

# Instalación y configuración del Apache TOMCAT

## 1. Introducción

El TOMCAT es un servidor web, un *Servlet Container* y un motor de páginas JSP desarrollado por Apache. Todos los alumnos deberán bajar de la página web de Apache un servidor TOMCAT (versión 10) e instalarlo en su cuenta para realizar las prácticas. Los profesores también emplearán la versión 10 para corregir las prácticas.

## 2. Entorno del laboratorio

Tras descargar y descomprimir el TOMCAT en vuestra cuenta, podéis ver que aparece un directorio que comienza por *apache-tomcat*. En él hay varios subdirectorios:

- Directorio *bin*, con los comandos para arrancar (“*./startup.sh*”) y detener (“*./shutdown.sh*”) el servidor TOMCAT. Para trabajar con el TOMCAT, el primer paso es arrancarlo con el comando “*./startup.sh*”.
- Directorio *conf*, con los ficheros de configuración. El más importante es el fichero *server.xml*, en el que se pueden configurar varias cosas:
  - El puerto TCP en el que escucha las solicitudes de apagado (*shutdown*) el TOMCAT. Es la línea que empieza por ‘<*Server port="8005" shutdown="SHUTDOWN"*>’. Cada alumno debe cambiarlo por el número ‘6000+*X*’ (siendo *X* el número de su cuenta). En el Tomcat de MOOVI se ha puesto un ‘6XXX’ para identificarlo.
  - El puerto TCP en el que escucha las solicitudes el TOMCAT. Es la línea que empieza por “<*Connector port='8080'*”. Cada alumno debe cambiarlo por el número ‘7000+*X*’ (siendo *X* el número de su cuenta). En el Tomcat de MOOVI se ha puesto un ‘7XXX’ para identificarlo.
  - Un contexto. Un contexto es un prefijo que redirige todas las solicitudes que empiecen por el mismo a una zona en particular del sistema de ficheros. Vosotros tenéis que declarar el contexto “*sintX*” para que se dirija al directorio “*public\_html/webapps*” de vuestra cuenta. Al final del fichero, tras el elemento <*Valve ...* />, hay que añadir lo siguiente, cambiando la *X* por vuestro número de cuenta (en el Tomcat de MOOVI ha se ha añadido, sólo tenéis que cambiar la *X*):

```
<Context path="/sintX" docBase="/home/eetlabs.local/sint/sintX/public_html/webapps/"
    debug="0" reloadable="true">
</Context>
```

Esto enlaza el contexto *sintX* con el directorio *public\_html/webapps* **que debéis crear en vuestra cuenta**. También se debe comprobar que el directorio especificado en *docBase* es correcto, que es la ruta real a vuestra cuenta.

- Directorio *logs*, donde están los ficheros donde quedan registrados los sucesos y errores del TOMCAT. El general se llama *catalina.out*, donde el TOMCAT escribe cualquier posible error que detecte, tanto al arrancar como durante la ejecución de un servlet. Es un fichero a consultar cuando ocurra algún tipo de error, para ver si tuvo lugar alguna excepción no capturada.

En ese mismo fichero se imprimirán los mensajes del método *System.out.println()* que se ejecute en vuestro servlet. Es decir, si en vuestro servlet ejecutáis *System.out.println("hola")*, ese mensaje se escribirá en el *catalina.out*.

## 3. El contexto del usuario

Una vez configurado el TOMCAT personal tal como se describe en la sección anterior, existirá un contexto (llamado *sintX*) para la cuenta del usuario. Eso quiere decir que todas las solicitudes que comiencen por la cadena “*sintX*” (tras el nombre de la máquina y el puerto en el que escucha) serán dirigidas a vuestro directorio *webapps*. Es decir, para la cuenta *sint199*, la solicitud:

*http://localhost:7199/sint199/pag.html*

muestra la página *pag.html* situada en el directorio *public\_html/webapps* de la cuenta *sint199*. Si creamos un directorio dentro de *webapps*, basta con reflejarlo en la ruta de la solicitud para acceder a sus contenidos.

Para poner en servicio un *servlet*, hay que crear una estructura de directorios dentro de vuestra cuenta:

- ☐ Dentro del directorio *webapps* hay que crear un directorio *WEB-INF*. Nada de lo que haya en *WEB-INF* es directamente accesible pidiéndolo desde un navegador.
- ☐ Y dentro del directorio *WEB-INF* hay que crear un directorio *classes*.

#### 4. Recomendaciones

- Después de lanzar el TOMCAT (comando “*./startup.sh*”) es conveniente comprobar que se está ejecutando, por si se ha producido un error en la configuración que impida su ejecución.
  - El comando “*ps auxw*” nos muestra la lista de procesos que se están ejecutando en la máquina. Con el comando “*ps auxw | grep tomcat*”) nos quedamos sólo con las líneas que contengan la palabra “*tomcat*”). Tras arrancar correctamente el TOMCAT, debe haber una línea de gran longitud que contenga esa palabra.
- Tras lanzar el TOMCAT es conveniente leer el fichero *catalina.out* para comprobar si se ha producido algún error (comprobando la fecha del mensaje para estar seguros de que el mensaje se corresponde con la ejecución actual). También es conveniente consultar este fichero cuando se produce un error en la ejecución de un *servlet* y nos devuelve una excepción, o no nos devuelve nada, o el resultado es inesperado.
- Después de trabajar con el TOMCAT, hay que detener su ejecución (comando “*./shutdown.sh*”). Comprobad que el proceso del TOMCAT realmente ya no existe (mediante el anterior comando ‘*ps*’):
  - Si el TOMCAT no se muere con el comando *shutdown.sh*, es posible matarlo mediante “*kill -9 PID*” (*PID* es el número de proceso del TOMCAT que se está ejecutando, el que aparece en la segunda columna que proporciona el comando ‘*ps*’).
- Tras producirse un error grave en la ejecución de un *servlet* (una excepción), es posible que el TOMCAT lo desactive, o que incluso desactive el contexto. A partir de ese momento, cualquier solicitud provoca un mensaje del tipo “*recurso inexistente*”, “*contexto no declarado*”, o similar. Para que vuelva a ejecutar el *servlet* será necesario reiniciar el TOMCAT.
- Normalmente, cualquier modificación del *servlet* y su posterior compilación es detectada por el TOMCAT y la nueva versión es cargada y ejecutada. Si no es así, y se sospecha que se está ejecutando la copia vieja y no la nueva, se debe reiniciar el TOMCAT.
- Si se modifica algún fichero asociado, por ejemplo, la hoja de estilo CSS, es posible que el Firefox siga empleando la antigua, recuperándola de la caché. Si vemos que nuestros cambios no tienen efecto, se debe recargar el fichero CSS. Para ello, en el Firefox, se abre el código fuente de la página que devuelve el *servlet*. En la línea donde se carga la hoja CSS, el nombre del fichero CSS es un enlace, y al pulsarlo se abre la hoja CSS. Si vemos que es la antigua, recargamos esa página para que aparezcan y se apliquen las modificaciones.