

Arquitectura e Integración de Aplicaciones Empresariales

Décima Sesión Arquitectura Orientada a Servicios

Universidad Autónoma Metropolitana
Casa abierta al tiempo  Azcapotzalco

Dra. Maricela Bravo
Cubículo H-287-B
mari_clau_18@hotmail.com

¿Qué es un Servicio Web?

Un servicio Web es un componente de software que:

- ▶ Está disponible en la Web
- ▶ Emplea el sistema de mensajes estandarizado XML
- ▶ Es independiente del sistema operativo y del lenguaje de programación.

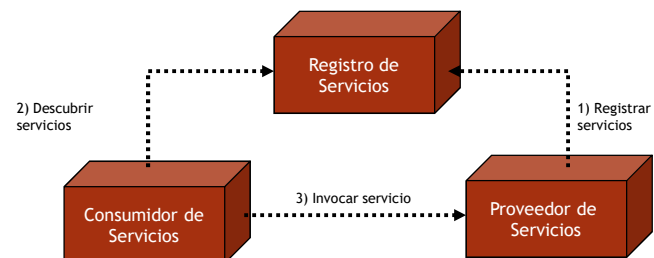
Arquitectura de los Servicios Web

Existen dos formas de estudiar la arquitectura de los servicios Web

1. A través de los roles de los servicios
2. A través de la pila de protocolos y estándares de los servicios

Roles de los Servicios Web

Existen tres roles principales



Pila de protocolos y estándares de los Servicios Web

Descubrimiento		Centralización de servicios
Descripción		Descripción de la interface de los servicios
Mensajes XML	XML-RPC, SOAP, XML	Codificación de mensajes en formato XML
Transporte	HTTP, SMTP, FTP, BEEP	Transporte de mensajes

Tipos de Mensajes XML

Existen dos formas de mensajes XML

- XML-RPC
- SOAP

¿Qué es XML-RPC ?

- ▶ Es un protocolo simple que utiliza mensajes en XML para realizar RPC
- ▶ Las solicitudes (request) son codificadas en XML y enviadas vía HTTP
- ▶ Las respuestas (response) son codificadas en XML y recibidas vía HTTP
- ▶ Esta es el protocolo de mensajes más sencillo de los servicios Web.

Ejemplo de XML-RPC Request

```
<methodCall>
  <methodName>
    com.agram.holaMundo
  </methodName>
  <params>
    <param>
      <value>Saludo</value>
    </param>
  </params>
</methodCall>
```

Ejemplo de XML-RPC Response

```
<methodResponse>
  <params>
    <param>
      <value>
        <string>Hola Mundo</string>
      </value>
    </param>
  </params>
</ methodResponse >
```

¿Qué es SOAP?

- ▶ Simple Object Access Protocol
- ▶ SOAP es un protocolo simple de acceso a objetos
- ▶ SOAP es un poco más complicado que XML-RPC
- ▶ SOAP es una extensión de XML-RPC
- ▶ Utiliza espacios de nombres y esquemas XML.

Mensaje SOAP

- ▶ **Envelope** es un envoltorio para el contenido
- ▶ **Header** es un elemento opcional que puede contener información del control
- ▶ **Body** este elemento incluye solicitudes y respuestas
- ▶ El elemento **Body** contendrá un elemento **Fault** para el caso de errores.



Ejemplo de SOAP Request

```
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

  <SOAP-ENV:Body>
    <ns1:holaMundo
      xmlns:ns1="http://agram.com/">
      <name xsi:type="xsd:string">Saludo</name>
    </ns1:holaMundo>
  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

Ejemplo de SOAP Response

```
<SOAP-ENV:Envelope
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

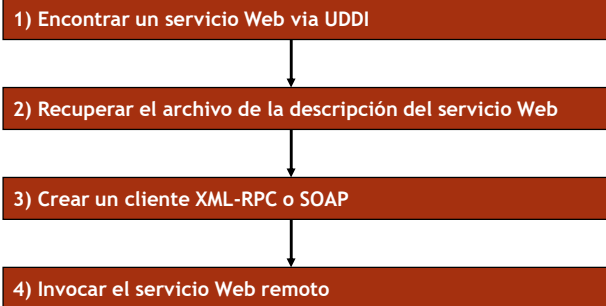
  <SOAP-ENV:Body>
    <ns1:holaMundoResponse
      xmlns:ns1="http://agram.com">
      <result xsi:type="xsd:string">Hola Mundo</result>
    </ns1:holaMundoResponse>
  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>
```

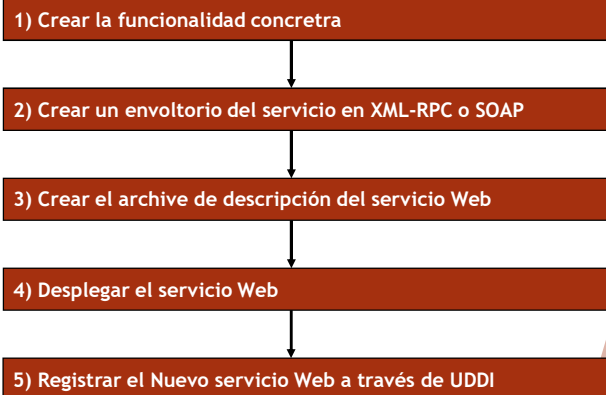
Repaso de WSDL

- ▶ **Web Services Description Language**
- ▶ **Elementos**
 1. **definitions** - define el nombre del servicio
 2. **types** - describe todos los tipos de datos que serán transmitidos
 3. **message** - define el nombre de los mensajes que serán transmitidos
 4. **portType** - define las operaciones
 5. **binding** - define cómo será transmitido el mensaje
 6. **service** - define dónde se localiza el servicio

Plan de desarrollo del Service Requestor



Plan de desarrollo del Service Provider



Axis 2

¿Qué es Apache Axis?

“Axis es esencialmente un motor SOAP - un entorno de desarrollo para construir procesos SOAP tales como clientes, servidores, etc. “

- Axis Website

Características de Apache Axis

- ▶ Es el sucesor de Apache SOAP
- ▶ Puede ejecutarse como un servidor standalone o como un servidor que se adjunta a un contenedor de servlets como Tomcat.
- ▶ Permite la generación automática de WSDL a partir de servicios desplegados.
- ▶ Java2WSDL - para generar WSDL desde Java interface
- ▶ WSDL2Java - para generar Java clases a partir de WSDL
- ▶ Fácil despliegue de servicios
- ▶ Se basa en JAX-RPC API

Instalación de Apache Axis

1) Descargar la distribución de Axis desde

<http://xml.apache.org/axis/>

- La distribución de Axis contiene una aplicación Web (webapps/axis/ directory) para adjuntar el servidor de Axis en un contenedor de servlets como Tomcat.

Introducción

- ▶ Apache Axis 2 es un motor de servicios Web de tercera generación.
- ▶ El proyecto Apache Axis2 es una implementación para clientes y proveedores de servicios Web basada en Java.
- ▶ Apache Axis2 proporciona un modelo orientado a objetos y una arquitectura completamente modular.
- ▶ Apache Axis2 te permite realizar las siguiente funciones:

¿Qué se puede hacer con Axis2?

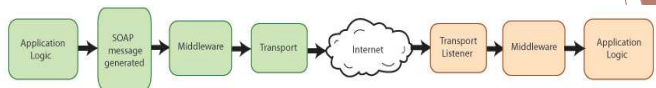
- ▶ Enviar mensajes SOAP
- ▶ Recibir y procesar mensajes SOAP
- ▶ Crear un servicios a partir de una clase de Java
- ▶ Crear calses de implementación para el proveedor y cliente de servicios utilizando WSDL.
- ▶ Facilmente recuperar el WSDL de un servicio
- ▶ Enviar y recibir mensajes SOAP con adjuntos
- ▶ Crear o utilizar servicios REST
- ▶ Entre otros...

Ciclo de vida de un mensaje de servicio Web con Axis2



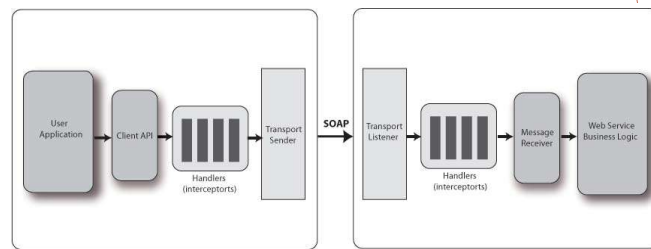
- ▶ La aplicación que envía crea el mensaje SOAP original, el cual es un mensaje XML que consiste de cabecera y un cuerpo.
- ▶ Si el sistema requiere del uso de alguna otra recomendación WS (WS-Addressing o WS-Security), entonces el mensaje debe pasar por otros procesamientos antes de ser enviado.
- ▶ Una vez que el mensaje está listo, se envía a través de un transporte particular como HTTP, JMS, u otro.

Ciclo de vida de un mensaje de servicio Web con Axis2



- ▶ El mensaje encuentra su camino hacia el receptor, el cual toma el mensaje a través de un “transport listener”.
- ▶ Si el mensaje requiere el uso de WS-Security u otra recomendación, necesitará procesamiento adicional con el propósito de verificar las credenciales o la descriptación de información sensible.
- ▶ Finalmente, un despachador determina la aplicación específica para el cual estaba destinado el mensaje, y se lo envía a ese componente.

Manejo de mensajes SOAP con Axis2



Manejo de mensajes SOAP con Axis2

- ▶ En cada lado, se tiene una aplicación diseñada para trabajar con los mensajes enviados o recibidos. En medio, se tiene Axis2.
- ▶ El valor de los servicios Web es que el que envía y el que recibe no necesitan estar en la misma plataforma, mucho menos ejecutando la misma aplicación.
- ▶ Asumiendo que Axis2 se está ejecutando en ambos lados, el proceso se realiza de la siguiente forma:

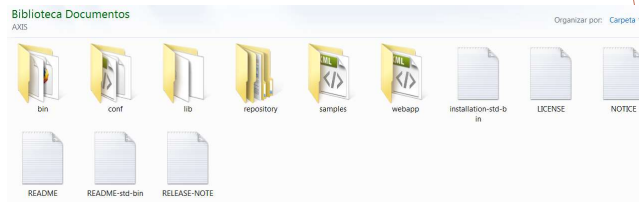
Manejo de mensajes SOAP con Axis2

- ▶ El que envia crea el mensaje SOAP.
- ▶ Los manejadores de Axis realizan las acciones necesarias sobre el mensaje, tales como la encriptación de mensajes con WS-Security.
- ▶ El transport sender envia el mensaje.
- ▶ En el lado del receptor, el transport listener detecta el mensaje.
- ▶ El transport listener pasa el mensaje a cualquier manejador en el lado receptor.
- ▶ Una vez que el mensaje ha sido procesado en la fase de "pre-despacho" es enviado a los despachadores, el cual es pasado a la aplicación apropiada.

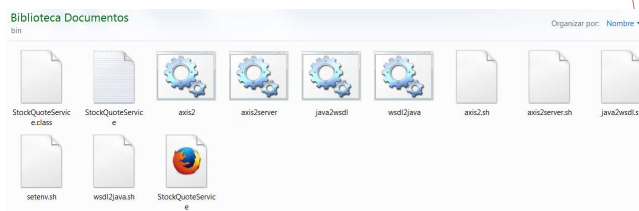
Distribuciones Axis2

1. Axis2 Standard Binary Distribution
2. axis2.war Distribution Directory Hierarchy

Axis2 Standard Binary Distribution



Axis2 Standard Binary Distribution



Configuración del entorno

1. Descargar e instalar Java
2. Establecer la variable de entorno JAVA_HOME con la ruta donde quedó el JDK instalado
3. Descargar Axis2 y extraerlo en algún directorio.
4. Copiar axis2.war al directorio webapps de tu motor de servlets.
5. Establecer la variable de entorno AXIS2_HOME hacia el directorio donde quedó instalado.
6. Los usuarios de Linux pueden alternativamente ejecutar setenv.sh file in the AXIS2_HOME/bin directory to set the AXIS2_HOME environment variable to the pathname of the extracted directory of Axis2.

Axis2 Standard Binary Distribution

- ▶ Levantando Axis2 Standalone Server
- ▶ El servidor Axis2 puede levantarse ejecutando alguno de los siguientes comandos:
%AXIS2_HOME%\bin\axis2server.bat (Windows)
\$AXIS2_HOME/bin/axis2server.sh (Unix)
- ▶ Para verificarlo hay que visitar
 - ▶ <http://localhost:8080/axis2/services/>

Tarea

1. Descargar e instalar la distribución binaria de Axis2
2. Descargar e instalar la distribución de axis2.war y levantarla en Tomcat
3. Crear un servicio Web de la calculadora (suma, resta, multiplicación y división) y desplegar el servicio (rol de proveedor de servicios) con ambas distribuciones
4. Usar el comando java2WSDL para generar el wsdl de un programa de java
5. Usar el comando WSDL2Java para generar la referencia en objetos de una descripción de servicio Web.