

# Sesiones

## Introducción (Manager)

---

### ► Interfaz **Manager** para gestionar las sesiones:

- **org.apache.catalina.Manager**
- Implementaciones:
  - **org.apache.catalina.session.StandardManager**
    - Implementación por defecto.
    - Tiempo limitado de persistencia de las sesiones.
  - **org.apache.catalina.session.PersistenManager**
    - Permite almacenar la sesiones en disco o en una base de datos.
  - **org.apache.catalina.ha.session.DeltaManager**
  - **org.apache.catalina.ha.session.BackupManager**
    - Para *clusters*.

# Sesiones

## Introducción (Manager)

---

- Configuración general del Manager
  - CATALINA\_BASE/conf/context.xml

```
<!-- The contents of this file will be loaded for each web application -->
<Context>

    <!-- Default set of monitored resources -->
    <WatchedResource>WEB-INF/web.xml</WatchedResource>

    <!-- Uncomment this to disable session persistence across Tomcat restarts -->
    <!--
    <Manager pathname="" />
    -->

    <!-- Uncomment this to enable Comet connection tacking (provides events
         on session expiration as well as webapp lifecycle) -->
    <!--
    <Value className="org.apache.catalina.valves.CometConnectionManagerValve" />
    -->
```

- Por defecto no está definido.

# Sesiones

## Tiempo de vida de las sesiones

---

- ▶ Tiempo de vida de las sesiones inactivas.
- ▶ Por defecto *Tomcat* define 30 minutos.
- ▶ Se puede definir en el descriptor de despliegue de la aplicación.
- ▶ Ejemplo (`web.xml`).

```
<session-config>  
  <session-timeout>15</session-timeout>  
</session-config>
```

# Sesiones

## StandardManager

---

- ▶ Implementación por defecto del Manager.
- ▶ Si no se configura un Manager en context.xml *Tomcat* (configuración por defecto) instancia un **StandardManager** por defecto.
  - Almacena/restaura la información de las sesiones activas en el disco cuando se reinicia el servidor.
  - Se almacenan en el directorio de trabajo de cada aplicación en el fichero **SESSIONS.ser**.

CATALINA\_BASE/work/Catalina/localhost/<app>/SESSIONS.ser

- Solo las sesiones activas.
- Las sesiones inactivas se pierden cuando se reinicia el servidor.

# Práctica

## ► Práctica 7.9

### ○ Sesiones.

```
<display-name>compras</display-name>

<welcome-file-list>
  <welcome-file>index.jsp</welcome-file>
</welcome-file-list>

<session-config>
  <session-timeout>15</session-timeout>
</session-config>
```

```
<servlet>
  <servlet-name>PonerEnCesta</servlet-name>
  <servlet-class>
</servlet>

<servlet>
  <servlet-name>
  <servlet-class>
</servlet>
```

<a href="#">/Calendar</a>	Ninguno especificado	gwt-Calendar Compiled: Sun Jan 27 06:17:26 GMT 2008	true	0	Arrancar Parar Recargar Replegar
					Expirar sesiones sin trabajar ≥ 30 minutos
<a href="#">/compras</a>	Ninguno especificado	compras	true	0	Arrancar Parar Recargar Replegar
					Expirar sesiones sin trabajar ≥ 15 minutos
<a href="#">/docs</a>	Ninguno especificado	Tomcat Documentation	true	0	Arrancar Parar Recargar Replegar
					Expirar sesiones sin trabajar ≥ 30 minutos

```
alumno@ServidorLinux01:/var/lib/tomcat7/work/Catalina/localhost/compras$ ls
org SESSIONS.ser
alumno@ServidorLinux01:/var/lib/tomcat7/work/Catalina/localhost/compras$ _
```

# Sesiones

## StandardManager

---

- ▶ Es posible personalizar la configuración del StandardManager usando el elemento `<Manager>` dentro de un fichero `context.xml`.

```
<Manager>
    className="org.apache.catalina.session.StandardManager"
    maxActiveSessions="-1"
    maxInactiveInterval = "30"
    sessionIdLength="32"
</Manager>
```

# Sesiones

## StandardManager

---

### ► Atributos (1)

- **classname**
  - Clase del servidor que implementa el gestor.
- **distributable**
  - Fuerza que se implementen todos los atributos de las sesiones.
- **maxActiveSessions**
  - Cuando se supera este limite, se empiezan a serializar a disco todas las nuevas sesiones.
  - -1 significa ilimitadas sesiones.
- **maxInactiveInterval**
  - Máximo intervalo de tiempo desde que una sesión es marcada como inactiva.

# Sesiones

## StandardManager

---

### ► Atributos (2)

- **sessionIdLength**
  - Longitud en bytes de los identificadores de sesión.
- **pathname**
  - *Path* absoluto relativo (al directorio de trabajo de la aplicación) donde se almacenan los ficheros de sesión.
  - Por defecto es **SESSIONS.ser**
- **processExpiresFrequency**
  - Frecuencia del proceso en *background* que comprueba las sesiones inactivas.
- **secureRandomClass**
  - Clase usada para generar los ids de sesión.



# Sesiones

## StandardManager

---

### ► Atributos (3)

- **secureRandomAlgorithm**
  - Algoritmo usando para generar números aleatorios.
- **secureRandomProvider**
  - Proveedor de generación de números aleatorios.

# Sesiones

## Deshabilitar las sesiones persistentes

---

- ▶ Por ejemplo, si quieres forzar a los usuario a crear nuevas sesiones si crea una nueva versión de la aplicación.
- ▶ Crear un `<Manager>` con el siguiente contenido.

```
<Manager pathname="" />
```

# Sesiones

## PersistentManager

- ▶ En aplicaciones con muchos usuarios (miles) simultaneemos la información de las sesiones puede ser muy grande (mucho espacio en memoria RAM).
- ▶ **PersitentManager** permite almacenar la información de las sesiones en almacenamiento externo (swap entre memoria y disco).

# Sesiones

## PersistentManager

---

- ▶ Ofrece dos implementaciones.
  - FileStoreManager -> Disco
  - JDBCStoreManager -> Base de datos.

```
<Manager className="org.apache.catalina.session.PersistentManager"
  saveOnRestart="true"
  maxActiveSession="-1"
  minIdleSwap="0"
  maxIdleSwap="60"
  maxIdleBackup="-1">
  <Store className="org.apache.catalina.session.FileStore" />
</Manager>
```

# Sesiones

## PersistentManager

---

```
<Manager
  className="org.apache.catalina.session.PersistentManager"
  saveOnRestart="true"
  maxActiveSessions="-1"
  minIdleSwap="-1"
  maxIdleBackup="-1">
  <Store className="org.apache.catalina.session.JDBCStore"
    driverName="org.gjt.mm.mysql.Driver"
    connectionURL="jdbc:mysql://localhost/tomcatsessions?user=tomcat; password=despliegue"
    sessionTable="tomcat_session_data"
    sessionIdCol="session_id"
    sessionAppCol="application_id"
    sessionDataCol="session_data"
    sessionValidCol="valid"
    sessionMaxInactiveCol="maxinactive"
    sessionLastAccessedCol="last_access"
    checkInterval="60"
  />
</Manager>
```

```
create database tomcatsessions;
use tomcatsessions;
create table tomcat_session_data (
  session_id varchar(100) not null primary key,
  valid char(1) not null,
  maxinactive int not null,
  last_access bigint not null,
  application_name varchar(255),
  session_data mediumblob
);
```