



JSP技术概述

JSP, Servlet, & Struts Training Courses: <http://courses.coreservlets.com>
Available in US, China, Taiwan, HK, and Worldwide

JSP and Servlet Books from Sun Press: <http://www.coreservlets.com>
*Available in English, Chinese (simplified and traditional script),
and 12 other languages*

2

议程

- 对JSP技术的需求
- 评估JSP的好处
- JSP相对于竞争技术的优势
- 对JSP的误解
- JSP的生命周期
- JSP页面的安装
- 实际应用中的JSP

3

对JSP的需求

- **使用servlet可以容易地完成下述任务：**
 - 读取表单数据
 - 读取HTTP请求报头
 - 设置HTTP状态代码和响应报头
 - 使用cookie以及进行会话跟踪
 - 跨servlet共享数据
 - 跨请求记录数据
 - 获得有趣且报酬丰厚的工作
- **但使用servlet也有不尽如人意的地方：**
 - 使用println语句生成HTML
 - 维护上述的HTML

4

JSP/servlet/Struts/JSP training: <http://www.coreservlets.com>

JSP框架

- **思想：**
 - 大部分的页面使用常规的HTML
 - 用特殊的标签将servlet代码标记出来
 - 整个JSP页面最终转换成servlet（仅执行一次），实际被调用的是servlet（每个请求）

- **示例：**

```
<!DOCTYPE ...>
<HTML>
<HEAD>
<TITLE>Order Confirmation</TITLE>
<LINK REL=STYLESHEET
      HREF="JSP-Styles.css"
      TYPE="text/css">
</HEAD>
<BODY>
<H2>Order Confirmation</H2>
Thanks for ordering
<I><%= request.getParameter("title") %></I>!
</BODY></HTML>
```

5

JSP/servlet/Struts/JSP training: <http://www.coreservlets.com>

JSP的好处

- 尽管JSP技术能够做的事情并不比servlet多，但JSP可以使下述任务更为容易：
 - 输出HTML
 - 阅读和维护这些HTML
- JSP使得下面的做法成为可能：
 - 使用标准的HTML工具，比如Macromedia DreamWeaver或Adobe GoLive。
 - 由小组的另外一些成员完成HTML布局，不关注Java编程。
- JSP鼓励我们：
 - 将创建内容（Java）代码同表示内容的（HTML）代码分离开来。

6

JSP/servlet/Struts/JSP training: <http://www.coreservlets.com>

JSP相对于竞争技术的优势

- 与ASP或ColdFusion相比
 - 对于动态部分来讲，是一种更好的语言
 - 可以移植到多种不同的服务器和操作系统
- 与PHP相比
 - 对于动态部分来讲，是一种更好的语言
 - 更好的工具支持
- 与纯servlet相比
 - 创建HTML更方便
 - 可以使用标准的工具（例如DreamWeaver）
 - 分而治之
 - JSP程序员依旧需要了解servlet编程

7

JSP/servlet/Struts/JSP training: <http://www.coreservlets.com>

JSP的好处 (续)

- **与Velocity或WebMacro相比**
 - 标准
- **与客户端JavaScript(在浏览器中)相比**
 - 能力大部分不与JSP交错，但是
 - 我们控制服务器，不是客户程序
 - 语言更强大
- **与服务器端JavaScript (如LiveWire , BroadVision) 相比**
 - 语言更强大
- **与静态HTML相比**
 - 提供动态特性
 - 添加动态特性不再需要在要么全部动态生成，要么不使用动态特性之间做出抉择

8

JSP/servlet/Struts/JSP training: <http://www.coreservlets.com>

开发环境的设置

- **设置CLASSPATH。不需要**
- **编译代码。不需要**
- **使用包来避免命名冲突。不需要**
- **将JSP页面放在特殊的目录中。不需要**
 - *install_dir*\webapps\ROOT\ (HTML和JSP -- Tomcat)
 - *install_dir*\servers\default\default-app (JRun)
- **使用专门的URL来调用JSP页面。不需要**
 - 使用和HTML页面相同的URL (除文件扩展名以外)
- **警告**
 - 前面介绍的有关CLASSPATH，安装目录等的规则依旧适用于JSP页面使用的常规Java类。

9

JSP/servlet/Struts/JSP training: <http://www.coreservlets.com>

示例

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
<TITLE>JSP Expressions</TITLE>
<META NAME="keywords"
      CONTENT="JSP, expressions, JavaServer Pages">
<META NAME="description"
      CONTENT="A quick example of JSP expressions.">
<LINK REL=STYLESHEET
      HREF="JSP-Styles.css"
      TYPE="text/css">
</HEAD>
```

10

JSP/servlet/Struts/JSF training: <http://www.coreservlets.com>

示例 (续)

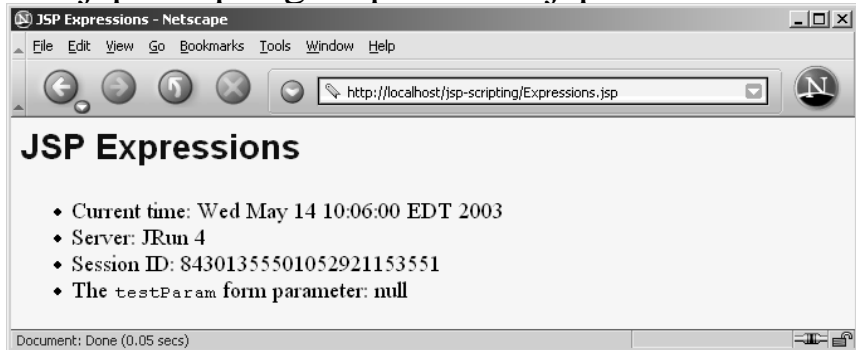
```
<BODY>
<H2>JSP Expressions</H2>
<UL>
  <LI>Current time: <%= new java.util.Date() %>
  <LI>Server: <%= application.getServerInfo() %>
  <LI>Session ID: <%= session.getId() %>
  <LI>The <CODE>testParam</CODE> form parameter:
      <%= request.getParameter("testParam") %>
</UL>
</BODY></HTML>
```

11

JSP/servlet/Struts/JSF training: <http://www.coreservlets.com>

示例 (结果)

- 如果文件所在位置为 :
 - C:\jakarta-tomcat-xx\webapps\ROOT\jsp-scripting\Expressions.jsp 或
 - C:\JRun4\servers\default\default-ear\default-war\jsp-scripting\Expressions.jsp
- 对应的URL将会是 :
 - <http://localhost/jsp-scripting/Expressions.jsp>



最常见的误解 忘记JSP是服务器端技术

- 最常见的问题
 - 我不能使用HTML来完成这个那个功能，使用JSP能够做到吗？
- 为什么这个问题没有意义？
 - JSP完全在服务器上运行
 - 它并不会改变客户程序（浏览器）所能够处理的内容
- 类似的问题
 - 如何将普通的applet放在JSP页面中？
答案：向客户程序发送<APPLET...>标签
 - 如何将图像放到JSP页面中？
答案：向客户端发送标签
 - 如何使用JavaScript/Acrobat/Shockwave/Etc？
答案：发送适当的HTML标签

次一级的最常见误解 转换期间与请求期间的混淆

- 在页面转换期间发生什么呢？
 - JSP构造被转换成servlet代码。
- 在请求期间发生什么呢？
 - servlet代码得到执行。在请求期间不涉及JSP的解释。在请求期间，完全不会理会初始的JSP页面；所使用的只是由它所产生的servlet。
- 页面的转换发生在什么时候？
 - 典型地，在JSP页面被修改后首次被访问时进行转换。但实际的用户不会遇到这种情况（开发人员应该测试他们安装的所有JSP页面。）
 - 页面转换不是每次请求都执行。

14

JSP/servlet/Struts/JSP training: <http://www.coreservlets.com>

JSP的生命周期

		请求 #1	请求 #2		请求 #3	请求 #4		请求 #5	请求 #6
JSP页面转换成servlet	页面首次编写完成	Yes	No	服务器重启	No	No	页面被修改	Yes	No
servlet编译		Yes	No		No	No		Yes	No
创建servlet实例并载入服务器的内存		Yes	No		Yes	No		Yes	No
init (或类似的函数) 被调用		Yes	No		Yes	No		Yes	No
doGet (或类似的函数) 被调用		Yes	Yes		Yes	Yes		Yes	Yes

15

JSP/servlet/Struts/JSP training: <http://www.coreservlets.com>

JSP/Servlet在实际中的应用

- ofoto.com : 打印和管理数码和传统照片。

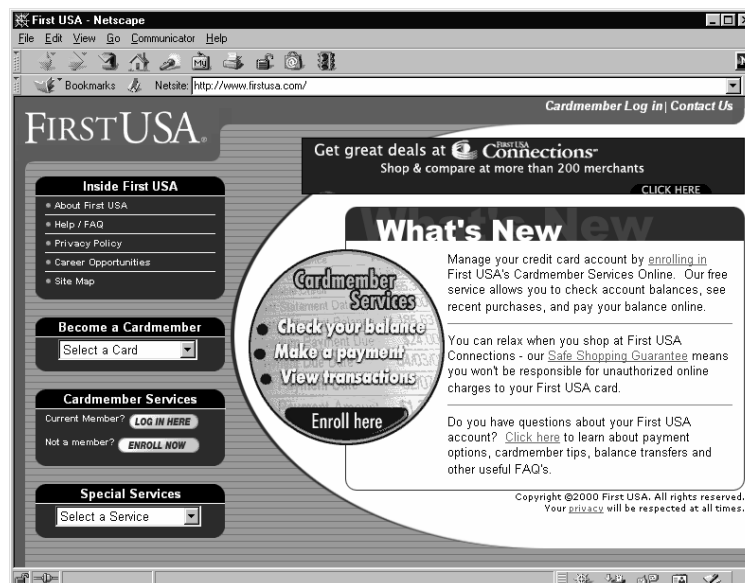


16

JSP/Servlet/Struts/JSP training: <http://www.coreservlets.com>

JSP/Servlet在实际中的应用

- First USA Bank : 世界最大的信用卡发行商 ; 拥有数量最多的在线银行业客户。

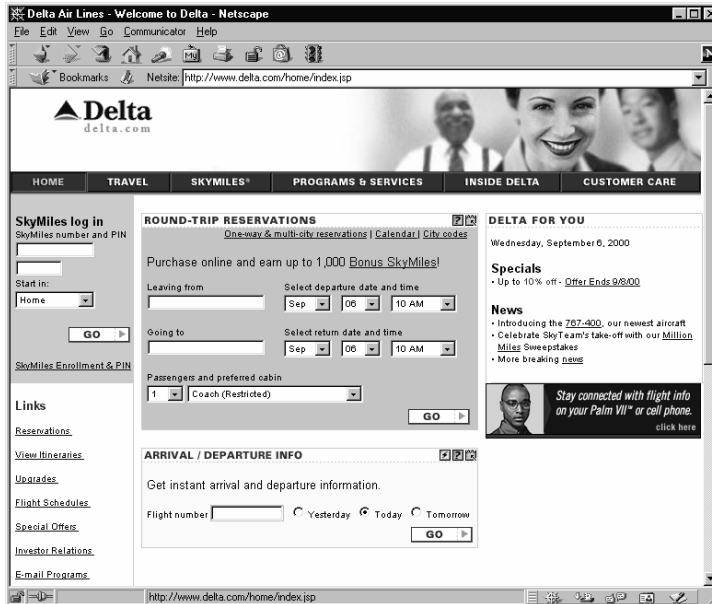


17

JSP/Servlet/Struts/JSP training: <http://www.coreservlets.com>

JSP/Servlet在实际中的应用

- Delta Airlines : 整个网站，包括实时的时间表信息。



18

JSP/Servlet/SUTUS/JSP training: <http://www.coreservlets.com>

JSP/Servlet在实际中的应用

- American Century Investments: 管理超过70种共同基金，900亿美元，2百万投资者

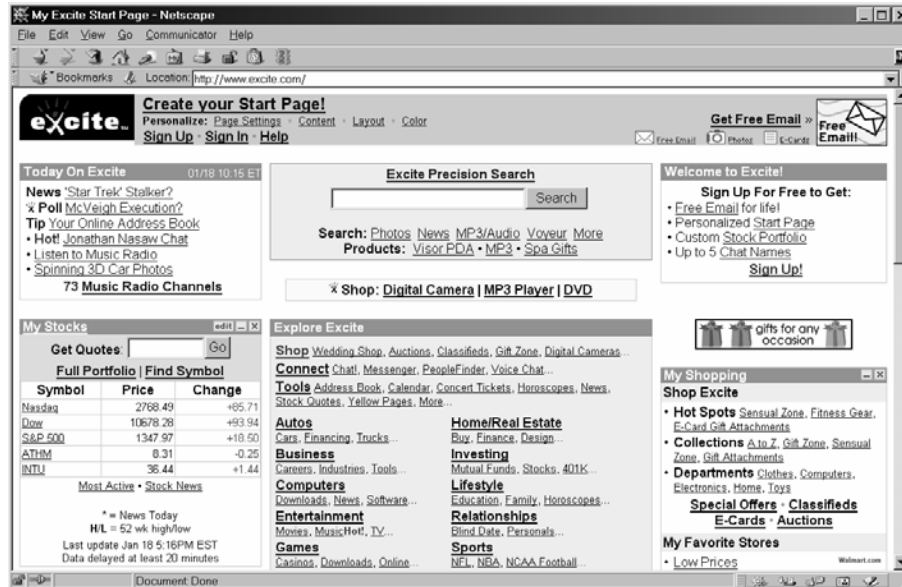


19

JSP/Servlet/SUTUS/JSP training: <http://www.coreservlets.com>

JSP/Servlet在实际中的应用

- **Excite** : 全球前五位的互联网门户网站 ; 网络上最繁忙的十个网站之一。



20

JSP/servlet/Struts/JSP training: <http://www.coreservlets.com>

小结

- **JSP使得创建和维护HTML更容易，同时依旧提供对servlet的完全访问（即可以使用各种servlet的特性）。**
- **JSP页面被转换成servlet**
 - 在请求期间运行的是转换得来的servlet
 - 客户不会看到与JSP相关的任何内容
- **我们依旧需要了解servlet**
 - 理解JSP如何工作
 - JSP调用的servlet代码
 - 知道什么时候使用servlet更佳
 - 如何混合使用servlet和JSP
- **其他技术使用类似的方式，但移植性上有差距并且不允许我们使用Java来编写实际的代码**

21

JSP/servlet/Struts/JSP training: <http://www.coreservlets.com>



问题？

JSP, Servlet, & Struts Training Courses: <http://courses.coreservlets.com>
Available in US, China, Taiwan, HK, and Worldwide

JSP and Servlet Books from Sun Press: <http://www.coreservlets.com>
*Available in English, Chinese (simplified and traditional script),
and 12 other languages*