



Servicios Web Seguros

Sesión 3: Desarrollo de servicios web



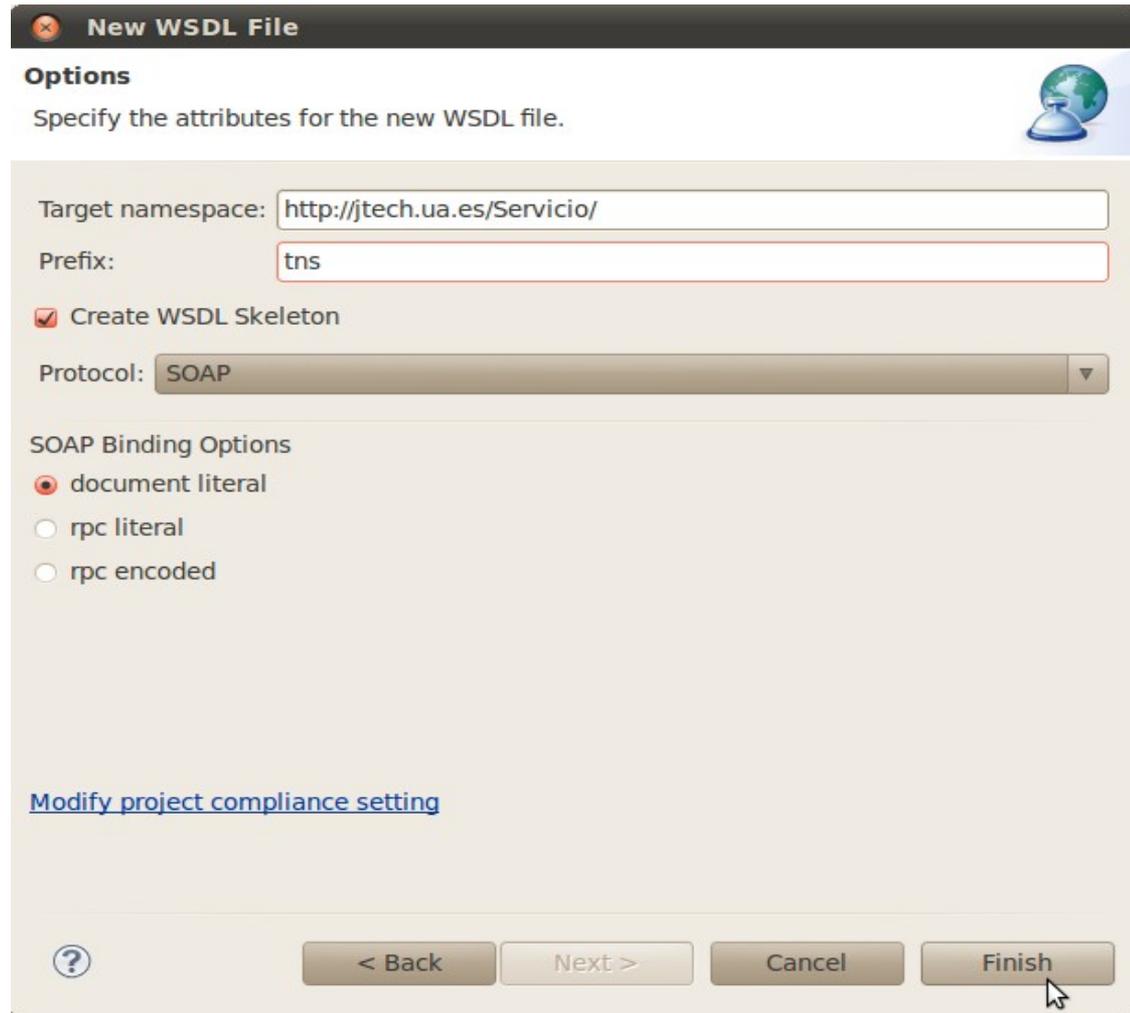
Puntos a tratar

- Diseño del documento WSDL
- Servicio con CXF
- Servicio con Axis2
- Instalación de Axis2



Diseño del WSDL

- Asistente en Eclipse





Diseño del WSDL

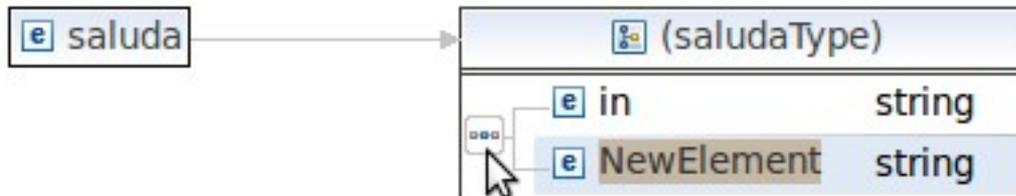
- Binding con la operación “saluda”





Diseño del WSDL

- Tipo del parámetro “saluda”





Servicio con CXF

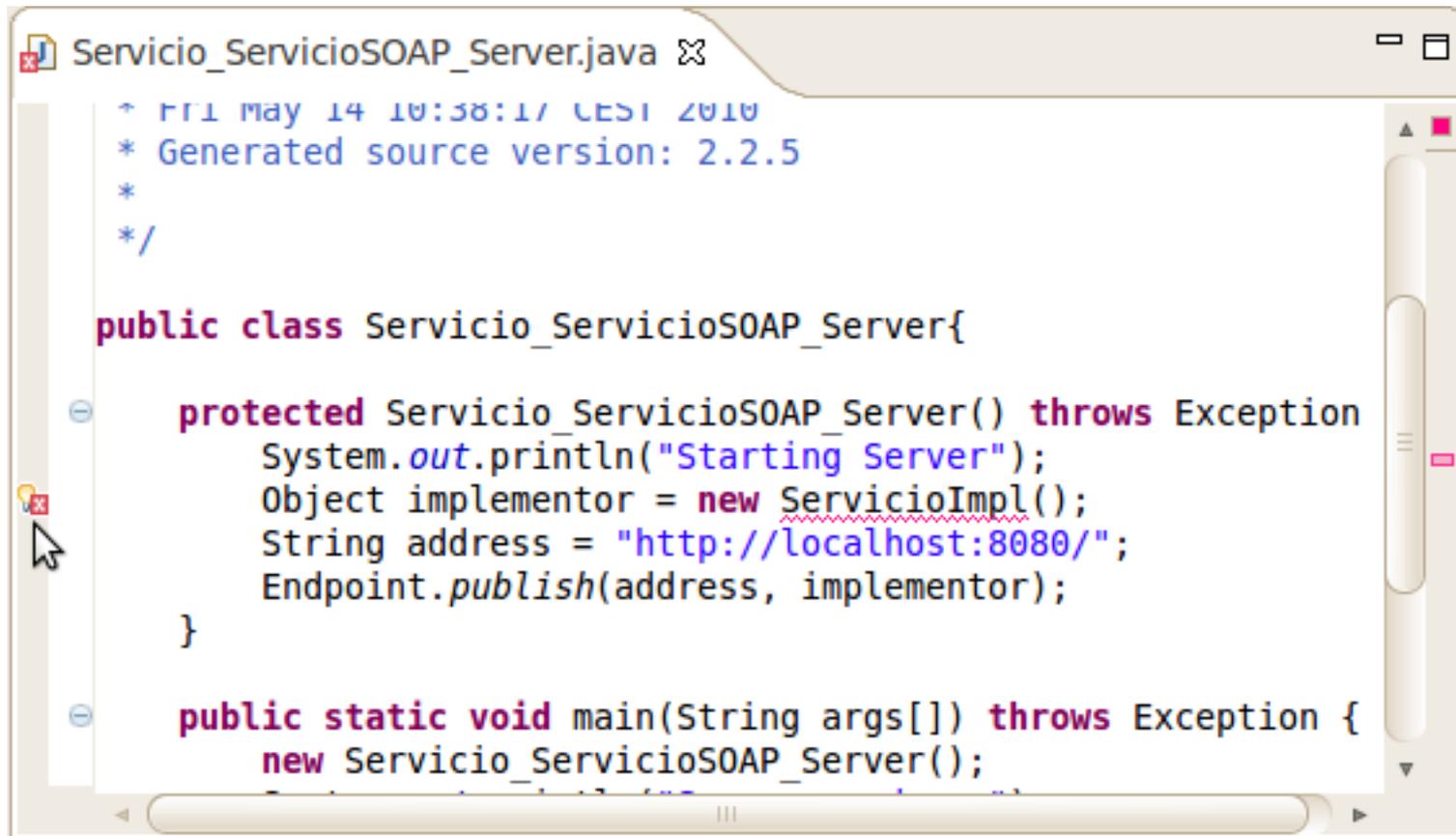
- Generación de código de servidor

```
WSDLToJava.main(new String[]{
    "-server",
    "-d", "src/main/java",
    "src/main/resources/Servicio.wsdl"
});
```



Implementación del servicio

- La implementación no se genera



```

Servicio_ServicioSOAP_Server.java
+ Fri May 14 10:58:17 CEST 2010
* Generated source version: 2.2.5
*
*/

public class Servicio_ServicioSOAP_Server{

    protected Servicio_ServicioSOAP_Server() throws Exception
        System.out.println("Starting Server");
        Object implementor = new ServicioImpl();
        String address = "http://localhost:8080/";
        Endpoint.publish(address, implementor);
    }

    public static void main(String args[]) throws Exception {
        new Servicio_ServicioSOAP_Server();
    }
}

```



Implementación del servicio

- La implementación la creamos manualmente

```
package es.ua.jtech.servicio;
public class ServicioImpl implements Servicio {
    @Override
    public String saluda(String nombre, String apellido) {
        return "¡Hola, " + nombre.substring(0,1) + ". " + apellido + "!";
    }
}
```

Ejecutar el servicio

```
Servicio_ServicioSOAP_Server [Java Application] /opt/jdk1.6.0_20/bin/java (14/05/2010 16:39:07)
INFO: Setting the server's publish address to be http://localhost:8080/
14-may-2010 16:39:07 org.mortbay.log.Slf4jLog info
INFO: Logging to org.slf4j.impl.JDK14LoggerAdapter(org.mortbay.log) via org.mortbay.log.Slf4jLog
14-may-2010 16:39:07 org.mortbay.log.Slf4jLog info
INFO: jetty-6.1.21
14-may-2010 16:39:07 org.mortbay.log.Slf4jLog info
INFO: Started SelectChannelConnector@localhost:8080
Server ready...
```

- Una vez ejecutado, podemos acceder a su WSDL con el navegador



WSDL publicado por el servicio

```
-<wSDL:definitions name="ServicioImplService" targetNamespace="http://servicio.jtech.ua.es/">
  <wSDL:import location="http://localhost:8080/saluda?wsdl=Servicio.wsdl" namespace="http://jtech.ua.es/Servicio/"> </wSDL:import>
  -<wSDL:binding name="ServicioImplServiceSoapBinding" type="ns1:Servicio">
    <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
    -<wSDL:operation name="saluda">
      <soap:operation soapAction="http://jtech.ua.es/Servicio/saluda" style="document"/>
      -<wSDL:input name="saluda">
        <soap:body use="literal"/>
      </wSDL:input>
      -<wSDL:output name="saludaResponse">
        <soap:body use="literal"/>
      </wSDL:output>
    </wSDL:operation>
  </wSDL:binding>
  -<wSDL:service name="ServicioImplService">
    -<wSDL:port binding="tns:ServicioImplServiceSoapBinding" name="ServicioImplPort">
      <soap:address location="http://localhost:8080/saluda"/>
    </wSDL:port>
  </wSDL:service>
</wSDL:definitions>
```

Terminado



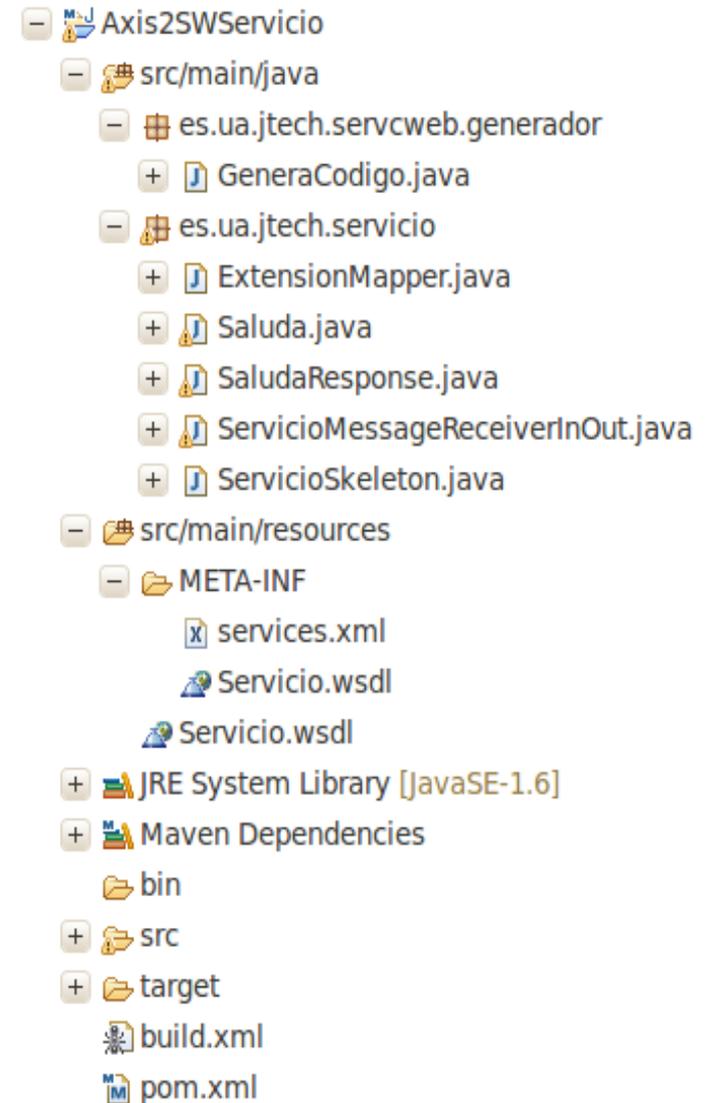
Servicio con Axis2

- WDSL → Generación de código → Implementación del servicio → Empaquetar en .aar → Despliegue del .aar en el servidor de Axis2 (que a su vez puede estar desplegado en un servidor de aplicaciones Java).

Servicio con Axis2

- Generación de código

```
package es.ua.jtech.servcweb.generador;
import org.apache.axis2.wsdl.WSDL2Code;
public class GeneraCodigo {
    public static void main(String[] args)
        throws Exception {
        WSDL2Code.main(new String[]{
            "-ss",
            "-sd",
            "-uw",
            "-S", "src/main/java",
            "-R", "src/main/resources/META-INF",
            "-uri", "src/main/resources/Servicio.wsdl"
        });
    }
}
```





Servicio con Axis2

- Generación paquete .aar (mismo que .jar)

```
[...]
[INFO] --- maven-jar-plugin:2.2:jar (default-jar) @
                                             SWServicio ---
[INFO] Building jar: /home/servicios/workspace/
Axis2SWServicio/target/SWServicio-0.0.1-SNAPSHOT.jar
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 3.583s
[INFO] Finished at: Sat May 15 20:17:50 CEST 2010
[INFO] Final Memory: 7M/17M
[INFO] -----
```



Instalación de Axis2

- Descargar desde <http://ws.apache.org/axis2/>
- Descomprimir
- Arrancar `/axis2-1.5.1/bin/axis2server.sh`

```
[INFO] Deploying module: addressing-1.5.1 - file:/home/servicios/
      axis2-1.5.1/repository/modules/addressing-1.5.1.mar
[INFO] Deploying module: metadataExchange-1.5.1 - file:/home/
      servicios/axis2-1.5.1/lib/mex-1.5.1.jar
[INFO] Deploying Web service: version.aar - file:/home/servicios/
      axis2-1.5.1/repository/services/version.aar
[INFO] [SimpleAxisServer] Started
[SimpleAxisServer] Started
[INFO] Listening on port 8080
```



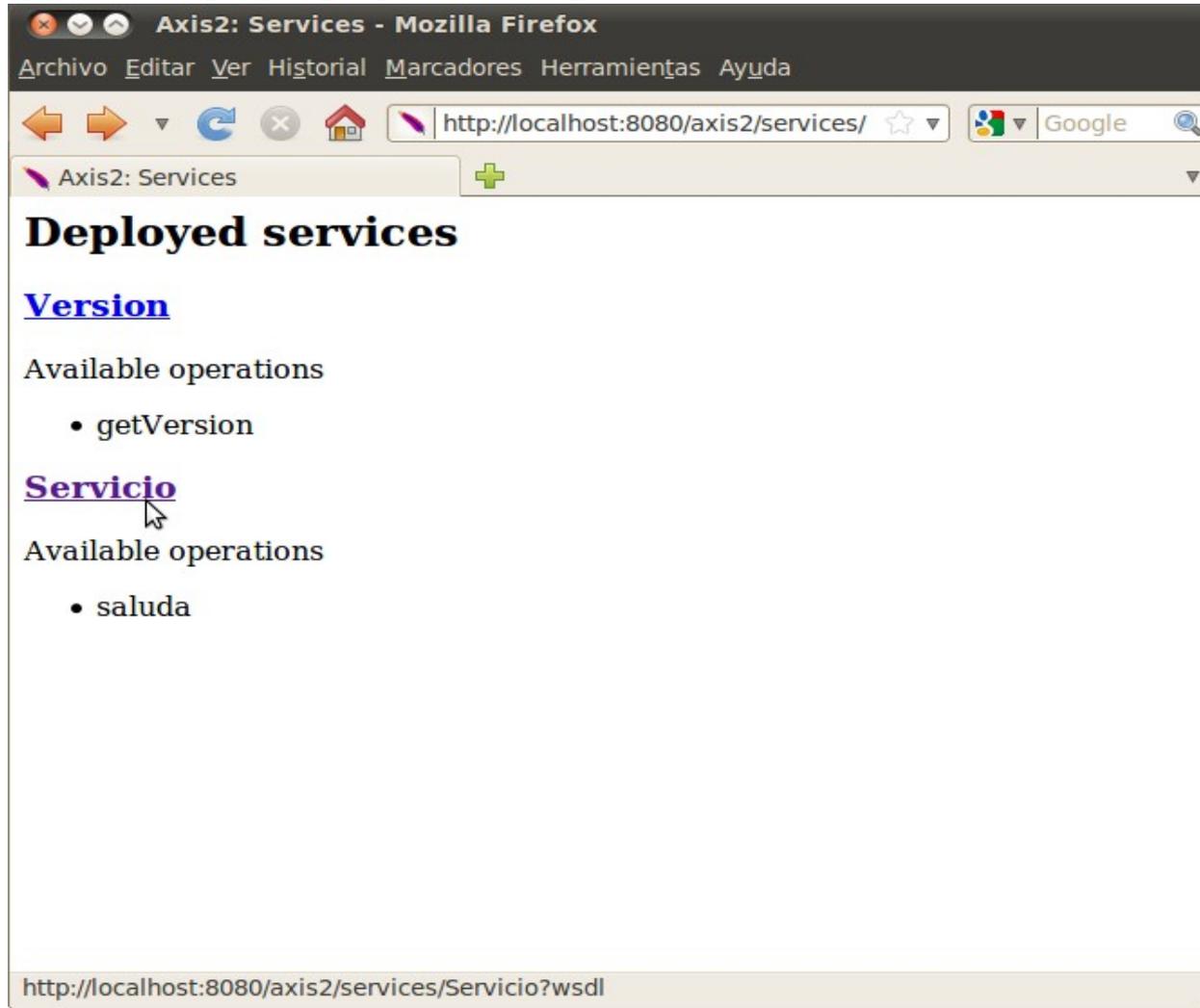
Desplegar en Axis2

- Copiar el .aar (o .jar) a
axis2/-1.5.1/repository/services/

```
[INFO] Deploying Web service: SWServicio.aar -  
file:/home/servicios/axis2-1.5.1/repository/services/SWServicio.aar
```



Interfaz web de Axis2





Invocar servicio desplegado en Axis2

- Debe acceder al endpoint que Axis2 le ofrece al servicio, en la carpeta /axis2/services/

```
package es.ua.jtech.servcweb.llamada;
import java.rmi.RemoteException;
import es.ua.jtech.servicio.ServicioStub;
public class SWCliente {
    public static void main(String[] args) throws RemoteException {
        ServicioStub servicio = new ServicioStub(
            "http://localhost:8080/axis2/services/Servicio");
        System.out.println(servicio.saluda("Boyan", "Bonev"));
    }
}
```



¿Preguntas...?