

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
Gestor de contenidos (Alfresco)
Aplicaciones híbridas

Tareas programadas (Quartz)
Gestor documental (Alfresco)
Inversión de control (Spring)

Control de autenticación y
acceso (Spring Security)
UDDI
Web Services
Rest Services
Social SSO
SSO (Cas)

JPA-Hibernate, MyBatis
Motor de búsqueda empresarial (Solr)
ETL (Talend)

Dirección de Proyectos Informáticos.
Metodologías ágiles
Patrones de diseño
TDD

BPM (jBPM o Bonita)
Generación de informes (JasperReport)
ESB (Open ESB)



Powered by autentia

Hosting patrocinado por

enredados

[Inicio](#)

[Quienes somos](#)

[Tutoriales](#)

[Formación](#)

[Empleo](#)

[Colabora](#)

[Comunidad](#)

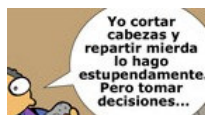
[Libro de Visitas](#)

[Comic](#)

NUEVO ¿Quieres saber cuánto ganas en relación al mercado? pincha aquí...

[Ver cursos que ofrece Autentia](#)

[Descargar comics en PDF y alta resolución](#)



[¡NUEVO!] 2008-04-01



2008-03-25



2008-03-17



2008-03-11

Estamos escribiendo un libro sobre la profesión informática y estas viñetas formarán parte de él. Puedes opinar en la sección [comic](#).

Tutorial desarrollado por



Carlos García Pérez

Creador del pionero Web [MobileTest](#).

Consultor tecnológico en el desarrollo de proyectos informáticos.

Ingeniero Técnico en Informática *

Puedes encontrarme en [Autentia](#)

Somos expertos en Java/J2EE

Catálogo de servicios de Autentia

[Descargar \(6,2 MB\)](#)

[Descargar en versión comic \(17 MB\)](#)

[AdictosAlTrabajo.com](#) es el Web de difusión de conocimiento de [Autentia](#).



[Catálogo de cursos](#)

Descargar este documento en formato PDF: [axiswssample.pdf](#)

Fecha de creación del tutorial: 2008-04-04

Axis2. Un completo ejemplo de creación de un servicio Web

En este tutorial vamos a realizar las siguientes tareas:

- Definición de una interface Java a partir de la cual generaremos el servicio Web.
- Generar la definición del servicio Web (WSDL) a partir del interface.
- Generar el esqueleto (Skeleton) del servicio Web del WSDL.
- Implementar la lógica de negocio del servicio Web.
- Generar el archivo del Servicio Web AAR (Axis Archive)
- Desplegar el servicio Web.
- Generar un cliente.
- Invocar el servicio Web.

Definición de una interface Java a partir de la cual generaremos el servicio Web

```
package com.autentia.ws.sample1;

/**
 * Interfaz Calculadora
 * @author Carlos García.
 * @see http://www.mobiletest.es
 */
public interface Calculadora {
    public int sumar(int op1, int op2);
}
```

Generar la definición del servicio Web (WSDL) a partir del interface.

```
%AXIS_HOME%/bin/java2wsdl -of calculadora.wsdl -cn com.autentia.ws.sample1.Calculadora
```

- En donde el parámetro **of** es el nombre del archivo WSDL que deseamos generar
- En donde el parámetro **cn** es el nombre de la clase a partir de la cual deseamos generar el WSDL.

Archivo generado: [calculadora.xsd1](#)

Catálogo de servicios Autentia (PDF 6,2MB)



[En formato comic...](#)



☐ Web

☒ [www.adictosaltrabajo.com](#)

Últimos tutoriales

2008-04-05
[JMX y monitorización de JBoss](#)

2008-04-05
[Jersey: la implementación de RESTful de Sun](#)

2008-04-05
[Metro: pila de webservices de Sun. Integración con Maven 2](#)

2008-04-05
[Metro: pila de webservices de Sun.](#)

2008-04-04
[Espectaculares efectos visuales en el escritorio de Linux, con Compiz Fusion](#)

2008-04-04
[Monitorización de Web Services con Glassfish Wsmonitor](#)

2008-04-04
[Axis2. Ejemplo de creación de un servicio Web](#)

2008-04-03
[Servicios Web RESTful en Axis 2](#)

2008-04-03
[XML Signature - Firma Digital sobre XML](#)

2008-04-03
[XML Encryption, Criptografía sobre XML](#)

```

<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/" xmlns:ns0="http://sample1.ws.autentia.com"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
  xmlns:ns1="http://org.apache.axis2/xsd" xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  targetNamespace="http://sample1.ws.autentia.com" >
  <wsdl:types>
    <xs:schema xmlns:ns="http://sample1.ws.autentia.com" attributeFormDefault="qualified"
      elementFormDefault="qualified" targetNamespace="http://sample1.ws.autentia.com" >
      <xs:element name="sumar">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" name="param0" type="xs:int"/>
            <xs:element minOccurs="0" name="param1" type="xs:int"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="sumarResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" name="return" type="xs:int"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:schema>
  </wsdl:types>
  <wsdl:message name="sumarRequest">
    <wsdl:part name="parameters" element="ns0:sumar"/>
  </wsdl:message>
  <wsdl:message name="sumarResponse">
    <wsdl:part name="parameters" element="ns0:sumarResponse"/>
  </wsdl:message>
  <wsdl:portType name="CalculadoraPortType">
    <wsdl:operation name="sumar">
      <wsdl:input message="ns0:sumarRequest" wsaw:Action="urn:sumar"/>
      <wsdl:output message="ns0:sumarResponse" wsaw:Action="urn:sumarResponse"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="CalculadoraSOAP11Binding" type="ns0:CalculadoraPortType">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
    <wsdl:operation name="sumar">
      <soap:operation soapAction="urn:sumar" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:binding name="CalculadoraSOAP12Binding" type="ns0:CalculadoraPortType">
    <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
    <wsdl:operation name="sumar">
      <soap12:operation soapAction="urn:sumar" style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:binding name="CalculadoraHttpBinding" type="ns0:CalculadoraPortType">
    <http:binding verb="POST"/>
    <wsdl:operation name="sumar">
      <http:operation location="Calculadora/sumar"/>
      <wsdl:input>
        <mime:content type="text/xml" part="sumar"/>
      </wsdl:input>
      <wsdl:output>
        <mime:content type="text/xml" part="sumar"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="Calculadora">
    <wsdl:port name="CalculadoraSOAP11port_http" binding="ns0:CalculadoraSOAP11Binding">
      <soap:address location="http://localhost:8080/axis2/services/Calculadora"/>
    </wsdl:port>
    <wsdl:port name="CalculadoraSOAP12port_http" binding="ns0:CalculadoraSOAP12Binding">
      <soap12:address location="http://localhost:8080/axis2/services/Calculadora"/>
    </wsdl:port>
    <wsdl:port name="CalculadoraHttpport" binding="ns0:CalculadoraHttpBinding">
      <http:address location="http://localhost:8080/axis2/services/Calculadora"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>

```

Últimas ofertas de empleo

2008-04-04
Banca - Genérico - MADRID.

2008-04-03
Banca - Genérico - MADRID.

2008-04-02
T. Información - Analista / Programador - MADRID.

2008-04-02
T. Información - Analista / Programador - MADRID.

2008-03-29
T. Información - Analista / Programador - MADRID.

Anuncios Google

Generar el esqueleto (Skeleton) del servicio Web a partir del WSDL.

```
%AXIS_HOME%/bin/wsdl2java -p com.autentia.ws.sample1.servidor -S . -or -ss -sd -u
--noMessageReceiver -uri calculadora.wsdl
```

- El parámetro **p** indica el nombre del paquete donde deseamos que se generen las clases.
- El parámetro **S** indica el nombre de la carpeta en donde deseamos que se generen las clases (por defecto crea una carpeta src)
- El parámetro **or** indica que se sobrescriban los archivos.
- El parámetro **ss** indica que se generen las clases del servicio web (Por defecto NO se generan)
- El parámetro **sd** indica que se genere el descriptor de despliegue del servicio web.
- El parámetro **u** indica que no se generen innerclases sino que cada clase está en un archivo propio.
- El parámetro **noMessageReceiver** indica que no se genere un Message Receiver específico para el servicio Web.
- El parámetro **uri** indica el WSDL de partida.

El descriptor de despliegue generado (**services.xml**):

```
<?xml version="1.0" encoding="UTF-8"?>
<serviceGroup>
  <service name="Calculadora">
    <parameter name="ServiceClass">com.autentia.ws.sample1.servidor.CalculadoraSkeleton </parameter>
    <operation name="sumar">
      <actionMapping>urn:sumar</actionMapping>
      <outputActionMapping>urn:sumarResponse</outputActionMapping>
      <messageReceiver class="org.apache.axis2.rpc.receivers.RPCMessageReceiver" />
    </operation>
  </service>
</serviceGroup>
```

Implementar la lógica de negocio del servicio Web.

```
/**
 * Implementación del Servicio Web
 * @author Carlos García.
 * @see Creador de http://www.mobiletest.es
 */
package com.autentia.ws.sample1.servidor;
public class CalculadoraSkeleton {
    public com.autentia.ws.sample1.SumarResponse sumar(com.autentia.ws.sample1.Sumar sumar) {
        com.autentia.ws.sample1.SumarResponse response = new com.autentia.ws.sample1.SumarResponse();
        response.set_return(sumar.getParam0() + sumar.getParam1());
        return response;
    }
}
```

Generar el archivo del Servicio Web AAR (Axis Archive)

Para desplegar un servicios Web en Axis2, debemos generar un archivo AAR.

Los archivos AAR, son como los archivos WAR, EAR. Es decir son archivos JAR o ZIP que contienen archivos y directorios bajo una estructura definida.

Los pasos generar el archivo aar son:

1. Creamos una carpeta de nombre *Calculadora*
2. Dentro de la carpeta *Calculadora* copiamos las clases *com.autentia.ws.sample1* manteniendo la estructura.
3. Dentro de la carpeta *Calculadora* creamos una carpeta de nombre *META-INF* y copiamos dentro el archivo *services.xml*.
4. Empaquetamos la carpeta *Calculadora* en un archivo *Calculadora.aar*. Para ello estando dentro del directorio *Calculadora* ejecutamos la siguiente instrucción:

```
%JAVA_HOME%/bin/jar -cvf Calculadora.aar *
```

Desplegar el servicio Web.

Existen varios métodos de despliegue de un Servicio Web en Axis2, los más comunes son:

1. A través la opción "Upload Service" de la [página de administración de Axis2](#).
2. Copiar el "AAR" al directorio `<AXIS_WEBAPP_HOME>\repository\services`. Siendo `AXIS_WEBAPP_HOME` el directorio donde está desplegado Axis2 en el servidor de aplicaciones o contenedor de Servlet.

Generar un cliente

Vamos a generar el cliente desde la propia descripción del servicio Web desplegado.

```
%AXIS_HOME%/bin/wsd12java -p com.autentia.ws.sample1.cliente -d adb -S . -or -uri
http://127.0.0.1:8080/axis2/services/Calculadora?wsdl
```

- El parámetro **p** indica el nombre del paquete donde deseamos que se generen las clases.
- El parámetro **d** indica el tipo de Databinding, es decir, el mapeo entre XML y objetos Java que deseamos. Usamos el tipo **ADB** (Axis Data Binding) que usa **AXIOM** (Axis Object Model) un modelo especialmente creado para Axis que proporciona un alto rendimiento debido a que está basado en el API de **StAX** (Streaming API for XML) (Alternativas a DOM y SAX) una forma de parsear XML **bajo demanda**, es decir, sólo se parsea lo que se necesita en cada momento.
- El parámetro **S** indica el nombre de la carpeta en donde deseamos que se generen las clases (por defecto crea una carpeta src)
- El parámetro **uri** indica el WSDL de partida.

Invocar el servicio Web

```

package com.autentia.ws.sample1.test;

import com.autentia.ws.sample1.cliente.*;
import com.autentia.ws.sample1.cliente.CalculadoraStub.Sumar;
import com.autentia.ws.sample1.cliente.CalculadoraStub.SumarResponse;

/**
 * Ejemplo de invocación SINCRONA del servicio Web
 * @author Carlos García.
 * @see Creador de http://www.mobiletest.es
 */
public class Appication1 {
    /**
     * Ejemplo de invocación del servicio
     */
    public static void main(String[] args) throws Exception {
        CalculadoraStub stub = new CalculadoraStub();
        Sumar operacion = new Sumar();
        SumarResponse response = null;

        // Establecemos Los parámetros de la operación
        operacion.setParam0(100);
        operacion.setParam1(200);

        // Invocamos el WS
        response = stub.sumar(operacion);

        // Mostramos el resultado
        System.out.println(response.get_return());
    }
}

```

Conclusiones

Bueno, como podéis ver la generación de servicios Web con Axis2 así como su despliegue o generación de clientes no tiene mucha dificultad.

Axis2 soporta los patrones de mensajería (Síncrona/Asíncrona, Comunicación 1 a 1, 1 a N, N a M, mensajería por eventos, etc.) definidos WSDL 2, así como varios tipos de DataBinding que se salen del alcance de este tutorial.

Un saludo, Carlos García.

- Puedes opinar sobre este tutorial [haciendo clic aquí](#).
- Puedes firmar en nuestro libro de visitas [haciendo clic aquí](#).
- Puedes asociarte al grupo AdictosAlTrabajo en XING [haciendo clic aquí](#).
- Añadir a favoritos Technorati. 



Esta obra está licenciada bajo [licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

Recuerda

Autentia te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#)). Somos expertos en: J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ... y muchas otras cosas.

¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?, ¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?

Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos ...

Autentia = Soporte a Desarrollo & Formación.

info@autentia.com

Formación en nuevas tecnologías

Servicio de notificaciones:

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales.

Formulario de suscripción a novedades:

E-mail

Tutoriales recomendados

Nombre	Resumen	Fecha	Visitas	pdf
--------	---------	-------	---------	-----

Nota:

Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento. Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores. En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo. Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.