

# SERVICIOS WEB



Tópicos Selectos de Computación I  
Marzo 2014

# AGENDA

- **Evolución de aplicaciones distribuidas**
- Problemática con las aplicaciones distribuidas tradicionales
- Aplicaciones Web
- Introducción a los servicios Web
- SOA
- Escenarios Comunes de Servicios Web

# APLICACIONES DISTRIBUIDAS

- Distribuir: Acción y efecto de distribuir¿?.
  - Darle a cada cual una porción de algo en función de algún criterio.
  - Otorgarle a cada parte de un todo una función determinada.
  - Colocar cada cosa en un lugar siguiendo algún criterio.

# DEFINICIÓN EN SISTEMAS

- Conjunto de computadoras independientes que se muestran al usuario como un sistema único coherente [Tanenbaum].
- Sistema en el cual componentes de hardware y software, localizadas en computadoras en red, se comunican y coordina sus acciones sólo por paso de mensajes [Coulouris].

# APLICACIONES DISTRIBUIDAS

- Ejemplo de distribución
  - La gente
- Ejemplo de centralizado
  - Una persona

\*Al ser la vida del ser humano centralizada repercuten en la construcción de sistemas distribuidos.

# CARACTERÍSTICAS

- **Transparencia.**- Ocultar la distribución.
- **Eficiencia.**- Al distribuir una tarea en un sistema distribuido se hacen más rápida.
- **Flexibilidad.**- Abierto a cambios y mejoras.
- **Escalabilidad.**- Aumentar o disminuir los elementos del sistema.
- **Fiabilidad.**- Un sistema distribuido no debe de estar ligado una computadora en particular. Una de las formas para lograr esto es con redundancia de nodos, computadoras.

# MODELOS EN SISTEMAS DISTRIBUIDOS

- Diseño de sistemas distribuidos
  - Observación experimental
    - Se construyen sistemas y se observa como se comportan en diferentes situaciones.
  - Modelado y análisis
    - Empleando una versión simplificada del objeto de estudio donde se definen las restricciones para el comportamiento del sistema.

# MODELOS EN SISTEMAS DISTRIBUIDOS

10/03/2014

- Un modelo es una colección de atributos y reglas que definen como éstos interactúan.
- Un modelo es analizable si al analizarlo se obtienen hechos que corresponden al objeto real.
- Un modelo es manejable si es posible analizarlo.



# PARADIGMAS

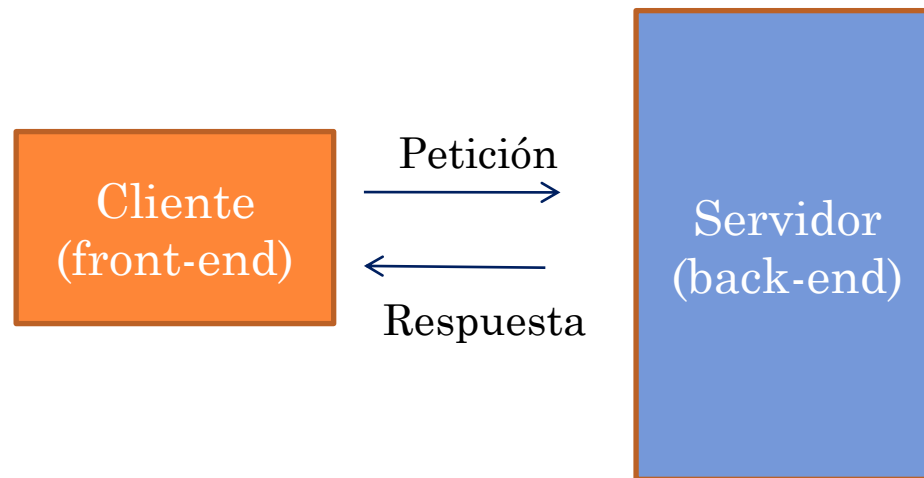
- Espacios de objetos
- Servicios en red (CORBA)
- Llamadas a procedimientos remotos (ONC RPC, DCOM)
- Modelo cliente-servidor
- Paso de mensajes

# PASO DE MENSAJES

- De la comunicación más simple a la comunicación entre procesos.
- Los datos que representan mensajes se intercambian entre 2 procesos, un emisor y receptor.
- Este paradigma es fundamental para aplicaciones distribuidas.

# CLIENTE-SERVIDOR

- Un modelo cliente-servidor puede definirse como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información en forma transparente.



# P2P

- Es una arquitectura donde todos los participantes son iguales.
- Puede existir un servidor sin embargo su participación es mínima.
- Ejemplos:
  - Telefonía sobre Internet
  - Mensajería instantánea.

Ofrece mayor escalabilidad y robustez.

# REDES P2P

- Redes P2P totalmente descentralizadas:
  - Son más versátiles.
  - No hay servidores.
  
- Redes P2P Híbridas:
  - Tiene la posibilidad de usar servidores para realizar algunas operaciones. Por ejemplo: Autenticación.

# AGENDA

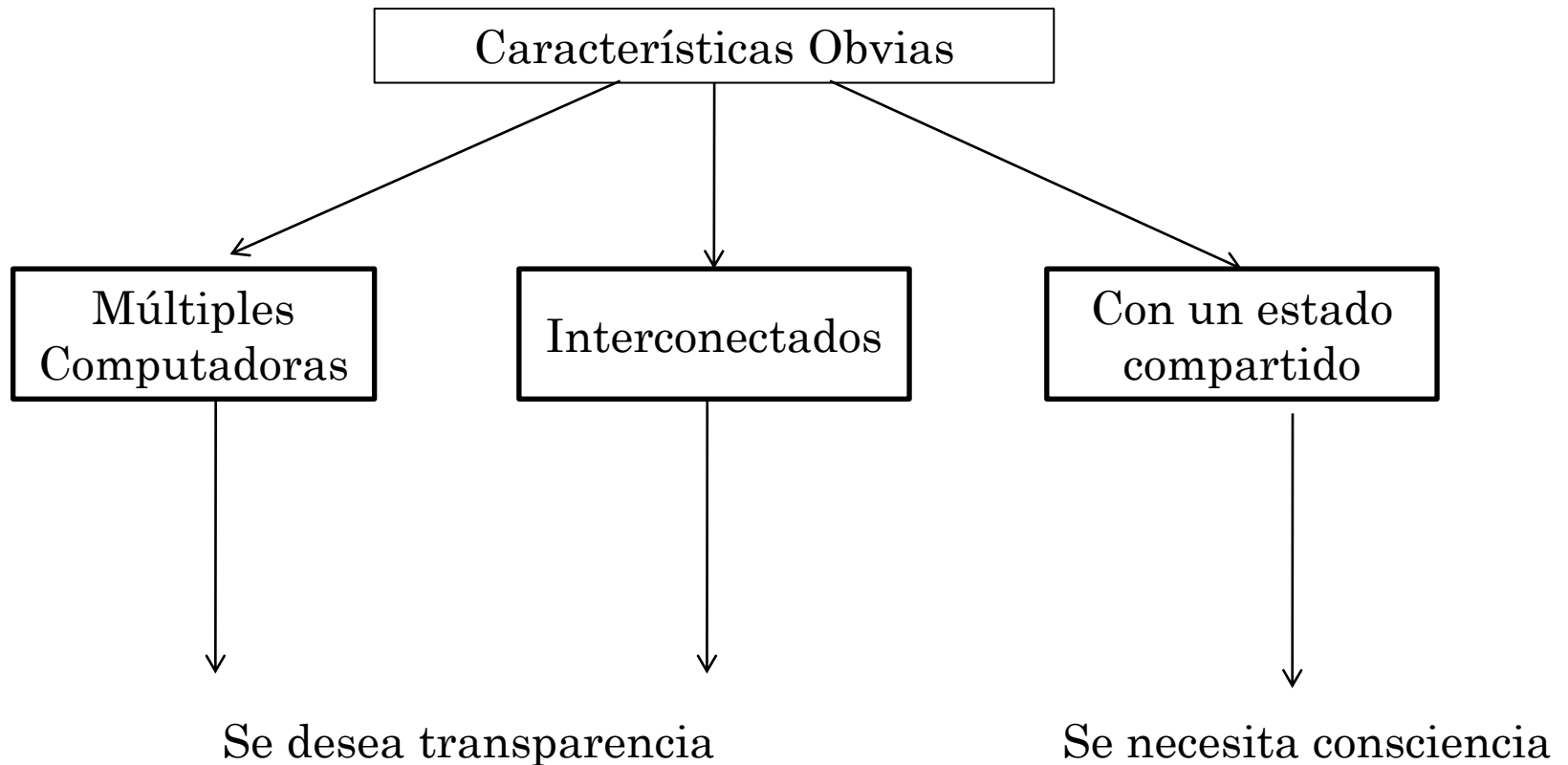
- Evolución de aplicaciones distribuidas
- **Problemática con las aplicaciones distribuidas tradicionales**
- Aplicaciones Web
- Introducción a los servicios Web
- SOA
- Escenarios Comunes de Servicios Web

# RETOS

## ○ En:

- En el diseño del sistema.
- Interoperabilidad, heterogeneidad.
- Comunicación.
- Manejo de estados compartidos.
- Manejo de sistema grandes(crecimiento , complejidad).
- Ofrecer robustez.

# CARACTERÍSTICAS





# AGENDA

- Evolución de aplicaciones distribuidas
- Problemática con las aplicaciones distribuidas tradicionales
- **Aplicaciones Web**
- Introducción a los servicios Web
- SOA
- Escenarios Comunes de Servicios Web

# APLICACIONES WEB

- Son aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador (browser).
- Es decir, es una aplicación de software que se codifica en un lenguaje soportado por los navegadores web (HTML, etc).

# APLICACIONES WEB

- Las aplicaciones web son populares debido a:
  - Lo práctico del navegador web como cliente ligero.
  - La independencia del sistema operativo.
  - La facilidad para actualizar y mantener la aplicación, pues no es necesario distribuir e instalar a los usuarios potenciales o finales.

# HTML - HYPERTEXT MARKUP LANGUAGE

10/03/2014

- Lenguaje de Marcado de Hipertexto.
  - Es el lenguaje de marcado predominante para la construcción de páginas web.
  - Es usado para describir la estructura y el contenido en forma de texto.
  - También es usado para integrar el texto con objetos tales como imágenes, etc.

# APLICACIONES WEB

- Las aplicaciones web generan dinámicamente páginas en un formato estándar, como:
  - HTML, XHTML.
- Se utilizan lenguajes interpretados en el lado del cliente para añadir elementos dinámicos a la interfaz de usuario. Lenguajes como:
  - JavaScript, Java (Applets), Flash, etc.

# APLICACIONES WEB

- Las aplicaciones web generan dinámicamente páginas en un formato estándar, como:
  - HTML, XHTML.
- Se utilizan lenguajes interpretados en el lado del cliente para añadir elementos dinámicos a la interfaz de usuario. Lenguajes como:
  - JavaScript, Java (Applets), Flash, etc.

# HTTP

- Es el protocolo usado en cada transacción de la World Wide Web.
- Es un protocolo de aplicación. Para su transporte se usa TCP.
- Ha sido desarrollado por el World Wide Web Consortium y la Internet Engineering Task Force.
- Se culminó en 1999 con la publicación de una serie de RFC, el más importante de ellos es el RFC 2616 que especifica la versión 1.1.

# HTTP

- Es orientado a transacciones y sigue el esquema solicitud-respuesta entre un cliente y un servidor.
- El cliente que efectúa la petición es el “User-Agent” (Navegador).



# HTTP - PETICIÓN

GET /index.html HTTP/1.1

Host: [www.google.com](http://www.google.com)

User-Agent: Chrome

# HTTP - RESPUESTA



A screenshot of a Telnet window titled "Telnet 127.0.0.1". The window displays an HTTP response from a local server. The response headers include status "200 OK", content type "text/html", last modified date, ETag, server version "Microsoft-IIS/7.5", and X-Powered-By "ASP.NET". The response body is an HTML document with a DOCTYPE declaration, XML namespace, head section with meta, title, and style tags, and a body section with CSS rules for color, background-color, and margin.

```
C:\> Telnet 127.0.0.1

HTTP/1.1 200 OK
Content-Type: text/html
Last-Modified: Sun, 06 Feb 2011 21:56:37 GMT
Accept-Ranges: bytes
ETag: "8a501eba48c6cb1:0"
Server: Microsoft-IIS/7.5
X-Powered-By: ASP.NET
Date: Wed, 17 Aug 2011 05:40:18 GMT
Content-Length: 535

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>IIS7</title>
<style type="text/css">
<!--
body {
    color:#000000;

    background-color:#B3B3B3;
    margin:0;
}
```

# HTTP – MÉTODOS DE PETICIÓN

- GET.- Pide un recurso especificado.
- POST.- Envía datos para que sean procesados para el recurso identificado.
- HEAD.- Pide una respuesta a la que correspondería una petición GET, pero sin el cuerpo de respuesta.
- PUT.- Para subir o cargar un recurso especificado.
  - PUT /archivo.html HTTP/1.1
- DELETE.- Borra un recurso especificado.
- TRACE.- Solicita al servidor que envíe de vuelta en un mensaje de respuesta con la que recibió del mensaje de solicitud.

# HTTP – MÉTODOS DE PETICIÓN

- **OPTIONS.-** Representa una petición de información sobre las opciones de comunicación disponibles.
  - OPTIONS / HTTP/1.0
  - Host: 127.0.0.1
- **CONNECT.-** Esta especificación se reserva el nombre del método CONNECT para usarlo con un proxy que puede cambiar dinámicamente a ser un túnel.

# HTTP ENCABEZADOS PARA SOLICITUD

Nombre del encabezado	Descripción
Accept	Tipo de contenido aceptado por el navegador (por ejemplo, <i>texto/html</i> ). Consulte <a href="#">Tipos de MIME</a>
Accept-Charset	Juego de caracteres que el navegador espera
Accept-Encoding	Codificación de datos que el navegador acepta
Accept-Language	Idioma que el navegador espera (de forma predeterminada, inglés)
Authorization	Identificación del navegador en el servidor
Content-Encoding	Tipo de codificación para el cuerpo de la solicitud
Content-Language	Tipo de idioma en el cuerpo de la solicitud
Content-Length	Extensión del cuerpo de la solicitud
Content-Type	Tipo de contenido del cuerpo de la solicitud (por ejemplo, <i>texto/html</i> ). Consulte <a href="#">Tipos de MIME</a>
Date	Fecha en que comienza la transferencia de datos
Forwarded	Utilizado por equipos intermediarios entre el navegador y el servidor
From	Permite especificar la dirección de correo electrónico del cliente
From	Permite especificar que debe enviarse el documento si ha sido modificado desde una fecha en particular
Link	Vínculo entre dos direcciones URL
Orig-URL	Dirección URL donde se originó la solicitud
Referer	Dirección URL desde la cual se realizó la solicitud
User-Agent	Cadena con información sobre el cliente, por ejemplo, el nombre y la versión del navegador y el sistema operativo

# HTTP

## ENCABEZADOS

### DE RESPUESTA

Nombre del encabezado	Descripción
Content-Encoding	Tipo de codificación para el cuerpo de la respuesta
Content-Language	Tipo de idioma en el cuerpo de la respuesta
Content-Length	Extensión del cuerpo de la respuesta
Content-Type	Tipo de contenido del cuerpo de la respuesta (por ejemplo, <i>texto/html</i> ). Consulte <a href="#">Tipos de MIME</a>
Date	Fecha en que comienza la transferencia de datos
Expires	Fecha límite de uso de los datos
Forwarded	Utilizado por equipos intermediarios entre el navegador y el servidor
Location	Redireccionamiento a una nueva dirección URL asociada con el documento
Server	Características del servidor que envió la respuesta

# ALGUNOS ASPECTOS DE SEGURIDAD

## ○ XSS

- Cross-site scripting es un tipo de inseguridad informática o agujero de seguridad basado en la explotación de vulnerabilidades del sistema de validación de HTML incrustado. Wikipedia.

## ○ SQL Injection (Inyección SQL)

- Inyección SQL es un método de infiltración de código intruso que se vale de una vulnerabilidad informática presente en una aplicación en el nivel de validación de las entradas para realizar consultas a una base de datos. Wikipedia.

# ALGUNOS ASPECTOS DE SEGURIDAD

10/03/2014

## ○ Clonación de cookies

- Una cookie es un fragmento de información que se almacena en el disco duro del visitante de una página web a través de su modo a petición del servidor de la página.
- Esta información puede ser luego recuperada por el servidor en posteriores visitas. En ocasiones también se le llama "huella".
- Inconveniente: identificación inexacta, robo de cookies, falsificación de cookies.



# ALGUNOS ASPECTOS DE SEGURIDAD

- Configuración errónea en los servidores.
- No cifrar la información crítica.
- Mal uso de memoria y recursos compartidos.
- Permitir Local File Inclusion (LFI), Remote File Inclusion (RFI).

# ALGUNOS ASPECTOS DE SEGURIDAD

- Alternativas de la cookies:
  - Por dirección IP.
  - Por URLs.
  - Persistencia en el navegador.
  - Autenticación HTTP.
  - Persistencia windows.name de javascript.
  - Objetos Macromedia Flash.

# AGENDA

- Evolución de aplicaciones distribuidas
- Problemática con las aplicaciones distribuidas tradicionales
- Aplicaciones Web
- **Servicios Web**
- SOA
- Escenarios Comunes de Servicios Web

# SERVICIOS WEB

- Un servicio web es un componente al que se podrá acceder mediante protocolos Web estándar.
- Los servicios Web cuentan con una interfaz que son invocados de manera remota.
- Son independientes de plataforma y del lenguaje de programación en el que estén implementados.

# SERVICIOS WEB - DEFINICIÓN 1

## ○ Servicios Web

- Un conjunto de aplicaciones o de tecnologías con capacidad para interoperar en la Web.

## ○ Tecnología

- Conjunto de teorías y de técnicas que permiten el aprovechamiento práctico del conocimiento científico. Diccionario de la Real Academia.

# SERVICIOS WEB - DEFINICIÓN 1

- Estas aplicaciones o tecnologías intercambian datos entre sí con el objetivo de ofrecer servicios.
- Los proveedores ofrecen sus servicios como procedimientos remotos y los usuarios solicitan un servicio llamando a procedimientos a través de la Web.

## SERVICIOS WEB - DEFINICIÓN 2

- Una aplicación de software identificada por una URI, cuyas interfaces y vinculaciones son capaces de ser definidas, descritas y descubiertas como artefactos XML. El SW soporta la interacción con otros agentes de software mediante el intercambio de mensajes en XML a través de protocolos en Internet. W3C.
- W3C (World Wide Web Consortium) es una comunidad internacional que desarrolla estándares que aseguran el crecimiento de la Web a largo plazo.
  - HTML5, XHTML5, XML 1.1 y CSS3

# DEFINICIÓN DE URI

## URI (UNIFORM RESOURCE IDENTIFIER)

- Identificador uniforme de recurso
  - Es una cadena de caracteres que identifica inequívocamente un recurso (servicio, página, documento, dirección de correo electrónico, enciclopedia, etc.).
  - Estos recursos pueden ser accesibles en una red o sistema. Los URI pueden ser localizadores uniformes de recursos, Uniform Resource Name, o ambos.
  - Por ejemplo:
    - <http://topicosselectos.org>



# ANTECESORES

10/03/2014

- CORBA. Flexible debido a que permite la comunicación entre aplicaciones programadas en lenguajes distintos y ejecutadas en diferentes plataformas.
- RMI. Los componentes deben ser programados en Java.
- DCOM. Solución de integración propietaria de Microsoft y sólo adaptable a sus lenguajes y sistemas operativos.

# SERVICIOS WEB

- Antecesoros de los servicios Web:
  - Existen varios problemas que son aplicables a las soluciones anteriores:
    - Complejidad en implementación
    - Incapacidad de usar el protocolo http.
    - Problemas con los firewalls.

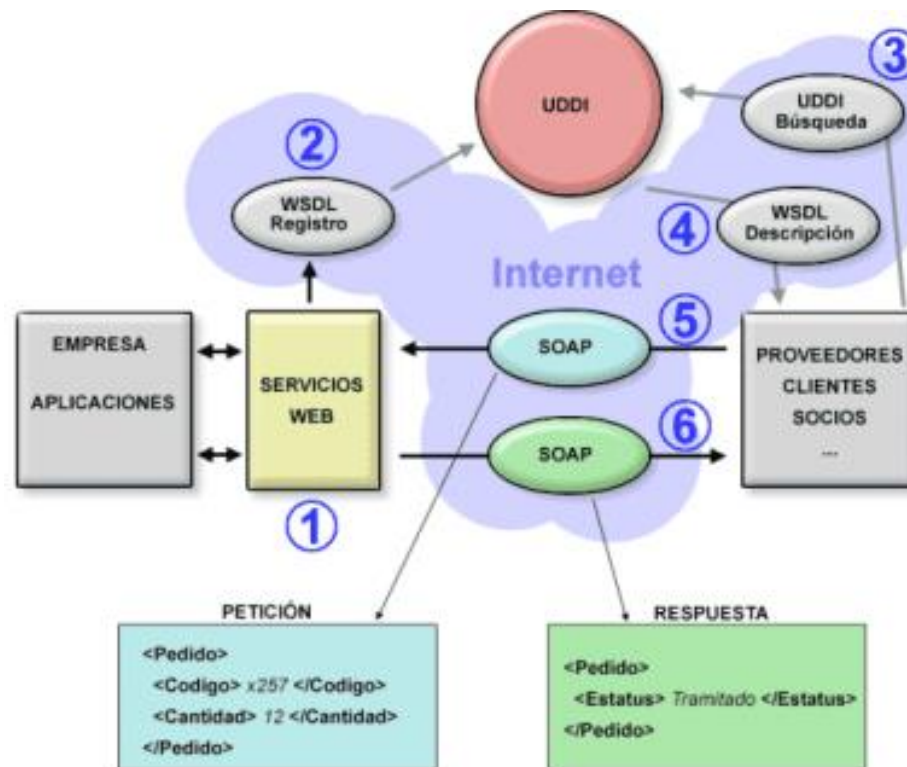
## OBJETIVOS (1)

- Independencia del lenguaje y de la plataforma.- Separación de la especificación de la implementación.
- Interoperabilidad.- Utilización de estándares UDDI, WSDL, XML, SOAP.
- Acoplamiento débil.- Sistemas basados en mensajes (interacciones síncronas y asíncronas).

## OBJETIVOS (2)

- A través de internet:
  - Sin control centralizado.
  - Utilización de Protocolos establecidos
  - Consideraciones de seguridad
- Modularidad y reusabilidad de servicios
- Escalabilidad.- Uno a uno frente a uno a muchos.

# CICLO DE VIDA



# ESTÁNDARES - UDDI

- UDDI

Son las siglas del catálogo de negocios de Internet denominado “Universal Description, Discovery and Integration”. El registro en el catálogo se hace en XML.

- Es uno de los estándares básicos de los servicios Web cuyo objetivo es ser accedido por los mensajes SOAP y dar paso a documentos WSDL, en los que se describen los requisitos del protocolo y los formatos del mensaje solicitado para interactuar con los servicios Web del catálogo de registros.

# ESTÁNDARES - UDDI

- El registro de un negocio en UDDI tiene tres partes:
  1. Páginas blancas - dirección, contacto y otros identificadores conocidos.
  2. Páginas amarillas - categorización industrial basada en taxonomías.
  3. Páginas verdes - información técnica sobre los servicios que aportan las propias empresas.

# FUNCIONAMIENTO DE UDDI

- Provider: Información sobre una entidad que ofrece el servicio.
- Service: Información sobre una familia particular de ofertas.
- Binding: Información técnica sobre un punto de entrada a un servicio.



# ESTÁNDARES - XML

- *XML es una tecnología clave para asegurar la interoperabilidad.*
- Primera recomendación (XML 1.0) publicada en Febrero de 1998.
- Ideado para: representación e intercambio de datos.
- Es el primer paso para la interoperabilidad.
  - Diferentes sistemas compartiendo información.
- Base de muchas tecnologías.

# ESTÁNDARES - XML

10/03/2014

```
<?xml version="1.0"?>
```

```
<nota>
```

```
  <para>Tove</para>
```

```
  <de>Jani</de>
```

```
  <asunto>Reminder</asunto>
```

```
  <contenido>Don't forget me this weekend!</contenido>
```

```
</nota>
```

# XML ESQUEMAS

## ○ XML Schema

- Es un lenguaje de esquema utilizado para describir la estructura y las restricciones de los contenidos de los documentos XML, va más allá de las normas sintácticas impuestas por el propio lenguaje XML.
- Se consigue una percepción del tipo de documento con un nivel alto de abstracción.
- Desarrollado por el World Wide Web Consortium (W3C).

# ESTÁNDARES - WSDL

- Descripción de un servicio:
  - El intercambio de mensajes entre las entidades está guiado por un protocolo que éstas deben seguir para que las transacciones lleguen a buen fin.
  - Este protocolo está documentado en la descripción del SW (WSD, Web Service Description).
  - La WS es una descripción completa del SW, escrito en un lenguaje denominado WSDL y que es legible tanto por las computadoras como por las personas, aunque no sin cierta dificultad para estos últimos.

# ESTÁNDARES - WSDL

- WSDL (Web Service Description Language)
  - Formato XML.
  - Permite describir los servicios Web.
    - ¿Que puede hacer el servicio?
    - ¿Dónde reside?
    - ¿Cómo invocarlo?
  - Evolución
    - WSDL 1.1 (2001), 1.2 (2004), 2.0 actualmente)
    - <http://www.w3.org/TR/wsdl20/>

# ESTRUCTURA DE WSDL

- La estructura de un servicio contiene:
  - types: Tipos de datos usado en los mensajes (XML-Schema)
  - message: Definición abstracta de los datos transmitidos.
  - portType: conjunto de operaciones abstractas.
  - port: Especifica una dirección para el enlace definiendo un único punto de destino.
  - service: Colección de puntos destino.

# ESTRUCTURA DE WSDL

Servicio

Tipos

10/03/2014

```
▼<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/" xmlns:tns="http://topicosselectos.org/"
xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://topicosselectos.org/">
▼<wsdl:types>
  ▼<s:schema elementFormDefault="qualified" targetNamespace="http://topicosselectos.org/">
    ▼<s:element name="Sumar">
      ▼<s:complexType>
        ▼<s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="x" type="s:int"/>
          <s:element minOccurs="1" maxOccurs="1" name="y" type="s:int"/>
        </s:sequence>
      </s:complexType>
    </s:element>
    ▼<s:element name="SumarResponse">
      ▼<s:complexType>
        ▼<s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="SumarResult" type="s:int"/>
        </s:sequence>
      </s:complexType>
    </s:element>
  </s:schema>
</wsdl:types>
▼<wsdl:message name="SumarSoapIn">
  <wsdl:part name="parameters" element="tns:Sumar"/>
</wsdl:message>
▼<wsdl:message name="SumarSoapOut">
  <wsdl:part name="parameters" element="tns:SumarResponse"/>
</wsdl:message>
▼<wsdl:portType name="CalculadoraSoap">
  ▼<wsdl:operation name="Sumar">
    <wsdl:input message="tns:SumarSoapIn"/>
    <wsdl:output message="tns:SumarSoapOut"/>
  </wsdl:operation>
</wsdl:portType>
▼<wsdl:binding name="CalculadoraSoap" type="tns:CalculadoraSoap">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
  ▼<wsdl:operation name="Sumar">
```



# ESTÁNDARES – SOAP

## (SIMPLE OBJECT ACCESS PROTOCOL)

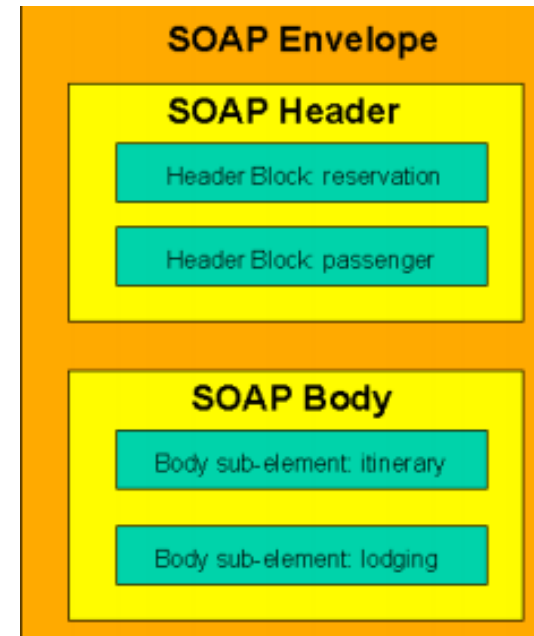
- Es un protocolo estándar que define el formato de los mensajes para su envío.
- Independiente del protocolo de transporte.
- Evolución:
  - Desarrollado a partir de XML-RPC.
  - SOAP 1.0 (1999), 1.1 (2000), 1.2 (2003).

\*XML-RPC es un protocolo de llamada a procedimientos remotos que usa XML para codificar los datos y HTTP como protocolo de transmisión de mensajes.



# SOAP ENVELOPE (ENVOLTURA)

- **Envelope** contiene un **header** opcional y el **body** obligatorio.
- **Body** contiene datos en XML.
- **Header** contiene meta-información.



# EJEMPLO SOAP

## SOAP 1.1

A continuación se muestra un ejemplo de solicitud y respuesta para SOAP 1.1. Es necesario reemplazar los **marcadores de posición** que aparecen con valores reales.

```
POST /Calculadora.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://topicosselectos.org/Sumar"
```

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Sumar xmlns="http://topicosselectos.org/">
      <x>int</x>
      <y>int</y>
    </Sumar>
  </soap:Body>
</soap:Envelope>
```

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length
```

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <SumarResponse xmlns="http://topicosselectos.org/">
      <SumarResult>int</SumarResult>
    </SumarResponse>
  </soap:Body>
</soap:Envelope>
```

# EJEMPLO SOAP

POST /Calculadora.asmx HTTP/1.1

Host: localhost:35918

Content-Type: application/soap+xml; charset=utf-8

Content-Length: 363

**<?xml version="1.0" encoding="utf-8"?>**

**<soap12:Envelope**

**xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"**

**xmlns:xsd="http://www.w3.org/2001/XMLSchema"**

**xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">**

**<soap12:Body>**

**<Sumar xmlns="http://topicosselectos.org/">**

**<x>2</x>**

**<y>3</y>**

**</Sumar>**

**</soap12:Body>**

**</soap12:Envelope>**

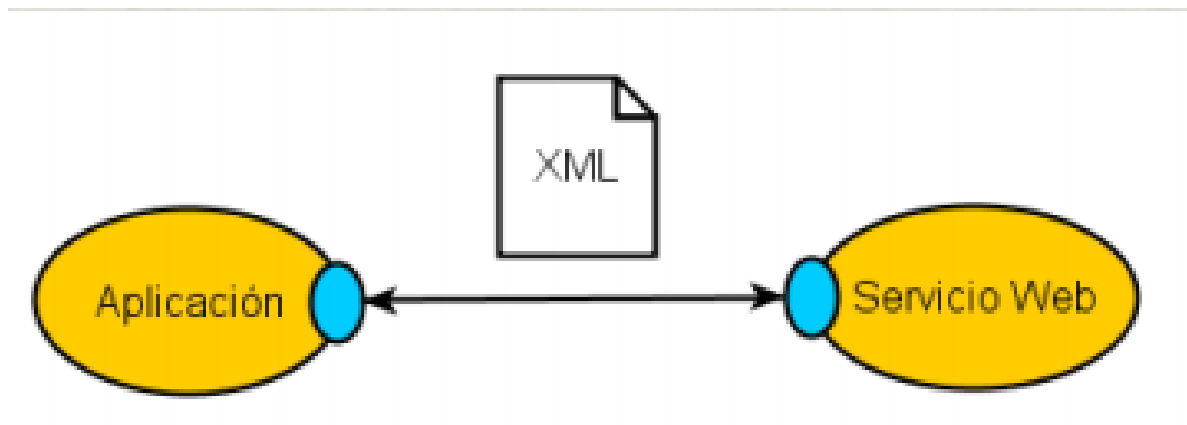
# SERVICIOS WEB Y XML

- Sistemas en la Web
  - Servicios Web: permiten interoperación entre máquinas sobre una red :
    - Diferentes aplicaciones / diferentes plataformas.
    - Tiene una interface bien definida.
    - Tiene un comportamiento previamente definido.
    - Típicamente, los envíos se realizan usando HTTP con XML.
  - Estándares para las interfaces de los programas e interacciones entre estos.
  - Facilitan la prestación y el uso de servicios (y de los datos que proveen).

# SERVICIOS WEB Y XML

10/03/2014

- Los servicios Web se basan en XML para estructurar la información :
  - Homogeneidad para facilitar la comprensión de las máquinas.
  - Diferentes plataformas / marcos de trabajo.



# PRINCIPALES ACCIONES

- **1. Representación de mensajes:**
  - Los mensajes suelen ser documentos XML.
  - SOAP es un vocabulario XML que permite incluir mensajes XML.
  - Además, en SOAP se representa otra información:
    - Cabecera (meta-información) y cuerpo del mensaje.

# PRINCIPALES ACCIONES

- **2. Transporte de mensajes.-** Habitualmente HTTP, aunque pueden utilizarse otros protocolos.
- **3. Descripción del servicio.-** Representación del tipo de operaciones y su funcionalidad (interfaz).
  - WSDL es el vocabulario más utilizado.
  - Define las operaciones y el tipo que tienen (no define funcionalidad).
- **4. Registro.-** Es necesario contar con un sistema que permita el registro (UDDI).

# ARQUITECTURA DE PROTOCOLOS

**Localización de servicios  
(UDDI)**

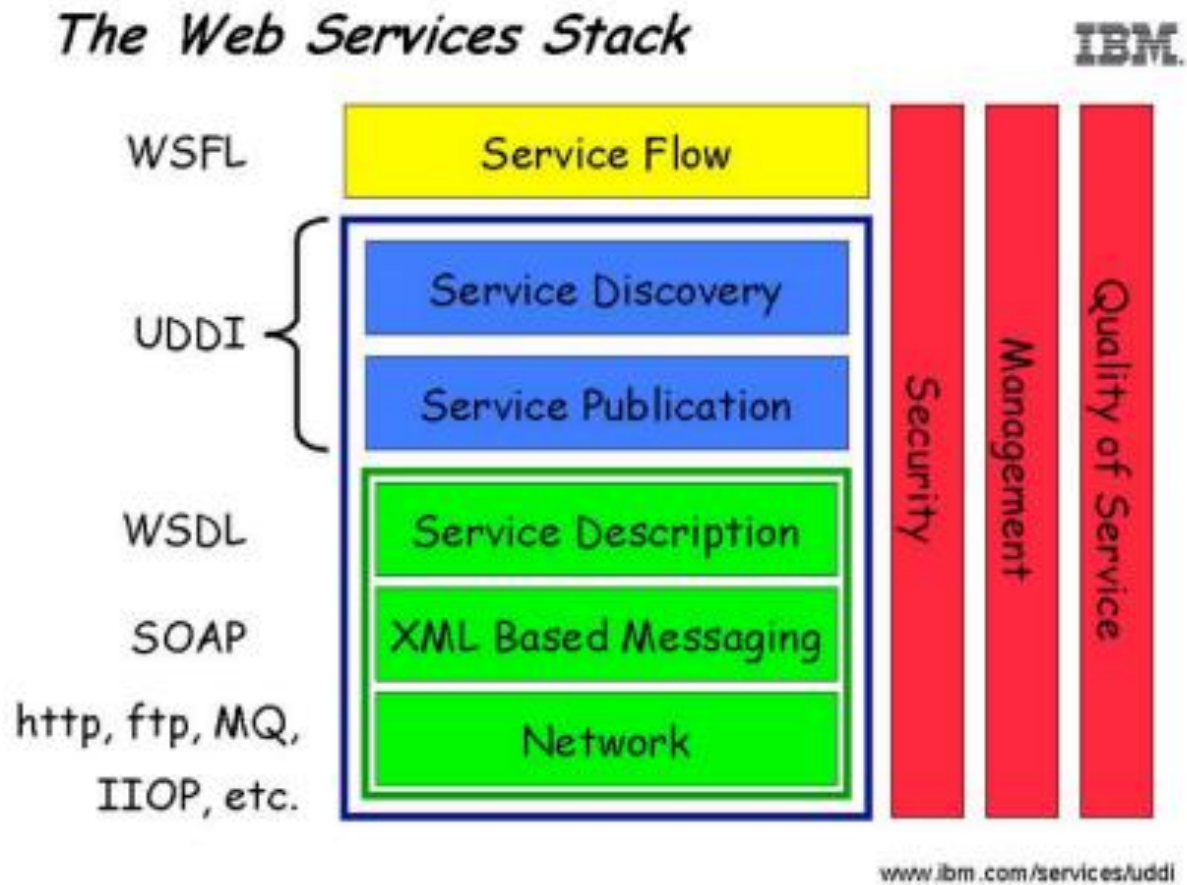
**Descripción de servicios  
(WSDL)**

**Mensajería  
(SOAP-XML-RPC)**

**Transporte de servicio  
(HTTP, SMTP, FTP, ...)**



# CAPAS



# AGENDA

- Evolución de aplicaciones distribuidas
- Problemática con las aplicaciones distribuidas tradicionales
- Aplicaciones Web
- Servicios Web
- **SOA**
- Escenarios Comunes de Servicios Web

# SOA

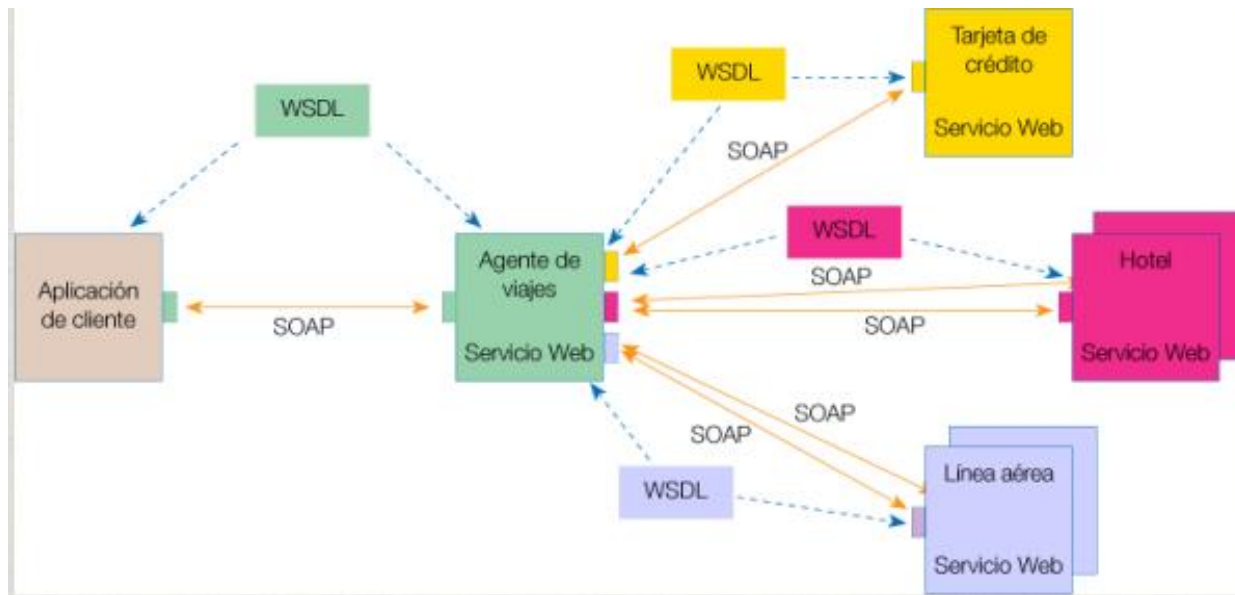
- Es un concepto de arquitectura de software que define la utilización de servicios para dar soporte a los requisitos del negocio.
- Los recursos de una red en un ambiente SOA se hacen independientes de la plataforma tecnológica que los soporta.
- Un aspecto importante es que apoyan a los procesos de negocios.

# SOA

- Una arquitectura orientada a servicios ofrece un conjunto de servicios residentes en Internet o en una intranet (servicios web).
- Existen estándares relacionados a los servicios web tales como:
  - XML
  - HTTP
  - SOAP
  - WSDL
  - UDDI

# ARQUITECTURA ORIENTADA A SERVICIOS

10/03/2014



# SOA

- Importancia de las interfaces
  - Descripción rigurosa de interfaces
  - Recomendación: desarrollar el sistema a partir de las interfaces.
- Modelo débilmente acoplados
  - Sistemas de comunicación asíncrona (Gestión de colas de mensajes)
- Interoperabilidad
  - Independencia del lenguaje y las plataformas
  - Adaptación de arquitecturas ya existentes.

# SOA

- Las metodologías tradicionales no son suficientes ni están optimizadas para SOA, por lo que se ha visto necesario crear una nueva metodología llamada SOAD (Service-Oriented Analysis and Design).
- Del mismo modo es muy importante usar un ciclo de vida apropiado dentro de la organización.
- Un ciclo de vida tipo cascada no es suficiente, se necesita algo más.

# AGENDA

- Evolución de aplicaciones distribuidas
- Problemática con las aplicaciones distribuidas tradicionales
- Aplicaciones Web
- Introducción a los servicios Web
- SOA
- **Escenarios Comunes de Servicios Web**



## VENTAJAS DE LOS SERVICIOS WEB

- Estándar aceptado por los principales fabricantes de software (incluido Microsoft).
- Permite la integración de aplicaciones entre distintos lenguajes, plataformas y sistemas operativos.
- Se comunican a través de http, por lo que puede utilizar Internet.
- Su desarrollo e implementación son bastantes sencillos, comparados con protocolos de integración anteriores (CORBA, RMI, DCOM,...).

# EJEMPLOS DE SERVICIOS WEB

- Ejemplos:
  - Yahoo
  - Google
  - Amazon
  - Wikipedia
  - Ebay