HTML>");
    out.close();
  }

  public String getServletInfo()
  {
    return "HelloClientServlet 1.0 by Stefan Zeiger";
  }
}
```

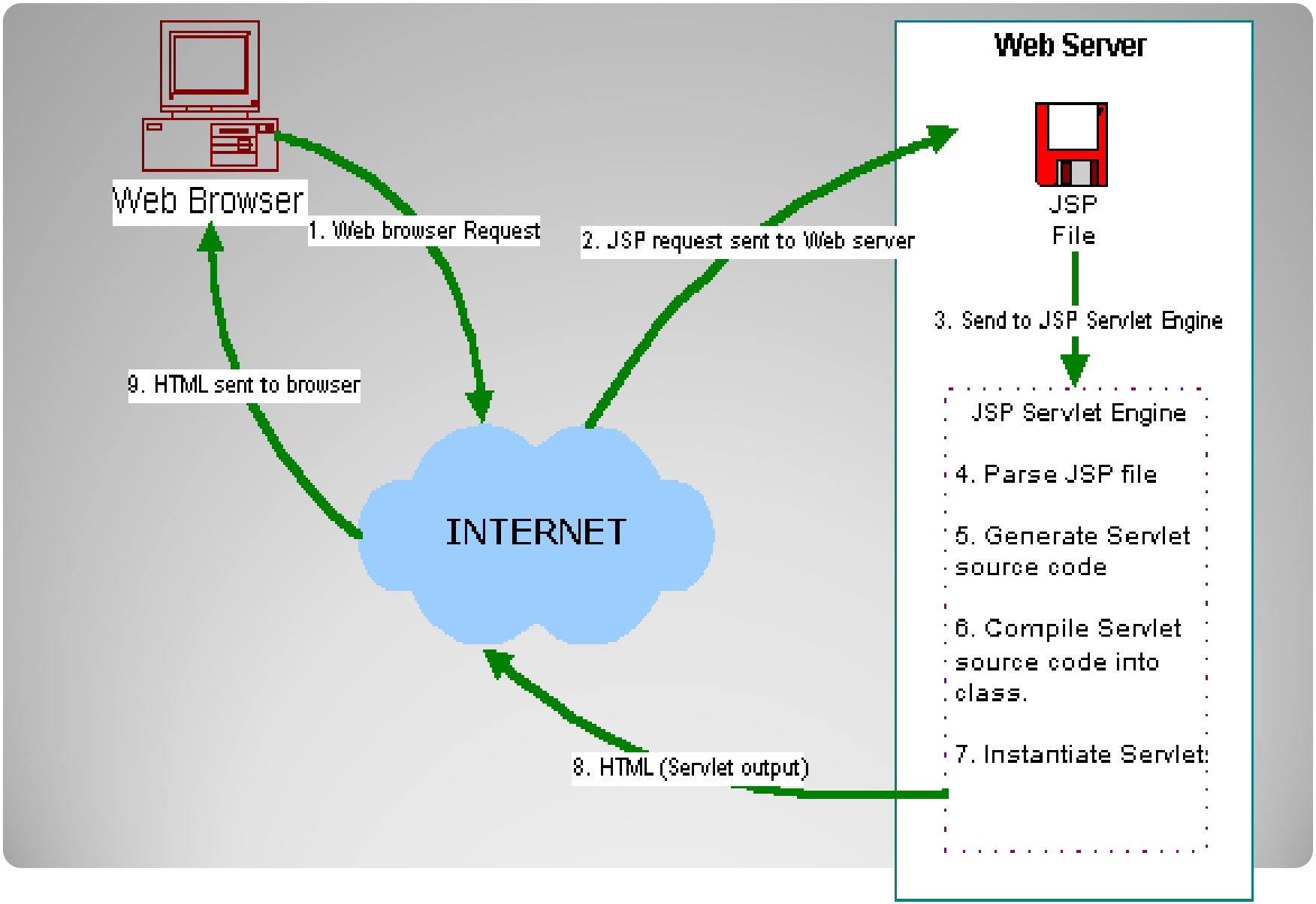
HelloClientServlet.java

JSP: Java Server Pages

- ASP was developed by Microsoft to allow HTML developers to easily provide dynamic content
  - supported as standard by Microsoft's free Web Server, Internet Information Server (IIS)
  - JSP is the equivalent from Sun Microsystems
- JSP pages are converted to Servlets so actually can do the same thing as old Java Servlets.
- See JSP Example on next slide

```
<HTML>
<HEAD>
<TITLE> JSP loop</TITLE>
</HEAD>
<BODY>
<font face=verdana color=darkblue>
JSP loop
<BR> <BR>
<%
        public String writeThis(int x) {
            String myText="";
            for (int i = 1; i <= x; i++ )
             myText += "<h1> This line is shown " +  i + " of " + x + " times in a JSP.</h1>";
                  return myText;
  }
%>
This is a loop example from the
<br>
<%= writeThis(8) %>
</font>
</BODY>
</HTML>
```

JSP Example


Web Server
Web Browser
1. Web browser Request
2. JSP request sent to Web server
JSP
File
3. Send to JSP Servlet Engine
9. HTML sent to browser
INTERNET
JSP Servlet Engine
4. Parse JSP file
5. Generate Servlet source code
6. Compile Servlet source code into class.
7. Instantiate Servlet:
8. HTML (Servlet output)

Steps required for a JSP request

- The user goes to a web site made using JSP.
  - The user goes to a JSP page (ending with .jsp).
  - The web browser makes the request via the Internet.
- The JSP request gets sent to the Web server.
- The Web server recognizes that the file required is special (.jsp),therefore passes the JSP file to the JSP Servlet Engine.
- If the JSP file has been called the first time, the JSP file is parsed, otherwise go to step 7.
- The next step is to generate a special Servlet from the JSP file. All the HTML required is converted to println statements.
- The Servlet source code is compiled into a class.
- The Servlet is instantiated,calling the init and service methods.
- HTML from the Servlet output is sent via the Internet.
- HTML results are displayed on the user's web browser.

- JSP Tutorial
  - http://java.sun.com/products/jsp/tags/11/syntaxref11.html
  - http://www.jsptut.com/

JSP Tutorial

JSP tags

- JSP scriplets:
  - `<% … %>`
  - Each java line followed by “;” as usual
- JSP expression values:
  - `<%= … %>`
- JSP comments:
  - `<%-- … --%>`
- JSP “static” declarations
  - Declared methods and variables shared by ALL page threads
  - `<%! … %>`

JSP directives

- JSP page directive
  - To import java packages, extend packages, etc …
  - `<%@ page language="java" import="java.sql.* errorPage="error.jsp" %>`
  - `<%@ page language="java" isErrorPage="true" %>`

JSP: Request Object

- **Contains info about request made to this page --- focus on the ones highlighted on RED**
- **`getParameter(String param)`**
  - Returns the value of the specified parameter as a string if it exists or null if it doesn't
- **`getParameterValues(String param)`**
  - Returns an array of String objects containing all of the values that the given request parameter has or null if the parameter doesn't have any values
- **`getParameterNames()`**
  - Returns an Enumeration object that contains the names of all the parameters contained in the request

JSP: Response Object

- **Contains info about response generated by this page--- focus on the ones highlighted on RED**
  - `setContentType()`
    - `response.setContentType("text/html");`
  - `addCookie(Cookie cookie)`
    - `response.addCookie(Cookie exforsys);`
  - `sendRedirect(String)`
    - `response.sendRedirect("http://???.???.com");`

Session Variables

- A website visitor might visit several pages and perform several interactions
- It is very helpful to be able to associate some data with each visitor
- Can set a session variable:
  - `<% String nameValue =???;`
  - `session.setAttribute( "theName", nameValue );%>`
- Access a session variable
  - `<%= session.getAttribute( "theName" ) %>`

JSP directives

- JSP Include Directive
  - Includes a static file in a JSP file, parsing the file's JSP elements
  - **`<%@ include file="date.jsp" %>`**
    - Including a response.sendRedirect in an included file might cause problems, sometimes, if were are doing some work on the output stream after the response header is sent back.
    - After the **response.sendRedirect("...")** add a return statement (i.e. **return;**).
    - The return will "stop" the processing of the rest of the JSP page, preventing any further manipulation of the output stream by the jsp to occur.