



Servidores Web

Sesión 3: Configuración de Tomcat. Configuración de aplicaciones web.



Puntos a tratar

- Configuración de Tomcat
 - Estructura física y lógica
 - Formas de cambiar la configuración
 - Configurar el host
 - Configuración en Eclipse
- Configuración de aplicaciones web
 - El descriptor de despliegue
 - El contexto de la aplicación en Tomcat



Estructura física de Tomcat

- Directorio *home* de Tomcat
 - bin
 - lib: **clases comunes a serv. y aplic.**
 - Por ejemplo, drivers de BD
 - conf: **configuración del servidor**
 - logs: **dir. por defecto de logs de depuración**
 - webapps: **aplic. web**
 - work, temp: **dir. temporales**

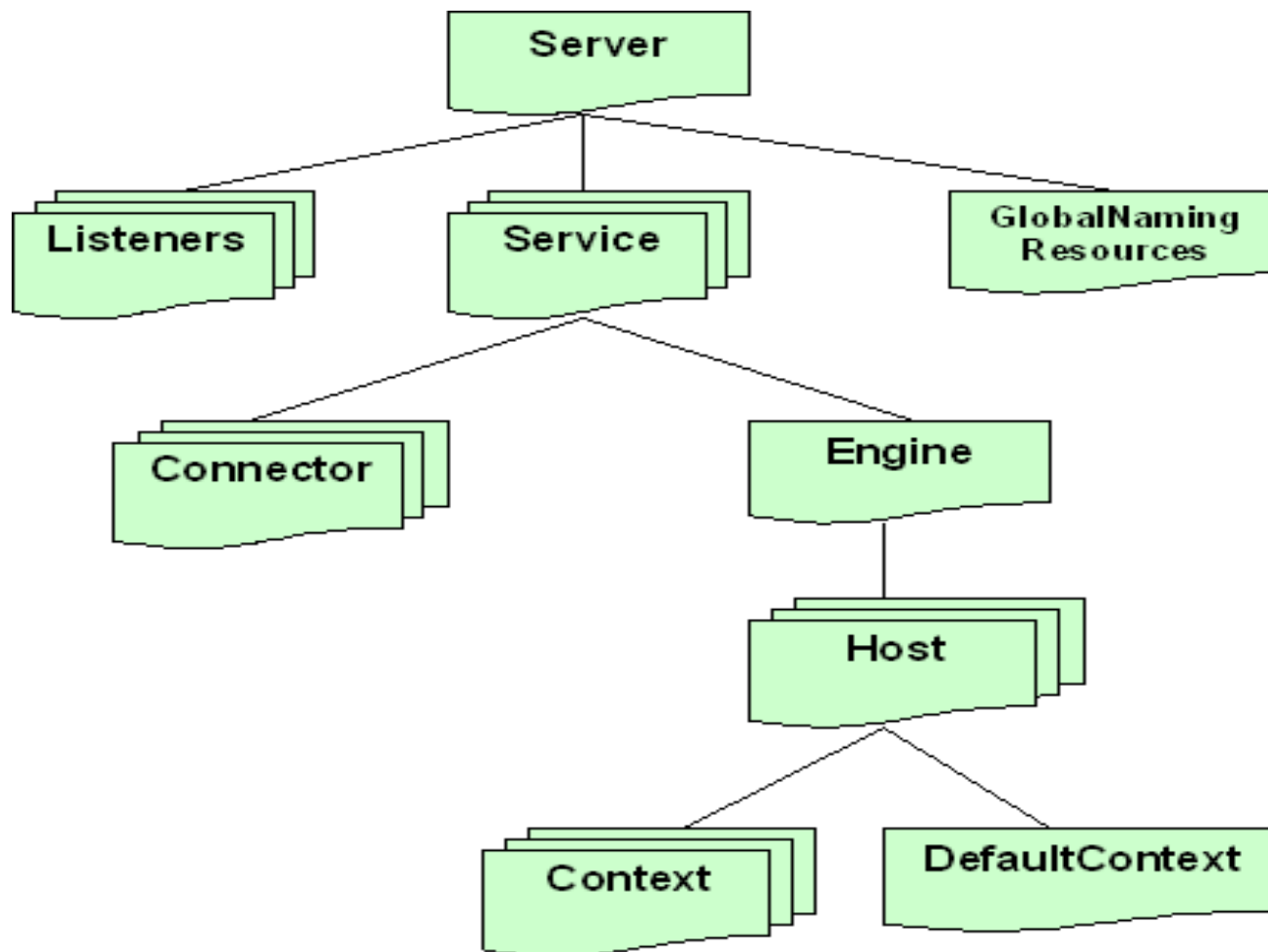


Ficheros de configuración

- Dentro de **conf**:
 - server.xml (configuración principal)
 - web.xml (config.global a todas las aplicaciones)
 - tomcat-users.xml (logins y passwords de usuarios)
 - catalina.policy (fichero de políticas de seguridad)



Estructura lógica de Tomcat (*server.xml*)





Componentes principales

- *Server*: el propio Tomcat. Sólo hay una instancia de este elemento
- *Listener*: monitorizan los contenedores web
- *GlobalNamingResources*: mapean variables JNDI
- *Service*: conjunto de conectores que reciben peticiones y un *engine* que las procesa
- *Connector*: acepta ciertas peticiones y las pasa al *engine*
- *Engine*: representa al contenedor web (p. ej. Catalina)
- *Host*: representa al host o host virtual (p.ej. localhost)
- *Context*: representa una aplicación web en un host

Configuración del host

- La etiqueta *Host* de *server.xml* define la configuración general para un host o host virtual (es decir, un subgrupo de aplicaciones dentro del servidor web)

```
<Host name="localhost" debug="0" appbase="webapps"  
      unpackWars="true" autoDeploy="true">
```

Atributo	Significado	Valor por defecto
name	nombre del host o host virtual	ninguno
debug	nivel de mensajes de depuración	0
appBase	directorio donde se instalarán las aplicaciones de este host (si es relativa, la ruta se supone con respecto al directorio de Tomcat)	ninguno
unpackWARs	si es true, las aplicaciones empaquetadas en WAR se desempaquetan antes de ejecutarse	true
autoDeploy	si es true, se utiliza el despliegue automático de aplicaciones	true
liveDeploy	si es true, se utiliza despliegue automático sin necesidad de reanunciar Tomcat	true



Componentes reubicables

- Hay algunos componentes que se pueden definir a varios niveles, según el ámbito que queramos que tengan:
 - *Valves*: para filtrar peticiones
 - *Loggers*: para logs de depuración de errores
 - *Realms*: define un conjunto de usuarios con permiso de acceso a un determinado contexto o aplicación web
 - *Managers*: implementan el manejo de sesiones HTTP.
 - *Loaders*: cargador de clases para aplicaciones web



Configuración de la aplicación: web.xml

- Cada aplicación web / contexto debe tener un fichero descriptor de despliegue
 - Fichero `web.xml`
 - Ubicado en la ruta `/WEB-INF/web.xml` del contexto
- Describe y configura la aplicación web
- Fichero XML definido de forma estándar para la configuración de las aplicaciones Java EE



Descriptor básico

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:web="http://java.sun.com/xml/ns/javaee/web-
app_2_5.xsd"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
id="WebApp_ID" version="2.5">
<web-app>
  <display-name>Mi Aplicacion Web</display-name>
  <description>
    Esta es una aplicacion web sencilla a modo de ejemplo
  </description>
  <!-- Resto de elementos -->
</web-app>
```



Elementos del descriptor

- En el descriptor de despliegue se configura:
 - Variable globales del contexto
 - Filtros
 - Servlets y mapeo a URLs
 - Listeners (de cambios en el contexto)
 - Sesiones
 - Página de inicio
 - Restricciones de seguridad
 - Librerías de *tags*
 - Etc ...



Contexto

- Cada Aplicación Web es un contexto
- A cada contexto se le asigna una ruta dentro del servidor
 - Por ejemplo, si asignamos la ruta `aplic` al contexto correspondiente a la siguiente estructura:

```
/pagina.htm
```

```
/WEB-INF/web.xml
```

- Podremos acceder a nuestra página con

```
http://localhost:8080/aplic/pagina.htm
```



Contexto de una aplicación web

- Elemento Context

Atributo	Significado	Valor por defecto
<code>crossContext</code>	indica si la aplicación web puede comunicarse con otras	<code>false</code> (por razones de seguridad)
<code>debug</code>	nivel de mensajes de depuración	0
<code>docBase</code>	directorio donde está la aplicación, bien especificado de modo absoluto o con respecto al <code>appBase</code> del Host	ninguno
<code>path</code>	<i>path</i> de esta aplicación, que se hace coincidir con la URL de la petición para ver qué aplicación debe servirla.	ninguno
<code>reloadable</code>	permite monitorizar los cambios de clases en <code>/WEB-INF/classes</code> y <code>/WEB-INF/lib</code> para que no sea necesario reiniciar Tomcat	<code>false</code> (se suele poner a <code>true</code> durante el desarrollo)



Dónde especificar el *Context*

- Dentro del `server.xml`

```
<Context docBase="miAplic" path="/miaplic"  
.../>
```

- Desaconsejado, hay que tocar la configuración del servidor entero
- Para la aplicación individualmente
 - Dentro de una carpeta **META-INF**, en un archivo llamado **context.xml**
 - Se puede hacer de otras formas, consultar la documentación.



Valves

- Componentes que se insertan en el ciclo de procesamiento de la petición para controlar varios aspectos:
 - Registro de accesos
 - Filtro de hosts o de IPs
 - Volcado de la petición
 - ...
- Clases Java que el usuario podría implementar
- Se pueden poner a distintos niveles: **engine, host o context**
 - O sea, para todas las aplicaciones o para una sola



Ejemplos de *Valves*

- Log de accesos

```
<Valve  
className="org.apache.catalina.valves.AccessLogValve"  
directory="logs" prefix="localhost_access_log."  
suffix=".txt" pattern="common" resolveHosts="false"/>
```

- Filtro de IPs

```
<Valve  
className="org.apache.catalina.valves.RemoteAddrValve"  
allow="127.0.0.1" />
```