

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)

	<p>Tutorial desarrollado por: Roberto Canales Mora 2003-2004.</p> <p>Si te gusta lo que ves, puedes contratarme para impartir cursos presenciales en tu empresa o para ayudarte en proyectos (Madrid).</p> <p>Estamos creando las bases de la empresa en la que seguro te gustaría trabajar ... ayudanos a hacerla crecer ... presentándonos a tus jefes</p> <p>Contacta: rcanales@autentia.com.</p>	
---	---	---

Una vez escuche esta frase:

Cuando todo esta bajo control, es que no vamos suficientemente deprisa

La mayoría de las veces construimos estos tutoriales al mismo tiempo que recordamos como hacer cosas o probamos nuevas tecnologías, poniendo más celo en el contenido que en la forma (cosa que veo que perturba a muchos de los lectores por lo que os pido disculpas). Es cuestión de criterios aunque trataremos de mejorar.

Espero que seáis comprensivos y generosos ayudando a encontrar las erratas, faltas de ortografía, enlaces rotos u otros posibles errores en estas páginas. Escribidme siempre que encontréis algún error y pronto ya no habrá y os estaremos francamente agradecidos. Este es tu Web colabora libremente ...

[Roberto Canales](#)

Descargar este documento en formato PDF [strutstiles.pdf](#)



Uso de plantillas (tiles) en Struts

Cuando construimos un Web, podemos considerar que éste está formado por una o varias plantillas en la que únicamente cada página cambia una porción (normalmente central) del contenido mostrado respecto a las restantes. La plantilla normalmente contiene un menú, publicidad, cabeceras, pies, etc.

Si os fijáis con detenimiento, nuestro Web www.adictosaltrabajo.com está construido así (aunque la tecnología es PHP básico). De este modo, para hacer crecer el Web solo necesitamos ir añadiendo nuevas porciones. Además, cambiar todo el L&F sería relativamente rápido (si alguno os animáis a colaborar, ya sabéis donde contactar).

Cuando utilizamos editores gráficos avanzados éstos nos permiten construir Webs con esta filosofía. Si posteriormente se quiere sustituir una porción del site (por ejemplo la cabecera) solo se toca en un sitio y el cambio se extiende a todo el sistema. El único problema puede ser que con esta filosofía los creativos pueden construir y mantener de un modo sencillo ramas estáticas de un Web pero en ramas dinámicas necesitamos reproducir este comportamiento a través de código.

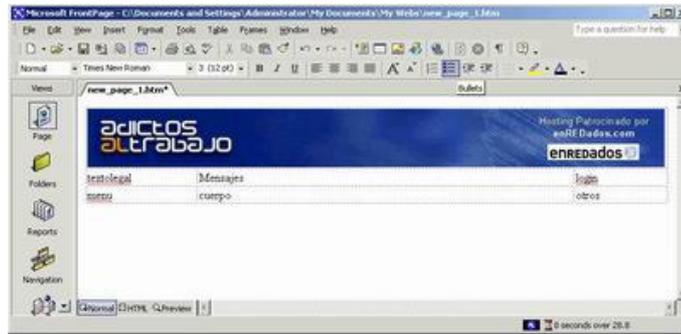
Hoy vamos a ver como se puede hacer esto con uno de los FrameWorks más extendidos en Java ([Struts](#)) y utilizando nuestro entorno de desarrollo habitual (NetBeans). Si aprendemos como lo hace este FrameWork podremos tener criterios comparativos cuando diseñemos en nuestro o tengamos que decidir que parte utilizamos y cual ignoramos o reemplazamos.

Definición de la plantilla

La construcción de páginas en base a plantillas se basa en la construcción de tres elementos:

- La plantilla que define elementos estáticos y áreas donde insertar otros contenidos
- La página real que utiliza la plantilla insertando textos y otras porciones dinámicamente.
- Las porciones particulares reutilizables (que pueden ser otros JSPs)

En un editor gráfico podemos construir nuestra plantilla basándonos en tablas (con bordes ocultos). A mi me gusta MS FrontPage (como un comienzo) porque es muy intuitivo. Los diseñadores gráficos suelen preferir herramientas más evolucionadas del tipo DreamWeaver.



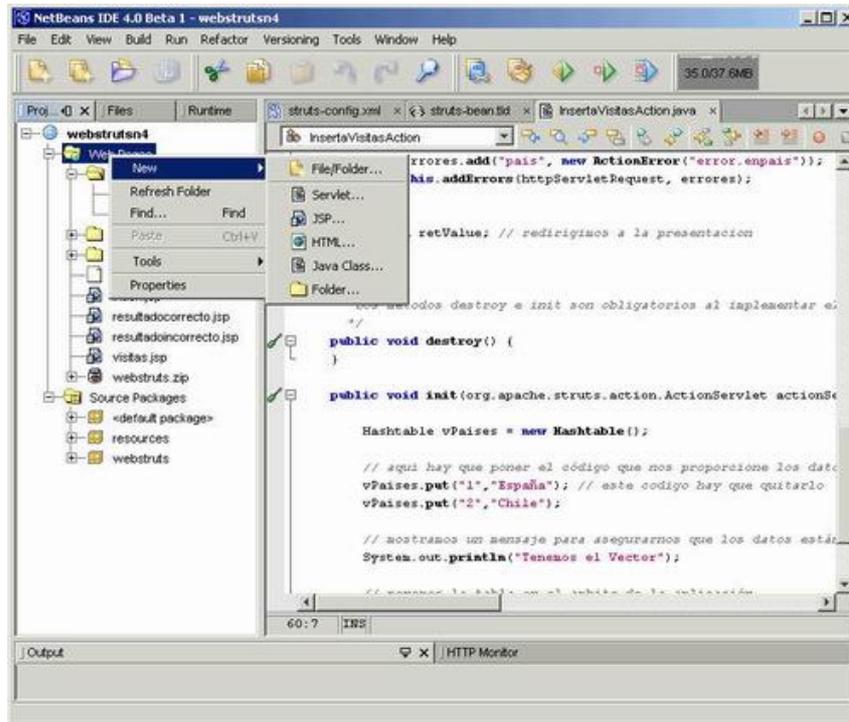
Lo que realmente interesa es el HTML (en esta etapa). Hemos marcado en rojo los huecos donde debemos insertar el código particular.

```
<html>
<head>
<title>Adictosaltrabajo.com</title>
</head>
<body>
<div align="center">
<center>
<table border="1" cellpadding="2" style="border-collapse:
collapse; border-width: 0" bordercolor="#111111" id="AutoNumber1" width="756">
<tr>
<td width="773" style="border-style: none; border-width: medium" colspan="3">
<a href="http://www.adictosaltrabajo.com/">

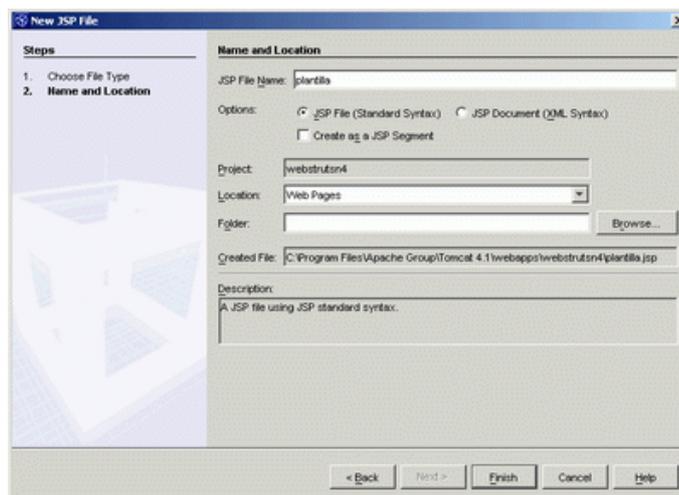
</a><a href="http://www.enredados.com/click.php">

</a></td>
</tr>
<tr>
<td width="140" style="border-style: none; border-width: medium">
textolegal</td>
<td width="508" style="border-style: none; border-width: medium">Mensajes</td>
<td width="117" style="border-style: none; border-width: medium">login</td>
</tr>
<tr>
<td width="140" style="border-style: none; border-width: medium">menu</td>
<td width="508" style="border-style: none; border-width: medium">cuerpo</td>
<td width="117" style="border-style: none; border-width: medium">otros</td>
</tr>
</table>
</center>
</div>
</body>
</html>
```

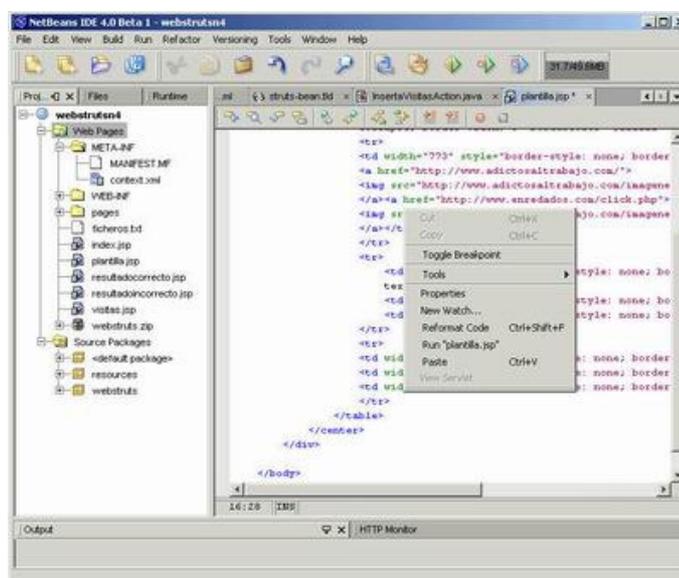
Creamos un JSP que será la plantilla en la aplicación Struts. La generación de este JSP la podemos hacer a mano en nuestro proyecto o a través de los asistentes gráficos (os lo recomiendo en un principio).



Le asignamos un nombre (no le pongáis la extensión JSP ya que se pone sola)



Una utilidad muy sencilla que podemos utilizar es el reformateo automático de código (pulsando el botón derecho)

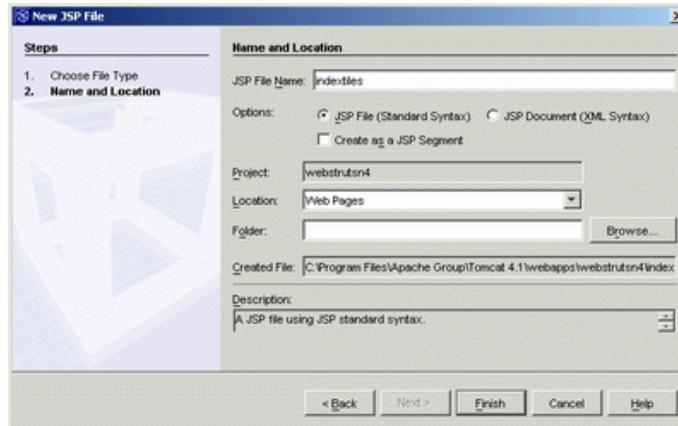


insertar un bloque de HTML y/o JSP.

También podemos indicar que lo que insertaremos será un texto directamente `<tiles:getAsString name="titulo" />`

La página

Nuestra página especificará la plantilla a utilizar y como rellenar cada apartado de la plantilla.

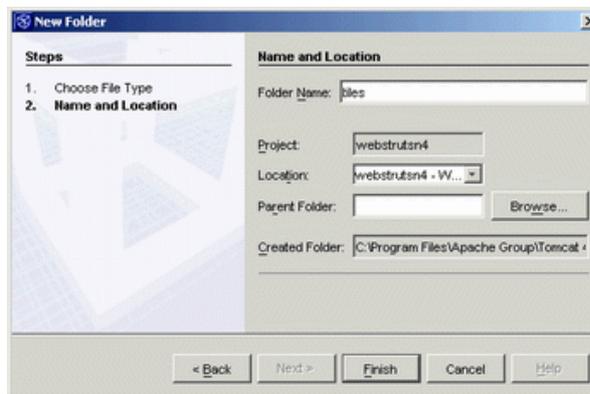


Ahora podemos completar nuestra página real.

Estamos indicando que en la plantilla el título se rellene con el texto "Home del Web con Tiles" y que la sección cuerpo se rellene con el JSP `/tiles/hometile.jsp`

```
<%@page contentType="text/html"%>
<%@ taglib uri="/tags/struts-tiles" prefix="tiles" %>
<tiles:insert page="/plantilla.jsp" flush="true" >
  <tiles:put name="titulo" value="Home del Web con Tiles" />
  <tiles:put name="cuerpo" value="/tiles/hometile.jsp" />
</tiles:insert>
```

Creamos una carpeta donde metemos el JSP con el detalle (en cuerpo de la página)



El cuerpo

Y creamos el JSP con dicho detalle

```
<table border="1" cellpadding="2" style="border-collapse: collapse"
bordercolor="#111111" width="100%" id="AutoNumber2">
<tr>
<td width="100%"><i><b><font size="6" color="#0000FF">Home</font></b></i><p>
  Aquí va a ir el cuerpo de la nueva home
</td>
</tr>
</table>
```

Ya tenemos la página operativa y con nuestro gestor de plantillas.



Opciones para rellenar el campo título

En una aplicación normal, es muy posible que, dependiendo del cuerpo de una página, nos interesase cambiar el título de una plantilla. Vamos a ver como podemos hacer esto.

Asignar un valor directamente con scriptlet

```
<%@page contentType="text/html"%>
<%@taglib uri="/tags/struts-bean" prefix="bean" %>
<%@ taglib uri="/tags/struts-tiles" prefix="tiles" %>

<bean:define id="titulo" value="Home del Web con Tiles y propiedades"
type="String" toScope="request" />

<tiles:insert page="/plantilla.jsp" flush="true" >
  <tiles:put name="titulo" value="<%= titulo %>" />
  <tiles:put name="cuerpo" value="/tiles/hometile.jsp" />
  <tiles:put name="textolegal" type="String">
    Soy el Texto Legal
  </tiles:put>
</tiles:insert>
```

Utilizar EL (Lenguaje de Expresiones)

```
<%@page contentType="text/html"%>
<%@taglib uri="/tags/struts-bean" prefix="bean" %>
<%@ taglib uri="/tags/struts-tiles" prefix="tiles" %>
<%@page isELIgnored="false"%>

<bean:define id="titulo" value="Home del Web con Tiles y EL"
type="String" toScope="request" />

<tiles:insert page="/plantilla.jsp" flush="true" >
  <tiles:put name="titulo" value="<%= titulo %>" />
  <tiles:put name="cuerpo" value="/tiles/hometile.jsp" />
  <tiles:put name="textolegal" type="String">
    Soy el Texto Legal
  </tiles:put>
</tiles:insert>
```

Utilizar la etiqueta beanName

```
<%@page contentType="text/html"%>
<%@taglib uri="/tags/struts-bean" prefix="bean" %>
<%@ taglib uri="/tags/struts-tiles" prefix="tiles" %>

<bean:define id="titulo" value="Home del Web con Tiles y beanName"
type="String" toScope="request" />

<tiles:insert page="/plantilla.jsp" flush="true" >
  <tiles:put name="titulo" beanName="titulo" />
  <tiles:put name="cuerpo" value="/tiles/hometile.jsp" />
  <tiles:put name="textolegal" type="String">
    Soy el Texto Legal
  </tiles:put>
</tiles:insert>
```

Utilizar la etiqueta struts <bean:write>

```
<%@page contentType="text/html"%>
<%@page pageEncoding="UTF-8"%>
<%@taglib uri="/tags/struts-bean" prefix="bean" %>

<%@ taglib uri="/tags/struts-tiles" prefix="tiles" %>
```

```

<bean:define id="titulo" value="Home del Web con Tiles y write"
type="String" toScope="request" />
<tiles:insert page="/plantilla.jsp" flush="true" >
  <tiles:put name="titulo">
    <bean:write name="titulo" />
  </tiles:put>
  <tiles:put name="cuerpo" value="/tiles/hometile.jsp" />
  <tiles:put name="textolegal" type="String">
    Soy el Texto Legal
  </tiles:put>
</tiles:insert>

```

Utilización de etiqueta con cuerpo de tipo cadena de caracteres

```

<%@page contentType="text/html"%>
<%@taglib uri="/tags/struts-bean" prefix="bean" %>
<%@ taglib uri="/tags/struts-tiles" prefix="tiles" %>

<bean:define id="titulo" value="Home del Web con Tiles y JSP Simple"
type="String" toScope="request" />

<tiles:insert page="/plantilla.jsp" flush="true" >
  <tiles:put name="titulo" type="string">
    <%= titulo %>
  </tiles:put>
  <tiles:put name="cuerpo" value="/tiles/hometile.jsp" />
  <tiles:put name="textolegal" type="String">
    Soy el Texto Legal
  </tiles:put>
</tiles:insert>

```

Conclusiones

Struts es un FrameWork básico que nos permite empezar a unificar criterios en el desarrollo de proyectos Java y solucionar las mismas situaciones de un modo homogéneo (aunque no estemos demasiado convencidos de su perfección).

Una de las cosas que menos me gustan (en contra de lo que pudiera parecer) de la tecnología JSP es que lo mismo se puede hacer de demasiados modos (y no hemos visto más que unos pocos). Esto aparentemente puede ser una ventaja pero cuando tienes un equipo muy grande de gente manteniendo aplicaciones, creedme que es un inconveniente.

Lanzo una pregunta al aire (que puede perturbar a departamentos de arquitectura de grandes organizaciones):

¿Creéis que es conveniente unificar todos los desarrollos en mi organización con un único FrameWork?

Mi opinión general es que **NO** (aunque esto dependerá del contexto)

.... y posiblemente necesite **DOS**.

Si desea contratar formación, consultoría o desarrollo de piezas a medida puede contactar con

Creatividad Internet

Somos expertos en:
J2EE, C++, OOP, UML, Vignette, Creatividad ..
 y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	<input type="text"/>
	<input type="button" value="Enviar"/>

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto	Descripción
Comunicación entre TAGs, Beans y JSPs	Os mostramos las posibilidades de comunicación entre JSPs, Bean y etiquetas de usuario.
Consola de administración de Struts	En este tutorial aprenderemos a simplificar la gestión de Struts a través de una consola gráfica gratuita
Aplicación de Patrones de Diseño en Java	En este tutorial os mostramos como las técnicas avanzadas de diseño (como patrones de diseño) contribuyen a la construcción de aplicaciones profesionales en Java.
Patrones de diseño J2EE	Os mostramos una interpretación particular de los patrones de diseño J2EE
Filtros de Servlets en Tomcat	En este tutorial os enseñamos la técnica (poco conocida) del encadenamiento de filtros en la activación de servlets, dentro del entorno Tomcat
Integración de Struts y eclipse	Alejandro Perez nos enseña como construir un entorno de alta eficiencia de desarrollo on Struts a través de plugins de eclipse
Struts Jakarta	Cuando se ha trabajado creando aplicaciones Java poco a poco se va viendo la necesidad de normalizar los desarrollo. Uno de los Framework (entornos) más extendidos es Struts
Aplicación profesional con Struts	En este tutorial, os mostramos como crear una aplicación profesional, usando numerosos patrones y utilizando Struts.
TagLibs y JSPs	Os mostramos como crear librerías de etiquetas para vuestros JSP y así simplificar su construcción.
JSP's y Modelo-Vista-Controlador	En este tutorial os enseñamos como crear un JSP, su relación con los servlets y como crear un ejemplo MVC en Tomcat

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)



www.AdictosAlTrabajo.com Optimizado 800X600