
INICIALIZACIÓN SERVLET Y LISTENERS DE CONTEXTO

Eduard Lara

INDICE

1. Parámetros iniciales del servlet
2. Recuperar los parámetros en el servlet
3. Parámetros iniciales de contexto
4. Listeners de contexto

1. PARÁMETROS INICIALES DEL SERVLET

- Un servlet puede tener cualquier número de parámetros de inicialización.
- El método init debe obtener los valores a partir del objeto ServletConfig.
- El contenedor web crea el objeto de configuración basándose en los parámetros de inicialización especificados en el descriptor de despliegue
- En el archivo web.xml debemos declarar los parámetros, tal como muestra la imagen.

```
<init-param>  
  <param-name>Oferta</param-name>  
  <param-value>Hoy todos los monitores al 10% de dto.</param-value>  
</init-param>
```

1.1. API DE SERVLETCONFIG

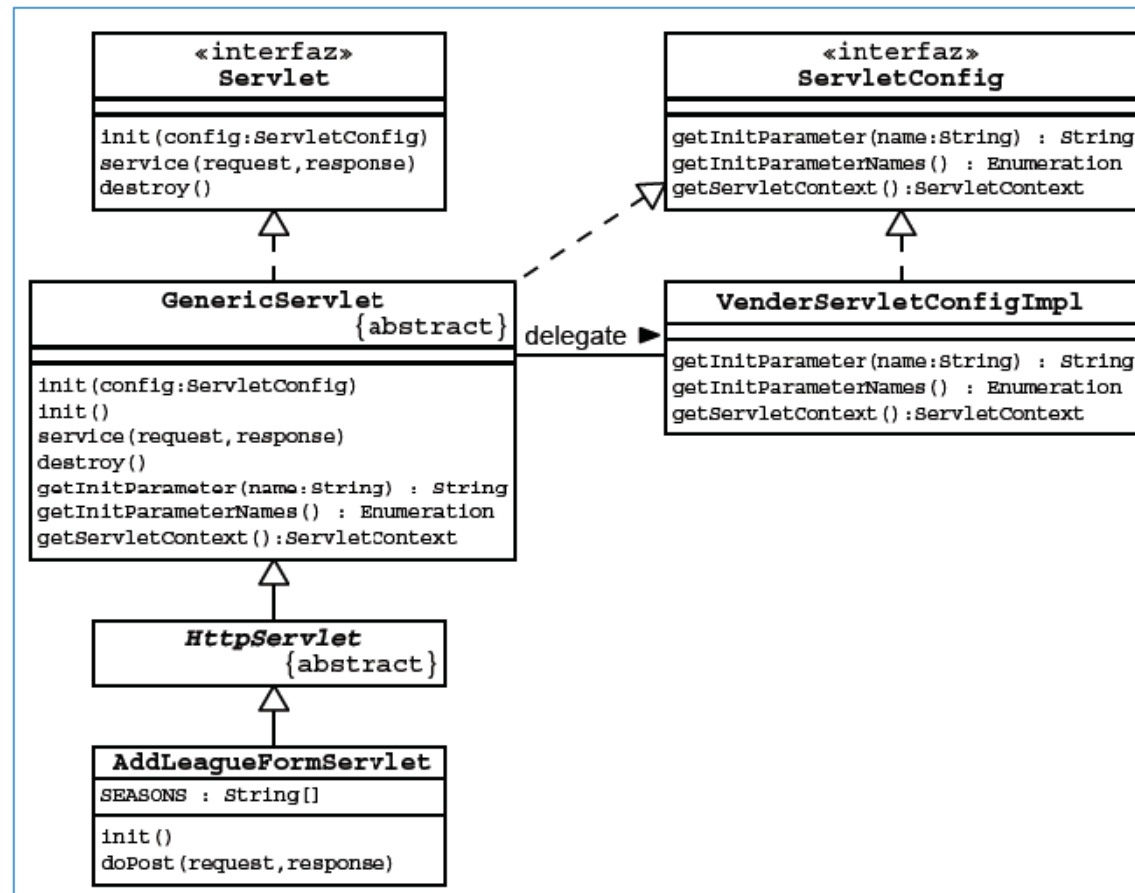
- Cada contenedor web debe implementar la interfaz ServletConfig.
- Las instancias de esta clase se pasan al método init(ServletConfig) definido en la interfaz de Servlet.
- La clase GenericServlet (suministrada en la API de servlet) implementa la interfaz de Servlet.
- Implementa el método init(ServletConfig), que almacena el objeto de configuración (delegate) y después llama al método init().
- Se trata de un método init sin argumentos que se sustituye en las clases de servlet.

1.1. API DE SERVLETCONFIG

- La interfaz `ServletConfig` proporciona el método `getInitParameter`, que permite recuperar los parámetros de inicialización del servlet.
- Por motivos prácticos, la clase `GenericServlet` también implementa la interfaz `ServletConfig`, y esos métodos delegan las llamadas en el objeto de configuración almacenado.
- Ello simplifica el código de servlet, porque permite llamar directamente al método `getInitParameter` (sin acceso directo al objeto de configuración).

1.1. API DE SERVLETCONFIG

La siguiente imagen ilustra la API de servlet y su relación con la interfaz ServletConfig.



2. RECUPERAR LOS PARÁMETROS EN EL SERVLET

Para recuperar los parámetros iniciales utilizamos el método `init(ServletConfig sc)` del servlet tal y como muestra la imagen:

```
public void init(ServletConfig config) throws ServletException {  
    // recuperamos el parametro inicial del servlet del objeto ServletConfig  
    String mensaje = config.getInitParameter("Oferta");  
    // Recuperar el contexto del servlet es lo mismo que el contexto de la aplicacion  
    ServletContext aplicacion = config.getServletContext();  
    aplicacion.setAttribute("msg", mensaje);  
    System.out.println(">>>>>>>>>>>>>>> Se ha creado la instancia del servlet");  
}
```

2. RECUPERAR LOS PARÁMETROS EN EL SERVLET



RECUERDA QUE...

- Los parámetros iniciales del servlet son datos que se recogen en el momento de la inicialización de este.
- Se declaran en el descriptor de despliegue web.xml y se recuperan en el método init().

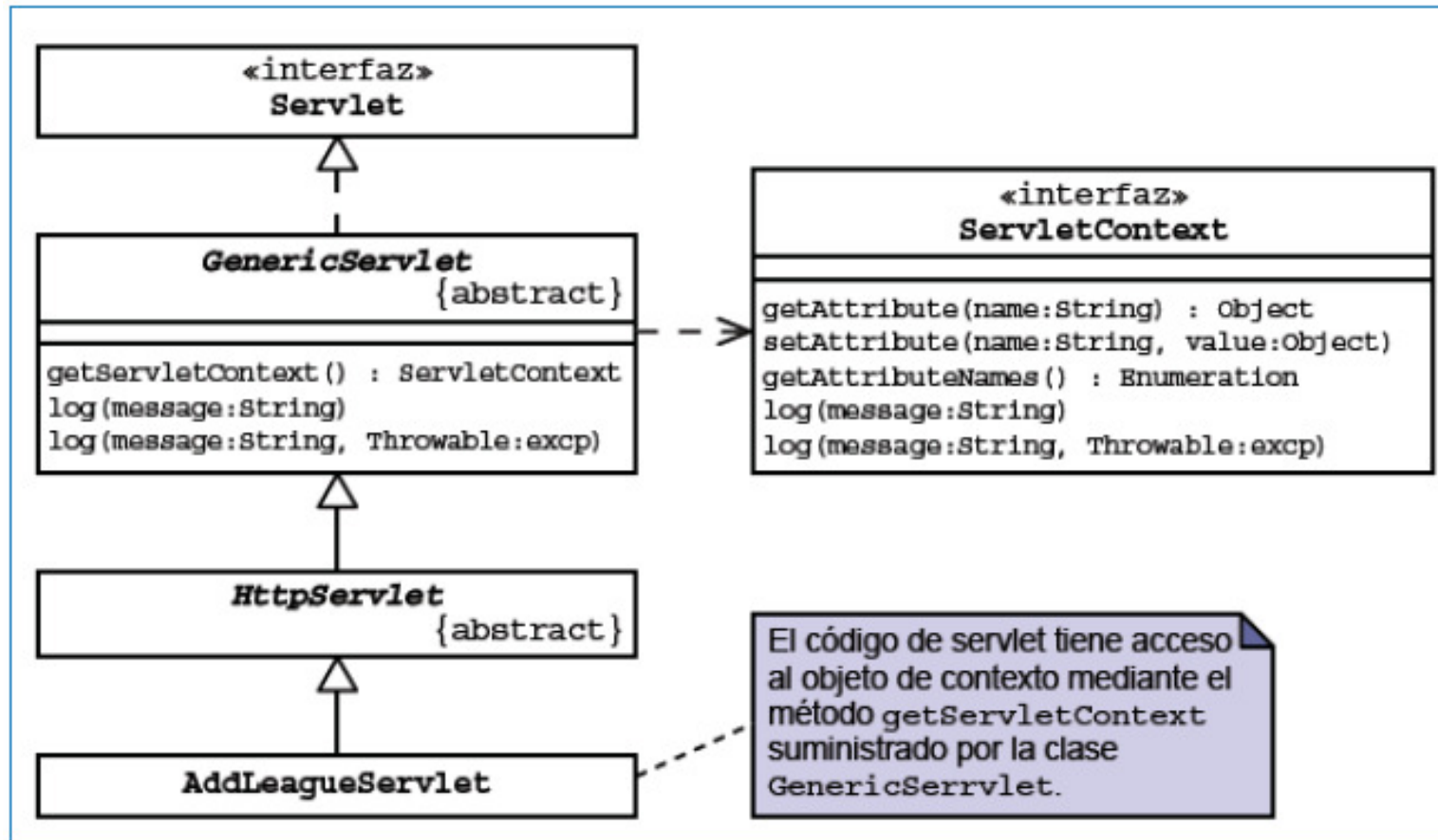
3. PARÁMETROS INICIALES DE CONTEXTO

- Una aplicación web es una colección autocontenida de recursos estáticos y dinámicos: páginas HTML, archivos multimedia, archivos de datos y de recursos, servlets (y páginas de JSP) y otros objetos y clases Java auxiliares.
- El descriptor de despliegue de la aplicación web sirve para especificar la estructura y los servicios utilizados por una aplicación web.
- Un objeto `ServletContext` es la representación de tiempo de ejecución de la aplicación web.
- El objeto de contexto es accesible a todos los servlets en una aplicación web.

3. PARÁMETROS INICIALES DE CONTEXTO

- En el objeto de contexto se pueden almacenar atributos de ámbito de contexto mediante el método `setAttribute`. Los atributos pueden recuperarse del objeto de contexto mediante el método `getAttribute`.
- También es posible eliminar un atributo de contexto con el método `removeAttribute`. El ámbito de contexto también se denomina ámbito de aplicación.
- El método `getServletContext` lo suministra la clase `GenericServlet`, ampliada por la clase `HttpServlet` y las clases de servlet del usuario.

3. PARÁMETROS INICIALES DE CONTEXTO



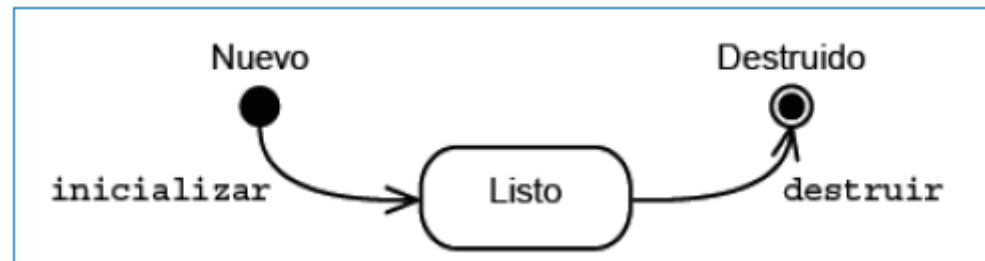
3. PARÁMETROS INICIALES DE CONTEXTO

En la siguiente figura, se puede ver cómo se hace la declaración de parámetros de contexto en el web.xml:

```
<context-param>  
  <param-name>OfertaMes</param-name>  
  <param-value>Este mes las impresoras con un 25% de dto</param-value>  
</context-param>
```

4. LISTENERS DE CONTEXTO

- La aplicación web (representada por el objeto de contexto) tiene un ciclo de vida gestionado por el contenedor web. Este ciclo de vida es similar al del servlet.

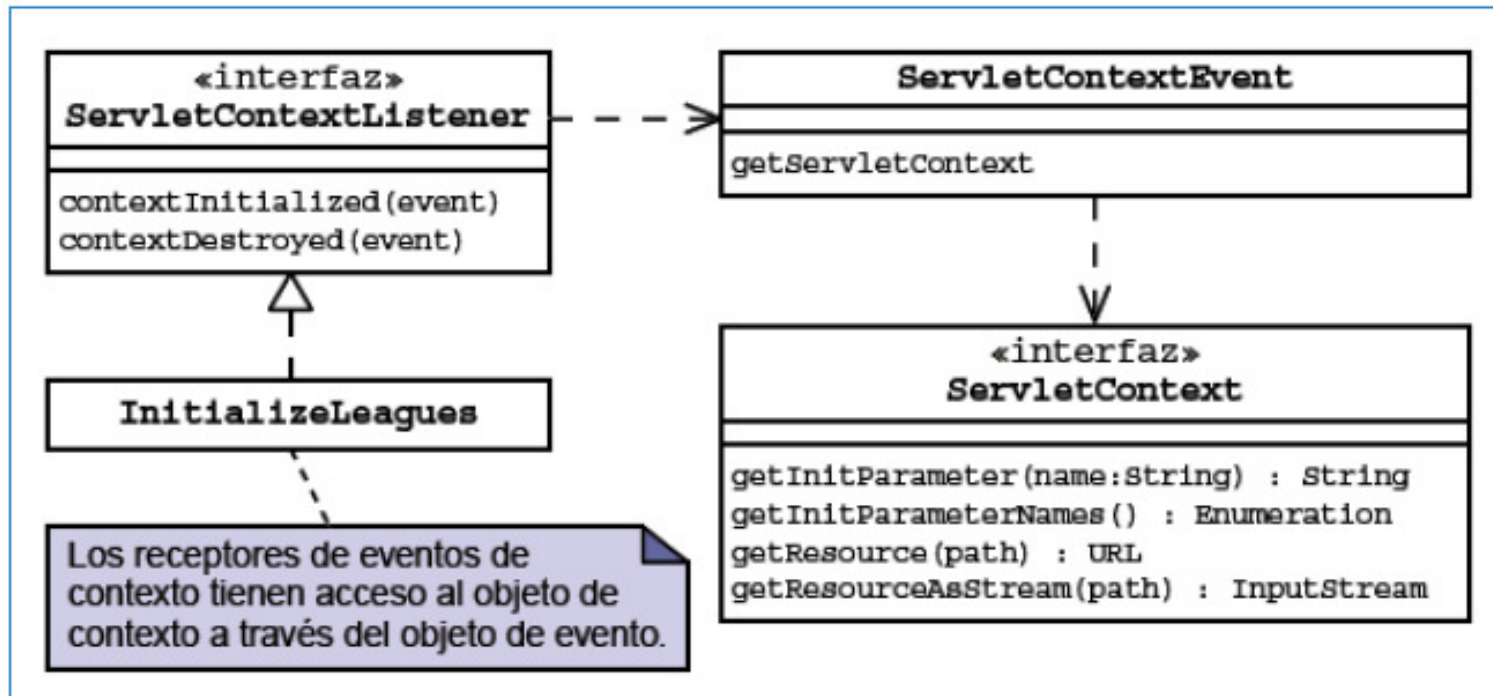


- Cuando se inicia el contenedor web, se inicializa cada aplicación web. Cuando se cierra el contenedor web, se destruye cada aplicación web.
- Para recibir estos eventos del ciclo de vida de la aplicación web se necesita un receptor de eventos de contexto de servlet.

4. LISTENERS DE CONTEXTO

- Normalmente, los datos compartidos de la aplicación deben hallarse en memoria antes de ejecutar cualquier solicitud HTTP en la aplicación web.
- La interfaz `ServletContextListener` tiene dos métodos.
 - El contenedor web invoca el método `contextInitialized` cuando se ha iniciado la aplicación web.
 - El contenedor web invoca el método `contextDestroyed` cuando se está cerrando la aplicación web.
- Ambos métodos incluyen un argumento de evento. El objeto de evento proporciona acceso al objeto de contexto mediante el método `getServletContext`. Los receptores de eventos de contexto deben implementar esta interfaz

4. LISTENERS DE CONTEXTO



4. LISTENERS DE CONTEXTO

Configuración del receptor de eventos

Debemos configurar el listener de contexto en el descriptor de despliegue web.xml.

```
<listener>
  <description>ServletContextListener</description>
  <listener-class>app.web.ListenerAplicacion</listener-class>
</listener>
```

Recuperar los parámetros de contexto

Los parámetros de contexto los recuperamos en el listener de la aplicación. Concretamente en el método contextInitialized.

```
public void contextInitialized(ServletContextEvent sce) {
    ServletContext aplicacion = sce.getServletContext();
    String mes = aplicacion.getInitParameter("OfertaMes");
    aplicacion.setAttribute("mes", mes);
}
```




RECUERDA QUE...

- Los parámetros de contexto se consideran parámetros iniciales de la aplicación.
- Se declaran en el web.xml como <context-param>
- Se recuperan a través de un Listener de contexto.