

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas
Introducción a la programación 2
Ing. William Escobar
Aux. Robinson Perez

SERVICIO WEB

Oscar Eduardo Mazariegos López
201602469
Guatemala 16 de Junio 2018

INTRODUCCION

La programación en internet ha sido muy amplia desde su aparición, lo que ha provocado que esta cambie, por lo que presenta nuevas formas para la construcción de aplicaciones orientadas a servicios. Además cuenta con diferentes aplicaciones, servicios de web y tecnologías básicas para las mismas.

A continuación se presenta lo que es el servicio web, así como las partes y lenguajes que lo componen.

SERVICIO WEB

Es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software son desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma.

Los servicios web son por ende un conjunto de funcionalidad expuesta en una intranet o a través de Internet, por y para aplicaciones y computadoras sin la intervención humana.

CARACTERÍSTICAS

- **Utilización de estándares de internet**
Para que los servicios web puedan ser utilizados por varios sistemas heterogéneos existentes en Internet se emplea el protocolo de transferencia de datos HTTP utilizado por todos los navegadores Web y XML.
- **Basados en tecnologías de paso de mensajes**
La interacción entre el cliente y el proveedor del servicio es empaquetada en unidades autodescriptivas denominadas mensajes. Dicha interacción se describe en función de los mensajes intercambiados.
- **Combinan lo mejor de la tecnología de componentes y de la tecnología Web**
Los servicios Web presentan una funcionalidad de caja negra que puede ser reutilizada sin preocuparse de cómo es implementada y ello proporciona interfaces bien definidas.

APLICACIONES

- **Servicio simple**
El escenario típico es el de un servicio Web que proporciona algún tipo de funcionalidad a los clientes que lo utilizan.
- **Integración de aplicaciones**
Los servicios Web pueden ser empleados para integrar aplicaciones existente, lo que puede ser de gran utilidad en aquellos casos en los que se tiene una serie de aplicaciones a medida que se usan, pero que permanecen aisladas, de forma que es posible exponer la funcionalidad de dichas aplicaciones como un servicio Web, y ello permitiría la creación de aplicaciones compuestas que hicieran uso de dicha funcionalidad.

VENTAJAS

Los Servicios Web pueden ser muy útiles en ciertos casos concretos de programación. Los expertos de marketing de empresas de software como Microsoft anuncian a bombo y platillo una revolución debido a la aparición de esta tecnología.

La única revolución que se puede ver en relación con los Servicios Web es la de cómo los servidores de internet hablan entre ellos y cada día son más independientes del programador.

Arquitectura de los Servicios Web

Los servicios Web presentan una arquitectura orientada a servicios que permite crear una definición abstracta de un servicio, proporcionar una implementación concreta de dicho servicio, publicar y localizar un servicio, seleccionar un instancia de un servicio, y utilizar dicho servicio con una elevada interoperabilidad. Es posible desacoplar la implementación del servicio Web y su uso por parte de un cliente. También es posible desacoplar la implementación del servicio y de cliente. Las implementaciones concretas del servicio pueden desacoplarse a nivel de lógica y transporte.

El proveedor del servicio define la descripción abstracta de dicho servicio utilizando un lenguaje de descripción de Servicios Web (WSDL: Web Services Description Language:). A continuación se crea un Servicio concreto a partir de la descripción abstracta del servicio, produciendo así una descripción concreta del servicio en WSDL. Dicha descripción concreta puede entonces publicarse en un servicio de registro como por ejemplo UDDI (Universal Description, Discovery and Integration). Un cliente de un servicio puede utilizar un servicio de registro para localizar una descripción de un servicio, a partir de la cual podrá seleccionar y utilizar una implementación concreta de dicho servicio.

La descripción abstracta se define en un documento WSDL como un PortType. Una instancia concreta de un Servicio se define mediante un elemento port de un WSDL (consistente a su vez en una combinación de un PortType, un binding de codificación y transporte, más una dirección). Un conjunto de ports definen un elemento service de un WSDL.

Tecnologías básicas para Servicios Web

Tenemos una serie de tecnologías, todas ellas basadas en XML, que son fundamentales para el desarrollo de Servicios Web. Estas tecnologías son independientes tanto del SO como del lenguaje de programación utilizado para implementar dichos servicios. Por lo tanto, serán utilizadas para cualquier Servicio Web, independientemente de la plataforma sobre la que construyamos dichos servicios (como puede ser J2EE o .NET).

Los protocolos utilizados en los Servicios Web se organizan en una serie de capas:

Capa	Descripción
Transporte de servicios	Es la capa que se encarga de transportar los mensajes entre aplicaciones. Normalmente se utiliza el protocolo HTTP para este transporte, aunque los servicios web pueden viajar mediante otros protocolos de transferencia de hipertexto como SMTP, FTP o BEEP.
Mensajería XML	Es la capa responsable de codificar los mensajes en XML de forma que puedan ser entendidos por cualquier aplicación. Puede implementar los protocolos XML-RPC o SOAP .
Descripción de servicios	Se encarga de definir la interfaz pública de un determinado servicio. Esta definición se realiza mediante WSDL .
Localización de servicios	Se encarga del registro centralizado de servicios, permitiendo que estos sean anunciados y localizados. Para ello se utiliza el protocolo

TIPOS DE SERVICIOS

Un servicio es un componente software proporcionado a través de un endpoint accesible a través de la red. Los servicios productores y consumidores utilizan mensajes para intercambiar información de invocaciones de petición y respuesta en forma de documentos auto-contenidos que hacen muy pocas asunciones sobre las capacidades tecnológicas de cada uno de los receptores.

A nivel técnico, los servicios pueden implementarse de varias formas. En este sentido, podemos distinguir dos tipos de servicios Web: los denominados servicios Web "grandes", los llamaremos servicios Web SOAP, y servicios Web RESTful.

Web SOAP

Utilizan mensajes XML para comunicarse que siguen el estándar SOAP (Simple Object Access Protocol), un lenguaje XML que define la arquitectura y formato de los mensajes. Dichos sistemas normalmente contienen una descripción legible por la máquina de la descripción de las operaciones ofrecidas por el servicio, escrita en WSDL (Web Services Description Language), que es un lenguaje basado en XML para definir las interfaces sintácticamente.

SOAP

Se trata de un protocolo derivado de XML que nos sirve para intercambiar información entre aplicaciones.

Normalmente utilizaremos SOAP para conectarnos a un servicio e invocar métodos remotos, aunque puede ser utilizado de forma más genérica para enviar cualquier tipo de contenido. Podemos distinguir dos tipos de mensajes según su contenido:

- **Mensajes orientados al documento:** Contienen cualquier tipo de contenido que queramos enviar entre aplicaciones.
- **Mensajes orientados a RPC:** Este tipo de mensajes servirá para invocar procedimientos de forma remota (Remote Procedure Calls). Podemos verlo como un tipo más concreto dentro del tipo anterior, ya que en este caso como contenido del mensaje especificaremos el método que queremos invocar junto a los parámetros que le pasamos, y el servidor nos deberá devolver como respuesta un mensaje SOAP con el resultado de invocar el método.
Puede ser utilizado sobre varios protocolos de transporte, aunque está especialmente diseñado para trabajar sobre HTTP.

Dentro del mensaje SOAP, encontramos los siguientes elementos:

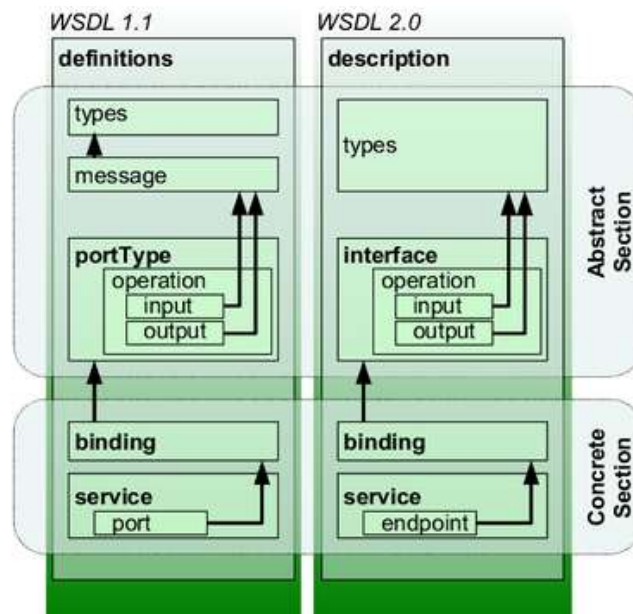
- **Un sobre (Envelope).** Describe el mensaje, a quien va dirigido, y cómo debe ser procesado. El sobre incluye las definiciones de tipos que se usarán en el documento. Contiene una cabecera de forma opcional, y el cuerpo del mensaje.
- **Una cabecera (Header) opcional.** Se puede incluir información sobre el mensaje. Por ejemplo, podemos especificar si el mensaje es obligatorio (debe ser entendido de forma obligatoria por el destinatario), e indicar los actores (lugares por donde ha pasado el mensaje).
- **El cuerpo del mensaje (Body).** Contiene el mensaje en sí. En el caso de los mensajes RPC se define una convención sobre cómo debe ser este contenido, en el que se especificará el método al que se invoca y los valores que se pasan como parámetros. Puede contener un error de forma opcional.

- **Un error (Fault) en el cuerpo del mensaje de forma opcional.** Nos servirá para indicar en una respuesta SOAP que ha habido un error en el procesamiento del mensaje de petición que mandamos.



WSDL

Es un lenguaje basado en XML utilizado para describir la funcionalidad que proporciona un servicio Web. Una descripción WSDL (fichero WSDL) de un servicio web proporciona una descripción entendible por la máquina (machine readable) de la interfaz del servicio Web, indicando cómo se debe llamar al servicio, qué parámetros espera, y qué estructuras de datos devuelve.



WSDL describe un servicio utilizando varios elementos (etiquetas xml). Dichos elementos podemos clasificarlos como abstractos o concretos. La parte WSDL abstracta describe las operaciones y mensajes con detalle. En otras palabras, la parte abstracta de un WSDL especifica QUÉ hace el servicio:

- Qué operaciones están disponibles
- Qué entradas, salidas, y mensajes de error tienen las operaciones
- Cuáles son las definiciones de los tipos para los mensajes de entrada, salida y error

En el mundo Java, podemos pensar en la parte abstracta de un WSDL como en la definición de una interfaz o una clase abstracta, con la definición de sus métodos, pero no sus implementaciones. La parte abstracta de un WSDL contiene dos componentes principales:

Las operaciones que forman la definición de la interfaz. Los tipos de datos para los parámetros de entrada, salida y error, de las operaciones. La parte WSDL concreta describe el cómo y dónde del servicio:

- Cómo tiene que llamar un cliente al servicio
- Qué protocolo debería usar
- Dónde está disponible el servicio

En el mundo Java podemos pensar en la parte concreta de un WSDL como en la implementación de la parte abstracta, aunque en términos de servicios Web, solamente describe dónde se encuentra dicha implementación para utilizarse. La parte concreta de un WSDL contiene dos componentes principales:

- Información de *enlazado* (*binding*) sobre el protocolo a utilizar
- La dirección en donde localizar el servicio

Web RESTful

Los servicios Web RESTful (Representational State Transfer Web Services) son adecuados para escenarios de integración básicos ad-hoc. Dichos servicios Web se suelen integrar mejor con HTTP que los servicios basados en SOAP, ya que no requieren mensajes XML o definiciones del servicio en forma de fichero WSDL

Los servicios Web REST utilizan estándares muy conocidos como HTTP, SML, URI, MIME, y tienen una infraestructura "ligera" que permite que los servicios se construyan utilizando herramientas de forma mínima. El desarrollo de servicios RESTful es barato y tiene muy pocas "barreras" para su adopción.

CONCLUSIONES

- Es importante conocer la composición de los servicios web, para la ejecución de una aplicación.
- Un servicio web, puede tener varios lenguajes que interactúan entre si, ya que estas pueden ser ejecutadas en diferentes plataformas.
- Existen diversas formas en las que se puede construir un servicio web.
- Combinan lo mejor de la tecnología de componentes y de la tecnología Web
- Utilizan de estándares de internet
- Se basan en tecnología de traslado de mensajes.

Upgrade to Pro

Learn

Community

Catalog



Learn SQL

Great job completing L

COMPLETED



201602469

310 pts

Ver mi perfil

Configuración de la cuenta

Community Forums

Ayuda

Log out

INTENSIVE

Learn an **Introduction to Data Analysis** by taking this accelerated 10-week program.

LEARN FASTER