



Arquitecturas Distribuidas

TEMA 5. Otras arquitecturas distribuidas
IV. Web Services

IV. Web Services

1. ¿Qué son los Web Services?
2. Ejemplos de Web Services
3. Tecnologías y arquitectura
 - 3.1. Arquitectura
 - 3.2. Lenguaje de descripción: WSDL
 - 3.3. Protocolo: SOAP
 - 3.4. Registro de servicios: UDDI
4. Service-Oriented Architecture (SOA)
5. Bibliografía

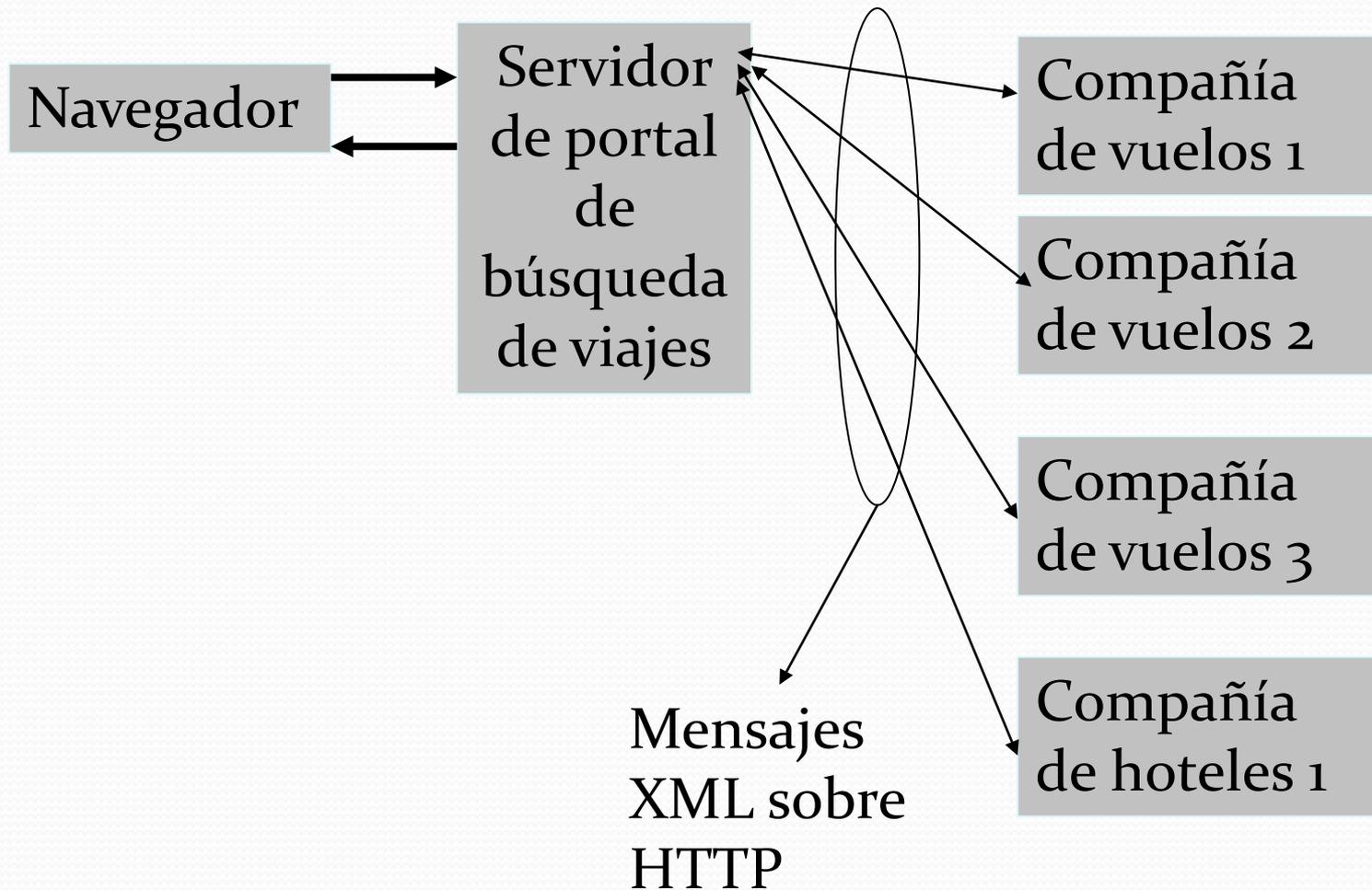
¿Qué son los Web Services?

- Es una tecnología de sistemas distribuidos que usa protocolos de estándar de Internet para intercambiar documentos XML entre procesos.
- Desde el punto de vista del programador funcionan como una llamada procedimiento remoto: una aplicación en el servidor local ejecuta una función en el servidor remoto y obtiene un resultado.
- Permite la interacción máquina-máquina a través de Internet
 - En términos simples permiten el intercambio de documentos XML sobre HTTP
 - Interfaz descrito en un formato procesable por una máquina (WSDL)
- Permite la exposición de “servicios” **a través de la web** (Internet)
- Permite la interacción entre aplicaciones (software) **a través de la web**
- Facilita la interoperabilidad entre sistemas heterogéneos

Ejemplos de Web Services

- Búsqueda y reserva de vuelos
 - El usuario entra a un portal de búsqueda de vuelos e introduce los parámetros
 - El portal debe interrogar a las compañías de vuelo...¿cómo? ¿utilizan los mismos sistemas todas? ¿hay alguna interfaz común? ¿alguna infraestructura común?
 - SI: Internet.
 - La aplicación en el servidor interroga a las compañías usando el protocolo común: HTTP
- Otros ejemplos: Amazon, Google, etc.

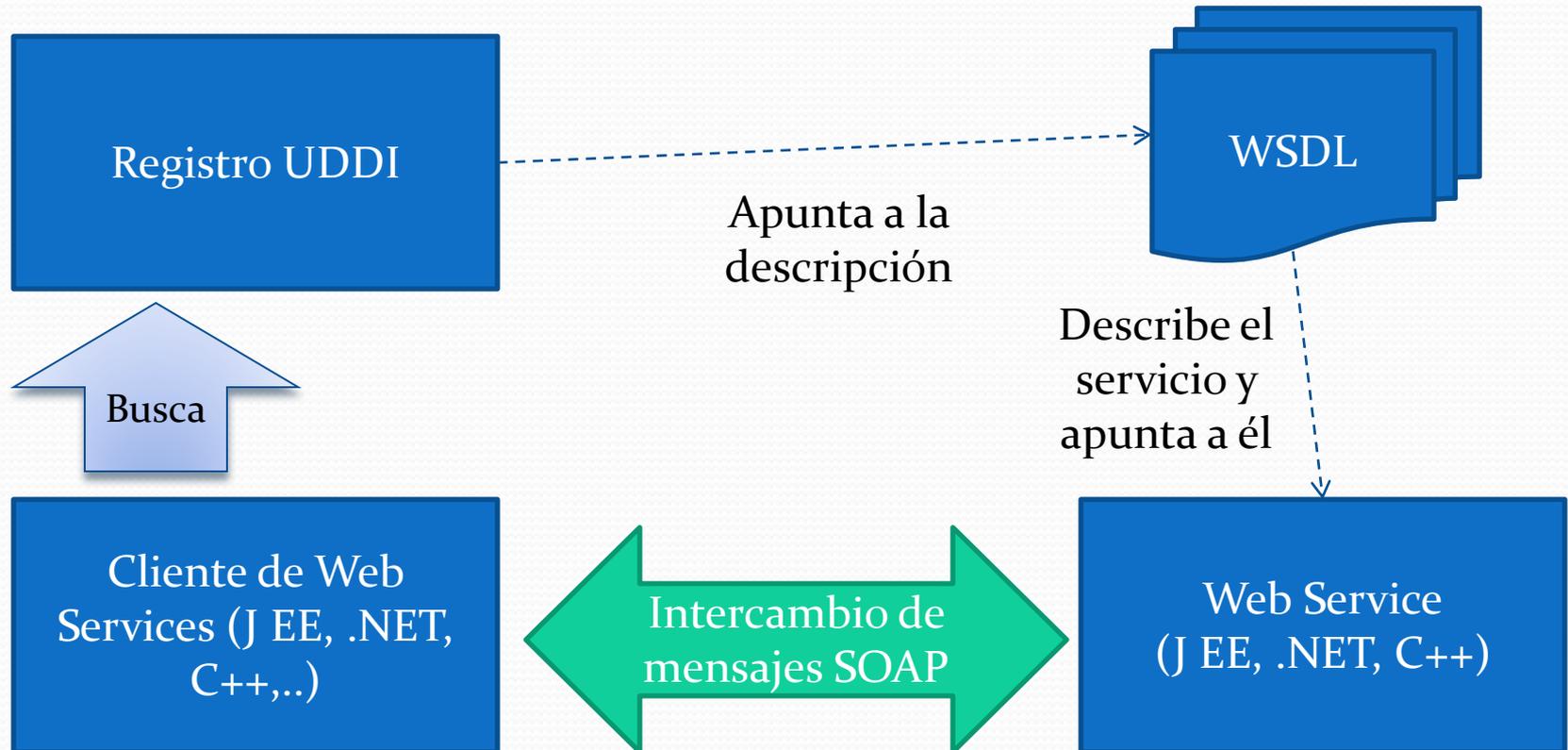
Ejemplos de Web Services



Tecnologías

- Internet está basada en texto arbitrario esencialmente
 - No es adecuado para interacción entre máquinas/aplicaciones, es necesario estructurarlo
 - Necesitamos mecanismos que permitan la ejecución de instrucciones que se habrían introducido mediante formularios
- Web Services utiliza tres tecnologías:
 - **SOAP**: protocolo de intercambio de mensajes
 - **WSDL**: lenguaje de definición de interfaces
 - **UDDI**: repositorio de servicios (páginas amarillas)

Arquitectura de Web Services



Tecnologías (II)

- Web Services funcionan como las llamadas a procedimiento remoto, pero sobre la infraestructura básica de Internet
 - Los métodos o funciones que se pueden invocar junto con los parámetros de entrada y de salida (interfaz del WS) se especifican mediante WSDL.
 - UDDI es un directorio de WS con sus correspondientes descripciones en WSDL
- Una vez disponemos de la descripción WSDL se invoca la función remota (WS) mediante intercambio de mensajes SOAP
 - SOAP es un lenguaje de marcas XML
 - Se pueden utilizar distintos protocolos pero lo habitual es usar SOAP sobre HTTP
- El servidor de WS implementa el servicio. Procesa la llamada y devuelve un resultado con XML encapsulado en SOAP.
- El proceso es independiente del lenguaje de programación y la plataforma.

Lenguaje de descripción: WSDL

- WS exponen una función (de negocio) con la que otras aplicaciones pueden interactuar a través de la web.
- Es necesario declarar una **interfaz del servicio**
 - Descripción de las funciones
 - Descripción de la forma de interacción con las funciones: qué parámetros recibe, que resultados devuelve, etc.
 - Puesto que la interacción se produce a través de una red es necesario especificar el protocolo a usar, la dirección destino, el puerto, etc.

Lenguaje de descripción: WSDL

- Web Services Description Language (WSDL)
 - Describir (declarar) y publicar el formato y protocolos de un WS de manera estándar.
 - Descripción (XML) de los datos que acepta el servicio
 - Descripción de las operaciones que se realizan sobre los datos
 - Indicación del protocolo o transporte que se utilizará

```
http://www.xmethods.net/sd_ibm/TemperatureService.wsdl - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://www.xmethods.net/sd_ibm/TemperatureService.wsdl
Google Search Web Search Groups PageRank Page Info Up Highlight Norton AntiVirus

<?xml version="1.0" ?>
- <definitions name="TemperatureService"
  targetNamespace="http://www.xmethods.net/sd/TemperatureService.wsdl"
  xmlns:tns="http://www.xmethods.net/sd/TemperatureService.wsdl"
  xmlns:xsd="http://www.w3.org/1999/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns="http://schemas.xmlsoap.org/wsdl/">
- <message name="getTempRequest">
  <part name="zipcode" type="xsd:string" />
</message>
- <message name="getTempResponse">
  <part name="return" type="xsd:float" />
</message>
- <portType name="TemperaturePortType">
  - <operation name="getTemp">
    <input message="getTempRequest" name="getTemp" />
    <output message="getTempResponse" name="getTempResponse" />
  </operation>
</portType>
- <binding name="TemperatureBinding" type="TemperaturePortType">
  <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http" />
  - <operation name="getTemp">
    <soap:operation soapAction="" />
    - <input>
      <soap:body use="encoded" namespace="urn:xmethods-Temperature"
        encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
    </input>
    - <output>
      <soap:body use="encoded" namespace="urn:xmethods-Temperature"
        encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
    </output>
  </operation>
</binding>
- <service name="TemperatureService">
  <documentation>Returns current temperature in a given U.S. zipcode</documentation>
  - <port name="TemperaturePort" binding="TemperatureBinding">
    <soap:address location="http://services.xmethods.net:80/soap/servlet/rpcrouter" />
  </port>
</service>
</definitions>
```

Parámetros y tipos

Transporte/protocolo

Descripción y localización

Protocolo: SOAP

- Simple Object Access Protocol (SOAP)
- Transporta documentos XML sobre la web
- Se utiliza habitualmente sobre HTTP aunque otros protocolos son posibles (o extensiones como HTTPS)
 - Utiliza el comando POST de HTTP
- Mensajes SOAP son documentos XML
- 3 elementos
 - SOAP Envelope: elemento raíz
 - SOAP Header: cabeceras (opcionales)
 - SOAP Body: cuerpo del mensaje

Protocolo: SOAP

```
<?xml version="1.0"?>
  <soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
    soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
    <soap:Header> <m:Trans xmlns:m="http://www.w3schools.com/transaction/"
      soap:mustUnderstand="1">234</m:Trans>
    </soap:Header>
    <soap:Body>
      <m:GetPrice xmlns:m="http://www.w3schools.com/prices">
        <m:Item>Apples</m:Item>
      </m:GetPrice>
    </soap:Body>
  </soap:Envelope>
```

Protocolo: SOAP

```
POST /InStock HTTP/1.1
Host: www.example.org
Content-Type: application/soap+xml; charset=utf-8
Content-Length: nnn
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
  soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
<soap:Body xmlns:m="http://www.example.org/stock">
  <m:GetStockPrice>
    <m:StockName>IBM</m:StockName>
  </m:GetStockPrice>
</soap:Body>
</soap:Envelope>
```

Protocolo: SOAP

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: nnn
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
  soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
<soap:Body xmlns:m="http://www.example.org/stock">
  <m:GetStockPriceResponse>
    <m:Price>34.5</m:Price>
  </m:GetStockPriceResponse>
</soap:Body>
</soap:Envelope>
```

Registro de servicios: UDDI

- Universal Description, Discovery and Integration (UDDI) Registry
 - Como unas páginas amarillas de servicios
 - Permiten el **descubrimiento** (cliente) y **publicación** (proveedor) de servicios
 - Devuelven la URL dónde se encuentra el documento WSDL que describe un servicio que coincide con la búsqueda
 - Una vez el cliente dispone de dicha URL, solicita directamente el WSDL

Service-Oriented Architecture (SOA)

- Arquitectura Basada en Servicios
 - Diseño y desarrollo de aplicaciones complejas mediante el uso y composición de distintos servicios *interoperables*
 - Los servicios se van componiendo, es decir, invocan y utilizan otros servicios publicados. Similar al uso de objetos en POO.
 - Los servicios son independientes del lenguaje y plataforma.
- Mediante WS se puede implementar una arquitectura tipo SOA

Referencias y bibliografía

- Libros:
 - “Understanding Web Services”, *Eric Newcomer*, Addison Wesley, 2002.

Referencias y bibliografía

- WWW:
- <http://www.w3.org/2002/ws/> →
Especificaciones WS
- <http://www.w3schools.com/webservices/default.asp> →
Tutoriales y ejemplos