

Ejercicios de JBoss y fuentes de datos

Índice

1	Despliegue de aplicaciones en JBoss.....	2
1.1	Desplegar un .war en JBoss.....	2
1.2	Ejecutar JBoss desde Eclipse.....	2
2	Fuentes de datos en Tomcat.....	2

1. Despliegue de aplicaciones en JBoss

Continuando con la aplicación web `comentarios` de las sesiones anteriores, vamos a tomar el proyecto de Eclipse y a desplegarlo en JBoss. Hay dos posibilidades: exportarlo como un `.war` al servidor ya arrancado o arrancar el servidor desde Eclipse directamente.

1.1. Desplegar un `.war` en JBoss

Tenemos que **asegurarnos de que JBoss está arrancado** previamente. En Eclipse, exportamos el proyecto como un `.war` directamente al directorio `server/default/deploy` de JBoss. Transcurridos unos segundos aparecerá un mensaje como el siguiente, indicando que se ha hecho el despliegue:

```
11:39:22,765 INFO [TomcatDeployer] deploy, ctxPath=/comentarios,
warUrl=.../tmp
/deploy/tmp24234Comentarios-exp.war/
```

Abrir la URL `localhost:8080/comentarios` en el navegador para comprobar el despliegue. Insertar un comentario de prueba.

Ahora podemos abrir la consola web de JBoss en `http://localhost:8080/web-console`. A la izquierda aparecerán las aplicaciones desplegadas en el servidor. Abrimos `J2EE Domains > JBoss > jboss.management.local` hasta encontrar `comentarios.war`. Aquí podemos encontrar estadísticas de uso/tiempo de la aplicación. Comprobar el número de veces que se ha invocado el servlet (debería ser 1).

1.2. Ejecutar JBoss desde Eclipse

En este caso **tenemos que tener JBoss parado**, ya que Tomcat lo pondrá en marcha

1. Crear el servidor en Eclipse: `File > New > Other`, buscar `Server` y elegir `JBoss`
 4. 2. Tendremos que decirle a Eclipse dónde está instalado el servidor. Una vez hecho esto, aceptamos las opciones por defecto
2. En las propiedades del proyecto, dentro de `targeted runtimes` deberíamos marcar también `JBoss`
3. Al ejecutar el proyecto con `Run As` nos dará a elegir el servidor. Elegir `JBoss`. A partir de aquí todo será equivalente a lo que antes hacíamos con Tomcat. Eclipse arrancará JBoss, desplegará la aplicación y nos mostrará su página principal.

Nota:

Obsérvese que, a diferencia de lo que ocurre con Tomcat, con el que se usa una configuración "temporal" propia de Eclipse, en JBoss estamos usando la configuración original del servidor.

2. Fuentes de datos en Tomcat

Para este ejercicio nos volveremos a pasar a Tomcat, ya que es el servidor empleado originalmente en SIGEM. De cualquier modo, como se ha visto en la sesión, la configuración en JBoss es similar. Asegurarse de **parar JBoss** desde Eclipse, ya que funciona en el mismo puerto que Tomcat.

Vamos a cambiar la aplicación de comentarios para que guarde la información en una base de datos. Para ello, seguiremos estos pasos:

1. Copiar el .jar con el driver de MySQL al directorio `lib` de Tomcat. El .jar está en las plantillas de la sesión. Si Tomcat estuviera arrancado **tendríamos que rearrancarlo** para que tome el driver.
2. Crear la base de datos: ejecutar el script `comentarios.sql` en MySQL, para crear la BD con la tabla `comentarios`. El script está en las plantillas de la sesión.
3. Crear la fuente de datos en el contexto. En el directorio `META-INF` debe haber un `context.xml` de sesiones anteriores. Editarlo para incluir la configuración de la fuente de datos:

```
<Context>
  <Resource
    name="comentariosDS"
    type="javax.sql.DataSource"
    auth="Container"
    username="root"
    password="root"
    driverClassName="com.mysql.jdbc.Driver"
    url="jdbc:mysql://localhost:3306/comentarios"
    maxActive="20"
    maxIdle="5"
    maxWait="10000" />
</Context>
```

4. Cambiar el código del servlet `addComentario` y de la página `comentarios.jsp` por los incluidos en las plantillas de la sesión, que hacen uso de la fuente de datos con JDBC. Añadir el archivo `FuenteDatos.java`, que se encarga del acceso al Datasource propiamente dicho.
5. Comprobar el correcto funcionamiento de la aplicación, verificando que almacena la información en la tabla.

