

¿Qué es el analizador léxico JAX?

Jax es un compilador léxico creado en lenguaje Java, que genera un escáner a partir de expresiones regulares que existen por defecto en un archivo de java.

JAX-WS (Java API for XML Web Services) es una interfaz de programación de aplicaciones (API) de Java en Extensible Markup Language (XML) para la creación de servicios web (WS). Es parte de la plataforma Java EE de Sun Microsystems. Al igual que las otras API de Java EE, JAX-WS utiliza anotaciones, introducidas en Java SE 5, para simplificar el desarrollo y despliegue de los clientes y puntos finales de servicios web. Es parte del Java Web Services Development Pack.

¿Cómo funciona?

Jax es el que se encarga procesar estas expresiones regulares y genera un ficher Java que pueda ser compilado por Java y así crear el escaner.

Los escaners generados por Jax tienen entradas de búfer de tamaño arbitrario, y es al menos más conveniente para crear las tablas de tokens, Jax utiliza solo 7 bits de caracteres ASCII, y no permite código Unario.

La implementación de referencia de JAX-WS se desarrolla como un proyecto de código abierto y forma parte del proyecto GlassFish, un servidor de aplicaciones Java EE de código abierto. Se llama JAX-WS RI (por Reference Implementation) y se dice que es la implementación de calidad de producción (contrariamente a la implementación de referencia antigua que era una prueba de concepto). Esta implementación de referencia es ahora parte de la distribución Metro.

Ejemplo

Podríamos compilar y generar un fichero .war con esta clase para desplegarla en un Tomcat o similar, pero podemos directamente ponerle un *main()* para hacerla ejecutable. Para hacer pública la clase *UnWebService* como *Web Service* podemos usar la clase *EndPoint* que viene con *jax-ws*. El código quedaría así

```
package com.chuidiang.ejemplos.jax_ws;  
  
import javax.jws.WebMethod;  
  
import javax.jws.WebService;
```

```
import javax.xml.ws.Endpoint;
```

```
@WebService
```

```
public class UnWebService {
```

```
    @WebMethod
```

```
    public float suma(float a, float b, UnDato c) {
```

```
        return a + b;
```

```
    }
```

```
    public static void main(String[] args) {
```

```
        Endpoint.publish("http://localhost:8080/UnWebService", new UnWebService());
```

```
    }
```

```
}
```