

# Estado del Arte: Servicios Web

Carlos Andrés Morales Machuca.  
Universidad Nacional de Colombia  
camoralesm@unal.edu.co

**Resumen**— Los servicios web son sistemas de software que permiten el intercambio de datos y funcionalidad entre aplicaciones sobre una red. Esta soportado en diferentes estándares que garantizan la interoperabilidad de los servicios. Los servicios web utilizan como su gran insumo el lenguaje extensible de marcado XML y se basa en una arquitectura en la que se define el servicio web a través de uno de los lenguajes estándar se publica en un directorio donde se halla la descripción anteriormente hecha y se utiliza de acuerdo a las expectativas de resolver una necesidad de acuerdo con la descripción provista. La arquitectura que mejor se ha adaptado al mundo de los servicios web es SOA brindando un enfoque que ha adoptado los negocios y ha incrementado el intercambio electrónico de datos y el comercio electrónico. Se plantea como problema la ausencia de arquitecturas que permitan a los empresarios medianos y pequeños ingresar sus organizaciones al esquema de Orientación a Servicios y Procesos de Negocios.

**Keywords**— Web Services, Servicios Web, UDDI, WSDL, SOAP, SOA.

## I. INTRODUCCIÓN

EL tipo de sociedad que el nuevo orden mundial ofrece, el desarrollo de las tecnologías de la información y la comunicación, las tendencias comerciales a través de medios electrónicos, las nuevas teorías organizacionales y el modus operandi del ser humano en el siglo XXI requieren la automatización de los procesos cotidianos y la despersonalización en muchos de ellos. Estos argumentos han sido algunos de los pilares que han hecho surgir nuevos desarrollos tecnológicos y entre ellos los que a software se refiere, creando una nueva perspectiva sobre el desarrollo de software imponiendo nuevas arquitecturas entre las que se desatacan las arquitecturas web. Sobre estas últimas, se han empezado a aprovechar una pila de estándares y protocolos que facilitan la interoperabilidad de las aplicaciones sobre la red y especialmente sobre internet naciendo así el concepto de servicio web.

## CONTENIDO

I.	Introducción.....	1
II.	Servicios Web.....	1
	1) Servicio De Transporte .....	2
	2) Mensajería Xml.....	2
	3) Descripción Del Servicio .....	4
	4) Descubrimiento De Servicios.....	4
III.	Arquitectura Orientada a Servicios.....	6
IV.	Problemática Planteada .....	7
V.	Trabajo realizado .....	7
VI.	Perspectiva de Trabajos Futuros.....	8
VII.	Conclusión.....	8
VIII.	Índice de ilustraciones .....	8
IX.	Índice de tablas .....	8
X.	Referencias .....	8

## II. SERVICIOS WEB

La World Wide Web Consortium lo define como “...un sistema de software diseñado para soportar interacción interoperable máquina a máquina sobre una red. Este tiene una interface descrita en un formato procesable por una máquina (específicamente WSDL). Otros sistemas interactúan con el servicios web en una manera prescrita por su descripción usando mensajes SOAP, típicamente enviados usando HTTP con una serialización XML en relación con otros estándares relacionados con la web” [1]. Se puede definir de manera más sencilla como un conjunto de tecnologías estándares de software para el intercambio de datos entre aplicaciones tales como SOAP, WSDL y UDDI. Estos pueden ser desarrollados en una gran variedad de lenguajes para ser implementados sobre muchos tipos de redes de computadores. El éxito de la interoperabilidad se consigue gracias es la adopción de protocolos y estándares abiertos. The Organization for the Advancement of Structured Information Standards y el World Wide Web Consortium son los responsables de la estandarización y arquitectura de los servicios web. La industria en su interés por el desarrollo de los servicios web ha creado la WS-I (Web Services Interoperability Organization) cuya intención es la integración de los estándares que garanticen y mejoren la interoperabilidad de los servicios web. Al conjunto de servicios y protocolos para los servicios web es conocido comúnmente como “Web Services Protocol Stack” y

básicamente son utilizados para definir, localizar, implementar y hacer que un servicio web interactúe con otro. Este conjunto está conformado esencialmente de cuatro subconjuntos:

- Servicio de transporte
- Mensajería XML
- Descripción del servicio
- Descubrimiento de Servicios

### 1) Servicio De Transporte

Es el encargado del transporte de los mensajes entre aplicaciones sobre la red. Incluye varios protocolos del nivel de aplicación. A continuación se relata sobre los más utilizados.

#### a) HTTP (HyperText Transfer Protocol):

Protocolo del nivel de aplicación más utilizado en la Internet. Es el protocolo que define la sintaxis y la semántica utilizada para la arquitectura web. En el contexto de los servicios web es utilizado para la transferencia de las transacciones XML a través de la red utilizando los mismos principios del HTML.

#### b) FTP (File Transfer Protocol):

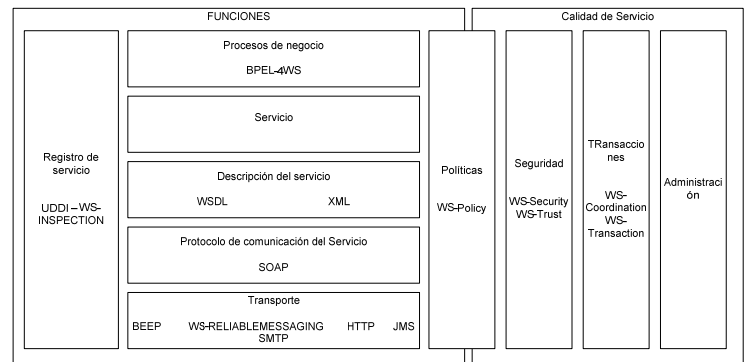
Es un protocolo de la capa de aplicación encargado de los servicios de transmisión de archivos a través de redes soportadas sobre TCP. En el ámbito de los servicios web el FTP permite realizar modificaciones en equipos remotos evitando el uso de permisos sobre los archivos en la máquina cliente en sistemas operativos diferentes a Windows.

#### c) SMTP (Simple Mail Transfer Protocol):

Es un estándar de la capa de aplicación ampliamente utilizado para el envío de mensajes de correo electrónico a través de Internet. Es un estándar de Facto basado en texto, que requiere como cliente software de tipo POP3 o IMAP.

#### d) BEEP (Block Exensible Exchange Protocol):

Es un protocolo del nivel de aplicación, también conocido como BXXP, está diseñado para la interacción asíncrona punto a punto sobre una red TCP/IP. Fue estandarizado por el IETF y provee un marco para administrar las conexiones punto a punto, autenticación, transporte de mensajes y manejo de errores [2].

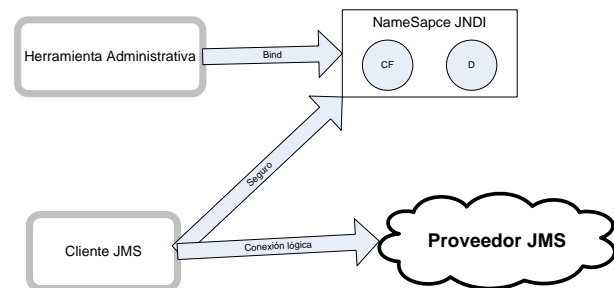


**Ilustración 1 Arquitectura general de los servicios web**

#### e) JMS (Java Message Service):

Es una aplicación de interface de programación para JAVA (API) para el envío de mensajes entre dos o más clientes. Soporta dos modelos el modelo punto a punto y el modelo de publicación y suscripción. Una aplicación JMS está compuesta por las siguientes partes [3]:

- Un proveedor JMS que implementa las interfaces que proveen las características de administración y el control.
- Clientes JMS que son los componentes escritos en JAVA que producen y consumen los mensajes.
- Los Mensajes que son los objetos dato entre los clientes JMS.
- Objetos administradores que son objetos configurados previamente por un administrador del sistema para el uso de los clientes.
- Clientes Nativos que son programas que utilizan los mensajes de la API de manera similar que la API JMS.



**Ilustración 2 Arquitectura API JMS [3]**

### 2) Mensajería XML

Es el conjunto encargado de la codificación de los mensajes en XML estándar y pueda así ser interpretado en cualquiera de los nodos de la red. Los componentes más utilizados en este conjunto son los siguientes:

#### a) REST (Representational State Transfer):

Fielding [4] da la siguiente definición: “estilo de arquitectura

de software para sistemas hipermedias distribuidos tales como la World Wide Web". En resumen, es un conjunto de principios para el diseño de redes, que es utilizado comúnmente para definir una interfaz de transmisión sobre HTTP de manera análoga a como lo hace SOAP. Aunque REST como tal no es un estándar, posee un conjunto de estándares tales como HTML, URL, XML, GIF, JPG y tipos MIME.

Los principios de REST son:

- Escalabilidad de la interoperabilidad con los componentes.
- Generalidad de Interfaces.
- Puesta en funcionamiento independiente.
- Compatibilidad con componentes intermedios.

b) *RPC (Remote Procedure Calls):*

Es una tecnología de software que permite ejecutar una rutina en un equipo o segmento de red de manera remota. Es un paradigma popular para la implementación de sistemas distribuidos bajo arquitecturas cliente servidor.

c) *XML-RPC:*

Es un protocolo de llamada remota que utiliza XML como lenguaje de codificación y HTTP como mecanismo de transporte. Es un protocolo sencillo ya que solo define algunos tipos de datos y comandos.

Nombre	Etiqueta de Ejemplo	Descripción
array	<pre>&lt;array&gt; &lt;data&gt;   &lt;value&gt;&lt;i4&gt;144&lt;/i4&gt;&lt;/value&gt;   &lt;value&gt;&lt;string&gt;Dato ejemplo&lt;/string&gt;&lt;/value&gt;   &lt;value&gt;&lt;i4&gt;4&lt;/i4&gt;&lt;/value&gt; &lt;/data&gt; &lt;/array&gt;</pre>	Arreglo de valores.
base64	<pre>&lt;base64&gt;eW91df3bid0IHJIYWQdGhpcyE= &lt;/base64&gt;</pre>	Datos binarios codificados en Base64
boolean	<pre>&lt;boolean&gt;0&lt;/boolean&gt;</pre>	Valor lógico booleano (0 ó 1)
date/time	<pre>&lt;dateTime.iso8601&gt;20080517T04:18:51&lt;/d ateTime.iso8601&gt;</pre>	Fecha y Hora en formato ISO 8601
double	<pre>&lt;double&gt;-22.31&lt;/double&gt;</pre>	Número en punto flotante de doble precisión
integer	<pre>&lt;i4&gt;45&lt;/i4&gt; or &lt;int&gt;22&lt;/int&gt;</pre>	Número entero
string	<pre>&lt;string&gt;Dato de ejemplo&lt;/string&gt;</pre>	Cadena de caracteres.
struct	<pre>&lt;struct&gt; &lt;member&gt;   &lt;name&gt;Ejm&lt;/name&gt;</pre>	Arreglo de estructura.

	<pre>&lt;value&gt;&lt;i4&gt;2&lt;/i4&gt;&lt;/value&gt; &lt;/member&gt; &lt;member&gt;   &lt;name&gt;Ejm1&lt;/name&gt;   &lt;value&gt;&lt;i4&gt;4&lt;/i4&gt;&lt;/value&gt; &lt;/member&gt; &lt;/struct&gt;</pre>	
nil	<pre>&lt;nil/&gt;</pre>	Representa a nulo.

**Tabla 1 Definición de datos XML-RPC**

Existen implementaciones de XML-RPC específicas para ActionScript, Delphi, C++, .NET, OClam, Common LISP, PHP y otros.

d) *XML (eXtended Markup Language):*

XML es uno de los lenguajes más utilizados para el intercambio de datos sobre la web. Su desarrollo se remonta en el año 1996 por el grupo de trabajo de la World Wide Web Consortium lanzando su primera versión el 10 de Febrero de 1998 [2]. El lenguaje XML está concebido para describir objetos de datos llamados Documentos XML y describir de cierta forma los programas que los procesan. Está restringido bajo la norma ISO 8879 el Estándar Generalized Markup Language [4]. Un documento XML es un objeto de datos que está bien formado, y se dice que lo está cuando tomado en su conjunto coincide con la producción del documento etiquetado, reúne todas las especificaciones de formato definidas y cada una de las entidades que se llaman directa o indirectamente están también bien definidas [4].

El XML es un lenguaje etiquetado, característica que le permite definir objetos de datos estructurados en partes bien definidas llamadas elementos. Una etiqueta es una señal realizada dentro del documento XML que delimita un segmento definido y con sentido de este documento.

Este es un ejemplo de XML:

```
<Edita_Mensaje>
<Mensaje>
  <Remite>
    <Nombre>Nombre del remitente</Nombre>
    <Correo>Correo del remitente</Correo>
  </Remite>
  <Destinatario>
    <Nombre>Nombre del destinatario</Nombre>
    <Correo>Correo del destinatario</Correo>
  </Destinatario>
  <Text>
    <Asunto>
      Este es un documento sencillo sin atributos
    </Asunto>
    <Parrafo>
      Este es un documento sencillo
    </Parrafo>
  </Text>
</Mensaje>
</Edita_Mensaje>
```

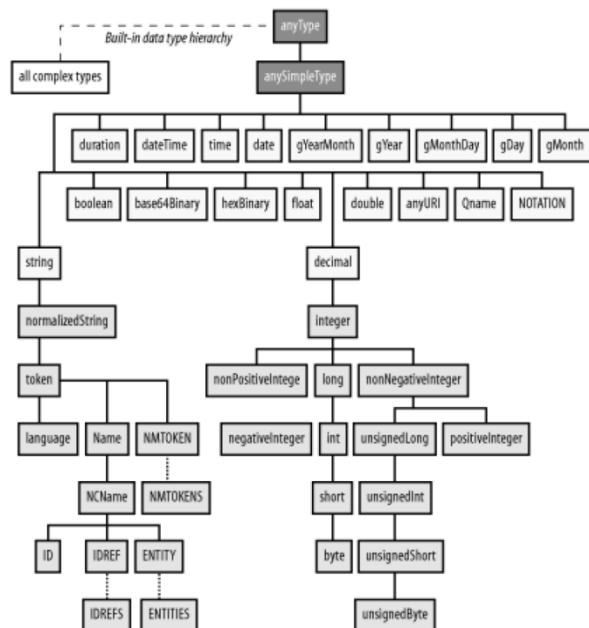
**Ilustración 3 Ejemplo de XML**

Adjunto al documento XML existe una definición de tipo de documento (DTD) en donde se describe la estructura y la definición de los datos de un documento XML. Una DTD describe normalmente los elementos, que son los segmentos etiquetados, la estructura, que es el orden en el cual van los elementos y el nivel de anidamiento.

```
<!ELEMENT Mensaje (Remite, Destinatario, Asunto, Text)*>
  <!ELEMENT Remite (Nombre, Correo)>
    <!ELEMENT Nombre (#PCDATA)>
    <!ELEMENT Correo (#PCDATA)>
  <!ELEMENT Destinatario (Nombre, Mail)>
    <!ELEMENT Nombre (#PCDATA)>
    <!ELEMENT Correo (#PCDATA)>
  <!ELEMENT Asunto (#PCDATA)>
  <!ELEMENT Text (Parrafo)>
    <!ELEMENT Parrafo (#PCDATA)>
```

#### Ilustración 4 Ejemplo de DTD

La definición de XML es bastante larga, así que en la figura 1 se describe de manera sencilla [5].



#### Ilustración 5 Definición de XML

##### e) SOAP (Simple Object Access Protocol):

SOAP es un protocolo de la capa de aplicación para el intercambio de mensajes basados en XML sobre redes de computadores. Básicamente es una vía de transmisión entre un SOAP Sender y un SOAP Receiver, pero los mensajes SOAP deben interactuar con un conjunto de aplicaciones para que se pueda generar un “diálogo” a través de mensajes SOAP. Un mensaje SOAP es la unidad fundamental de una comunicación entre nodos SOAP [6]. SOAP es básicamente un paradigma de

una sola vía pero con la ayuda de las aplicaciones se puede llegar a crear patrones más complejos. SOAP básicamente está constituido por:

- Un marco que describe el contenido del mensaje e instrucciones de proceso.
- Un conjunto de reglas para representar los tipos de datos definidos.
- Convenciones para representar llamadas a procedimientos remotos y respuestas.
- Y algunos lineamientos entre SOAP y HTTP [2].

#### 3) Descripción Del Servicio

El servicio web debe contar con una interfaz pública la cual es descrita por un formato llamado WSDL (Web Services Description Languages).

##### a) WSDL (Web Services Description Language):

WSDL es un tipo de documento XML que describe lo que hace un servicio web, donde se encuentra y la forma de ser invocado [7]. Este provee información muy importante para los desarrolladores, este lenguaje describe el formato de los mensajes que utiliza y a cuales puede responder. Siempre un documento XML WSDL presenta los siguientes elementos:

- Tipos: Tipos de datos usados por los mensajes.
- Mensaje: Que datos son enviados desde un nodo a otro.
- Tipo de puerto: Define las operaciones que pueden ser llamadas.
  - Operación: Define la configuración de mensajes de entrada, salida y error.
  - Entrada: Mensaje que es enviado hacia el servidor.
  - Salida: Mensaje enviado hacia el cliente.
  - Falta: Error en el envío de un mensaje.
- Límite: Es la descripción del protocolo que se está utilizando para transportar el mensaje que puede ser HTTP POST, HTTP GET, SOAP y MIME.
- Servicio: Define una colección de puertos (nodos); el puerto especifica una dirección para el límite definiendo así la comunicación para un nodo específico.

#### 4) Descubrimiento De Servicios

UDDI (Universal Description Discovery and Integration):

UDDI es un marco independiente de la plataforma para describir servicios, negocios e integrar servicios de negocios. La estructura de UDDI está basada sobre los servicios estándares de la web, lo que quiere decir que UDDI es accesible como otros servicios web. UDDI es un esfuerzo de la industria iniciada en Septiembre de 2000 por Ariva, IBM, Microsoft y otras 33 compañías [9]. Los propietarios de los

Servicios Web los publican en el registro UDDI. Una vez publicados se mantienen allí apuntadores a la descripción del Servicio Web y al servicio. UDDI permite a los clientes buscar tal registro, encontrar el servicio deseado y extraer sus detalles. Estos detalles incluyen el punto de invocación así como otras características del servicio y su funcionalidad. La estructura de datos con UDDI está compuesta en cuatro partes:

businessEntity  
businessService  
bindingTemplate  
tModel [10]

#### businessEntity

Describe al proveedor del servicio web. Tiene datos como nombre de compañía, detalle de contacto y otra información del negocio.

#### businessService

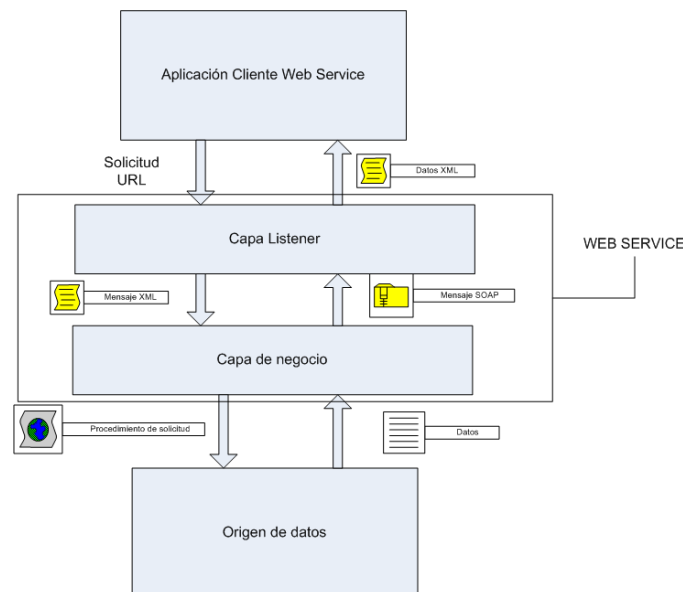
Describe un conjunto lógico de uno o muchos servicios web.

#### bindingTemplate

Describe un único Servicio Web, describe toda la información técnica para que el cliente pueda interactuar con él.

#### tModel

Representa especificaciones técnicas, metadatos sobre las especificaciones del documento, el nombre puntero URL, es presentado en forma de un documento WSDL.



**Ilustración 6** Funcionamiento de Servicio web

Algunas especificaciones adicionales han sido desarrolladas o están empezando a ser desarrolladas con el ánimo de extender las capacidades de los servicios web. De manera habitual estas especificaciones son nombradas como ws-??. Algunas de las más importantes son:

#### WS-Security:

Es un protocolo de comunicaciones encargado de proveer seguridad a las aplicaciones de Servicios Web. Fue desarrollado originalmente por Microsoft, IBM, Verisign y Forum Systems, ahora el protocolo es llamado WSS. WSS ofrece mejoras en el sistema de mensajería SOAP para proveer calidad en la protección a través de la integridad de mensajes, confidencialidad y autenticación [12]. Define como usar encriptación XML y firma XML en SOAP, es una alternativa diferente a HTTPS.

#### WS-Reliability

Es un protocolo basado en SOAP para el intercambio de mensajes con distribución garantizada [13], sin duplicados y garantizando el orden del mensaje.

#### WS-ReliabilityMessaging

Esta especificación describe un protocolo que permite enviar mensajes SOAP de manera confiable entre sistemas distribuidos en presencia de fallas de los sistemas, componentes o aplicaciones. El objetivo de esta especificación es asegurar que el mensaje enviado por el emisor sea recibido por el receptor. La confiabilidad en los Servicios Web es algo difícil de definir, pero se puede realizar un símil de WSRM para XML como JMS para Java.

#### WS-Adressing

Esta es una especificación de mecanismos de transporte que permite a los Servicios Web comunicar información direccionada. Tiene una estructura compuesta en principio por dos partes. La primera, es una estructura para comunicar una referencia al nodo final del servicio web, y la segunda, es un conjunto de propiedades de direccionamiento con las cuales se asocia la información direccionada con un mensaje en particular [14]. Las propiedades de direccionamiento son:

- Destinación de mensaje URI
- Origen del nodo final.
- Reenvío de nodo final.
- Falla del nodo final.
- Acción.
- Identificador único del mensaje.
- Relación con mensajes previos.

#### WS-Transaction

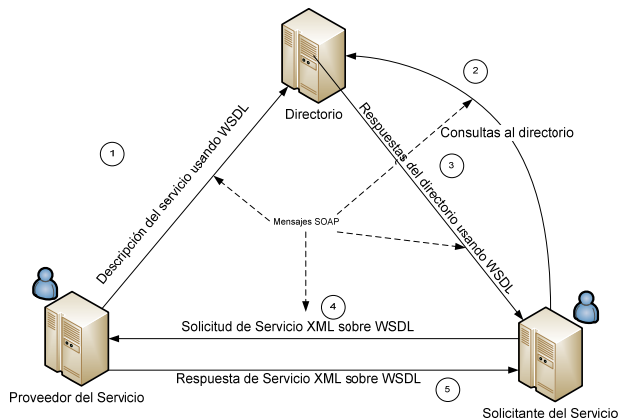
Es una especificación desarrollada inicialmente por Microsoft, IBM y BEA Systems. Esta describe tipos de coordinación que son usadas con el marco extensible de coordinación descrito en la especificación WS-Coordination [15].

#### WS-Coordination

Es una especificación que describe un marco extensible para proveer protocolos que coordinen las acciones de aplicaciones



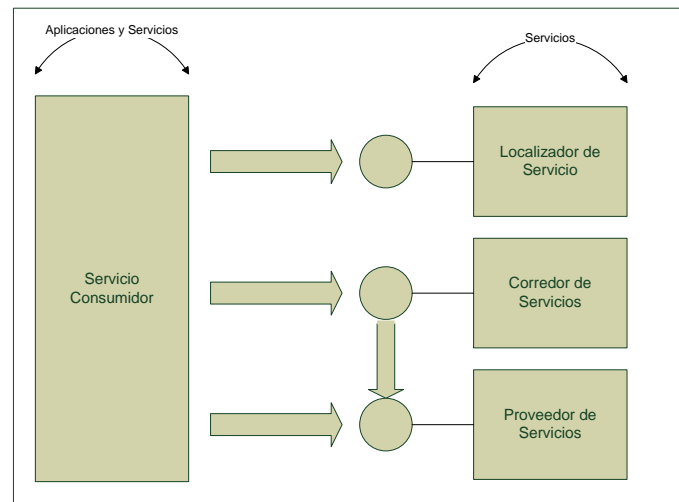
distribuidas. El marco definido en tal especificación habilita un servicio de aplicación para crear un contexto necesario para propagar una actividad a otros servicios y registrarlos a protocolos de coordinación [16].



**Ilustración 7 Modelo básico del funcionamiento de los Servicios Web**

### III. ARQUITECTURA ORIENTADA A SERVICIOS

Es la arquitectura más difundida en el mundo de los servicios Web. SOA es un modelo arquitectónico de software creado y usado para diseñar modelos de negocio empaquetados como servicios. Una solución SOA es un diseño aplicando conceptos SOA, para lo cual es necesaria la utilización de un conjunto de herramientas de software, tecnologías y plataformas específicas [17]. El enfoque de esta arquitectura hace que todo el modelo de los Web Services gire en torno a los negocios. Los gerentes de las tecnologías de la información tienen el reto de hacer reducir los costos y aprovechar al máximo las tecnologías existentes, pero también deben servir lo mejor posible a un conjunto de clientes, ser más competitivos y brindar las mejores respuestas a las prioridades estratégicas de los negocios. Las empresas de hoy en día tienen diferentes sistemas de información en sus organizaciones, de diferentes épocas y con tecnologías diversas, las características principales del sector empresarial y comercial son heterogeneidad y cambio. Siempre fue un terrible problema combinar tecnologías de diferentes proveedores y diferentes plataformas. El cambio también está presente debido a factores contextuales tales como la globalización y el comercio electrónico que nos obligan a pensar de manera rápida en formas efectivas de combatir los obstáculos interpuestos por la heterogeneidad y el cambio expuesto. Una de las soluciones planteadas para este tipo de situación precisamente es SOA, y es mostrada como uno de los grandes logros de la ingeniería de software. En [18] incluye la noción de servicios, en donde la describe como un componente definiéndola como “una unidad de código ejecutable que provee un encapsulamiento de caja negra física de servicios relacionados. Sus servicios pueden ser únicamente accedidos por una interface publicada consistente, que incluye una interacción estándar. Un componente debe ser capaz de ser conectado con otros componentes para un largo grupo”.



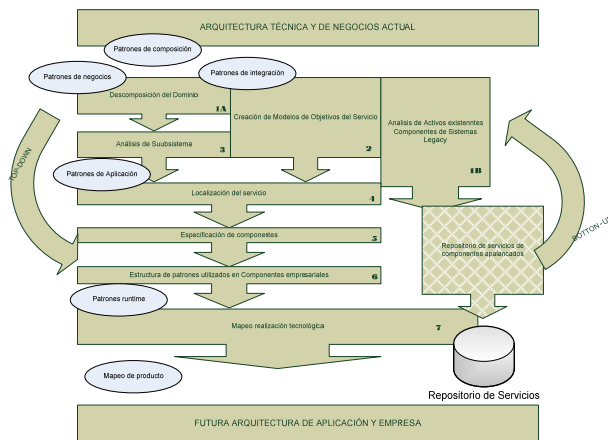
**Ilustración 8 Terminología Orientada a Servicios**

En la ilustración 7 se define:

- **Servicio:** entidades lógicas, los contratos definidos por una o más interfaces publicadas.
- **Proveedor de servicios:** La entidad de software que implementa una especificación de servicio.
- **Consumidor de servicio:** La entidad de software que llama a un proveedor de servicio. Un consumidor de servicio puede ser una aplicación de usuario final u otro servicio.
- **Localizador de servicio:** Una clase específica de servicio que se comporta como un registro y permite la búsqueda de interfaces de proveedores de servicios y localización de servicios.
- **Corredor de servicios:** Es una tipo especial de servicio que puede pasar ante una solicitud de servicio uno o varios servicios [19].

Una arquitectura orientada a servicios es descrita como un conjunto de servicios que apuntan a los negocios que son combinados (composición y orquestación) para cumplir con los objetivos del negocio. Las Tecnologías de la Información y la Comunicación permiten a través de sus herramientas cumplir con esta tarea a cabalidad.

Existen unos pasos principales para aplicar el enfoque SOA y se puede tomar como un patrón:



### Ilustración 9 Siete pasos del enfoque SOA [19]

Una arquitectura Orientada a Servicios es una forma de arquitectura de sistemas distribuidos que es típicamente caracterizada por las siguientes propiedades [1]:

- **Vista lógica:** Es una vista que proporciona una imagen de los componentes del sistema tales como bases de datos, procesos de negocios, programas, etc., explicando que hace cada uno de ellos, normalmente llevándolos a la operación del nivel del negocio.
- **Orientación al mensaje:** Se define el servicio en términos de los mensajes intercambiados por el agente solicitante y el agente proveedor. En SOA es abstraído algunas características de los agentes tales como la estructura de la base de datos, sus lenguajes de implementación, estructuras de procesos, etc. SOA se preocupa por los detalles que son expuestos en la descripción del servicio.
- **Orientación a la descripción:** Un Servicio Web es descrito por metadatos procesables por máquina. La descripción debe soportar la naturaleza pública de la SOA. La semántica del servicio debe ser definida completamente en su descripción.
- **Granularidad:** Los servicios deben tener la tendencia a realizar un pequeño número de operaciones con una gran cantidad de mensajes.
- **Orientación a la red:** Los Servicios Web deben conservar la tendencia de ser concebidos para ser usados sobre una red, sin embargo no es un requerimiento absoluto.
- **Plataforma Neutral:** Los mensajes deben ser creados para una plataforma neutral, utilizando un lenguaje estándar a través de las interfaces. XML es el lenguaje que mejor cumple con esta restricción.

#### IV. PROBLEMÁTICA PLANTEADA

La nueva dinámica de la sociedad en el mundo, la globalización, el desarrollo de las tecnologías de la información y las comunicaciones, la evolución de las teorías organizacionales y sociales han apoyado el desarrollo de nuevas herramientas tecnológicas que permiten la

conectividad e intercambio electrónico de datos fortaleciendo el muy nombrado concepto de “aldea global”. Los Servicios Web hacen parte del conjunto de herramientas que apoyan este paradigma social y su desarrollo científico e inversión de investigación ha sido realizado en gran parte por organismos de origen privado que buscan establecer estándares que dependerían totalmente de sus especificaciones y parámetros. Una forma de brindar equidad en el proceso de formación de una herramienta tan poderosa y permitir el acceso a toda la comunidad de negocios en el mundo es la incorporación de software libre y software de código abierto para el desarrollo integral de los Servicios Web que cumpla todas las características de interoperabilidad, seguridad, integridad, acoplamiento y todas las características que presenta un Servicio Web confiable. Además se debe observar si las utilidades del software libre son capaces de proveer una extensión a la arquitectura actual utilizada para la implementación de los Servicios.

## V. TRABAJO REALIZADO

Algunos organismos gubernamentales han visto en la Arquitectura Orientada a Servicios una alternativa de implementación de los programas con un enfoque social. El Ministerio de Salud Pública de la República de Cuba ha definido en la primera década del siglo XXI la informatización como una de sus prioridades así que decide desarrollar una arquitectura basada en componentes y orientada a servicios un sistema que permita integrar diferentes áreas de manera eficiente. La estrategia del Sistema Nacional de Salud fundamenta el desarrollo de sus aplicaciones sobre Software Libre [20]. Investigadores del Departamento de Geografía de la Universidad de Colorado están estudiando la implementación de Sistemas de Información Geográfica aplicando el modelo de Arquitectura Orientada a Servicios utilizando la versatilidad de la interoperabilidad ofrecida por los Servicios Web apuntando a la reducción de altos costos, complejidad y requerimientos especiales desplegando los datos y capacidad de geo-procesamiento sobre la World Wide Web con alternativas sin costo de Open Source Software ante los software propietario, en los campos de sistemas operativos, servidores web y sistemas de administración de bases de datos relacionales [21]. Los investigadores de la Universidad de Colorado implementaron un prototipo en México de un Sistema con capacidades de geo-procesamiento que no están presentes en ningún producto comercial de Web-GIS. En la última década la verificación formal de componentes de hardware y software atraen los intereses tanto de la academia como de la industria. La expansión de técnicas de razonamiento automatizado requiere de herramientas que sean de fácil uso y permitan el intercambio de datos a través de los formatos estándar. En FB Informatik, una publicación de Universität des Saarlandes, en Alemania Jürgen Zimmer presenta The MathWeb Software Bus, que es el primer paso a los servicios de razonamiento reutilizables [22]. The MathServe system está disponible bajo la licencia GNU Public License. El 5 de Mayo de 2008 la Free Software Foundation

organizó en la ciudad de San Francisco un evento llamado *Liberating Web Services* donde resalta la importancia de vincularse a esta nueva tendencia y el incremento masivo de las aplicaciones web. En este evento se discutieron algunas preguntas como ¿Qué significa libertad de los usuarios y las aplicaciones en el entorno de los Servicios Web? ¿Qué riesgos existen? ¿Podría la comunidad de software libre y la Free Software Foundation asegurar la libertad de los usuarios en este nuevo ambiente tecnológico?

El paradigma de cliente-servidor ha sido desplazado por otro tipo de paradigmas tales como el desarrollo web, peer to peer o grid, los cuales son implementados sobre una red. El enfoque de los servicios web orientado a los servicios y procesos de negocio también empiezan a hacer parte de los paradigmas anteriormente mencionados [23]. En la National Chiao Tung University de Taiwan trabajan sobre modelos difusos para la toma de decisiones sobre arquitecturas orientadas a servicios, planteando un nuevo esquema de arquitectura planteando ventajas sobre las arquitecturas tradicionales de flexibilidad, compatibilidad y administración del flujo de trabajo.

## VI. PERSPECTIVA DE TRABAJOS FUTUROS

Los países en vía de desarrollo presentan la necesidad de hacer parte de la dinámica presente en el orden mundial contemporáneo incrementado por la evolución de las Tecnologías de la Información y las Comunicaciones para integrarse en lo que muchos científicos de las ciencias sociales denominan “Sociedad de la Información”. Uno de los grandes motores de la economía y dinámica social de estos países, es el desarrollo de la mediana y pequeña empresa, las cuales no tienen acceso debido a los costos a herramientas tecnológicas que le faciliten la integración social requerida. Una posible solución, y sería el campo en el que se centraría la investigación, es proponer una arquitectura que se adapte a la infraestructura actual de las organizaciones de pequeña y mediana envergadura para brindarles acceso al paradigma Orientado a Servicios, mediante el cual podrán sacar ventaja competitiva y volver más provechosa la cadena de valor de este nicho empresarial.

Por otro lado, existen diversos sectores públicos que no cuenta con la suficiente asignación presupuestal de parte de los entes gubernamentales encargados de definir los presupuestos necesarios, pero que igual necesitan este nivel de integración para cumplir con el objeto social con el que fueron originalmente concebidas; podemos observar carencias de este tipo en entidades del área de la salud, educación, promoción social y comunitaria, organizaciones sin ánimo de lucro, organizaciones religiosas, científicas entre otras.

Los entes gubernamentales y las universidades públicas deben enfatizar la investigación como proveedor de productos y servicios con objetivo social para intentar disminuir la brecha existente en el campo tecnológico entre las diferentes clases y grupos económicos.

## VII. CONCLUSIÓN

El uso de estándares en el ámbito tecnológico de los Servicios Web ha permitido una evolución veloz y segura, manteniendo un horizonte unívoco para todas las organizaciones que incursionan en este tema. La esencia que permite que dos entidades de software autómatas sean capaces de dialogar entre sí sin la intervención humana arrojando insumos y resultados para un sistema de información y además esto presente un valor agregado en el sector productivo hace que los Servicios Web sean una alternativa cautivadora. El interés de grandes compañías de software líderes en campos específicos pero diversos, de comunidades científicas y académicas, comunidades tecnológicas y comunidades sociales hace resaltar el buen futuro que le depara al paradigma Orientado a Servicios. Los países en vía de desarrollo deben estar al frente de este modo de concebir los sistemas de información y es obligación de toda su sociedad científica llevar al país a la vanguardia en este tema.

## VIII. ÍNDICE DE ILUSTRACIONES

Ilustración 1 Arquitectura general de los servicios web.....	2
Ilustración 2 Arquitectura API JMS (3).....	2
Ilustración 3 Ejemplo de XML .....	3
Ilustración 4 Ejemplo de DTD.....	4
Ilustración 5 Definición de XML.....	4
Ilustración 6 Funcionamiento de Servicio web .....	5
Ilustración 7 Modelo básico de los Servicios Web .....	6
Ilustración 8 Terminología Orientada a Servicios.....	6
Ilustración 9 Siete pasos del enfoque SOA (19) .....	7

## IX. ÍNDICE DE TABLAS

Tabla 1 Definición de datos XML-RPC.....	3
------------------------------------------	---

## X. REFERENCIAS

- [1]. **W3C Consortium**. Web Services Architecture. [En línea] 11 de Febrero de 2004. [Citado el: 22 de Abril de 2008.] <http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/#whatis>.
- [2]. **Van de Putte, Geert, y otros**. *Using Web Services for Bussiness Integration*. s.l. : IBM, 2004. 0738425486.
- [3]. **Sun Microsystems, Inc**. Java™ Message Service Tutorial. [En línea] 2002. [Citado el: 15 de Mayo de 2008.] [http://java.sun.com/products/jms/tutorial/1\\_3\\_1-fcs/doc/basics.html](http://java.sun.com/products/jms/tutorial/1_3_1-fcs/doc/basics.html).
- [4]. **Fielding, Roy T**. Architectural Styles and the Design of Network-based Software Architectures. *PhD Thesis*. [En línea] 2000. [Citado el: 22 de Abril de 2008.] <http://roy.gbiv.com/pubs/dissertation/top.htm>.
- [5]. **Graham, Steve, y otros**. *Building Web Services with Java. Making Sense of XML, SOAP, WSDL and UDDI*. United States : Sams Publishing, 2001. 0-672-32181-5.



- [6]. **W3C Consortium**. Extensible Markup Language (XML) 1.0 (Fourth Edition). *W3C Consortium*. [En línea] 16 de Agosto de 2006. [Citado el: 22 de Abril de 2008.] <http://www.w3.org/TR/2006/REC-xml-20060816/>.
- [7]. **Tidwell, Doug, Snell, James y Kulchenko, Pavel**. *Programming Web Services with SOAP*. s.l.: O'Really, 2002.
- [8]. **W3C Consortium**. SOAP Version 1.2 Part 1: Messaging Framework (Second Edition). *W3C Consortium*. [En línea] 27 de Abril de 2007. [Citado el: 22 de Abril de 2008.] <http://www.w3.org/TR/2007/REC-soap12-part1-20070427/#encapsulation>.
- [9]. *Modeling of Web Services Flow*. **Thomas, Johnson, Mathews, Thomas y Guinea, George**. s.l.: IEEE International Conference on E-Commerce (CEC'03), 2003. 0-7695-1969-5/03.
- [10]. **Benz, Brian**. *XML Programming Bible*. s.l.: John Wiley & Sons, Incorporated, 2004.
- [11]. **bea systems**. [edocs.bea.com](http://edocs.bea.com). *Publishing and Finding Web Services Using UDDI*. [En línea] Web logic 7.0. [Citado el: 28 de Abril de 2008.] <http://edocs.bea.com/wls/docs70/webserv/uddi.html>.
- [12]. **IBM Corporation**. Web Services Security. [En línea] 05 de Abril de 2002. [Citado el: 12 de Mayo de 2008.] <http://www-128.ibm.com/developerworks/library/specification/ws-secure/>.
- [13]. **Iwasa, Kazunori, y otros**. Web Services Reliable Messaging TC WS-Reliability 1.0. *OASIS*. [En línea] 15 de Noviembre de 2004. [Citado el: 17 de Mayo de 2008.] [http://docs.oasis-open.org/wsrn/ws-reliability/v1.1/wsrn-ws\\_reliability-1.1-spec-os.pdf](http://docs.oasis-open.org/wsrn/ws-reliability/v1.1/wsrn-ws_reliability-1.1-spec-os.pdf).
- [14]. **WC Consortium**. Web Services Addressing 1.0 - Core. [En línea] 9 de Mayo de 2006. [Citado el: 18 de Mayo de 2008.] <http://www.w3.org/TR/2006/REC-ws-addr-core-20060509/>.
- [15]. **Cox, William, y otros**. Web Services Transaction (WS-Transaction). [En línea] 30 de Enero de 2004. [Citado el: 22 de Mayo de 2008.] <http://dev2dev.bea.com/pub/a/2004/01/ws-transaction.html>.
- [16]. **Cabrera, Luis Felipe, y otros**. Web Services Coordination (WS-Coordination). [En línea] Agosto de 2005. [Citado el: 22 de Mayo de 2008.] <http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-tx/WS-Coordination.pdf>.
- [17]. *The Art of Service Orientation, SOA WebServices Journal*. **Anantha Rangachar, Raghu**. 6, Montvale, US : SYS-CON MEDIA, 2006, Vol. 6. 1535-6906.
- [18]. **Allen, Paul**. *Component-Based Development for Enterprise Systems*. s.l.: Cambridge University Press, 1998.
- [19]. **Endrei, Mark, y otros**. *Patterns: Service-Oriented Architecture and Web Services*. s.l.: IBM Corp., 2004. 073845317X.
- [20]. *Registro Informatizado de Salud (RIS)*. **Delgado Ramos, Ariel, Cabrera Hernández, Mima y Juncal, Virginia**. 2, Habana : Centro Nacional de Información de Ciencias Médicas, República de Cuba , 2006.
- [21]. *Building Web-Based Spatial Information Solutions around Open Specifications and Open Source Software*. **Anderson, Geoffrey y Moreno Sanchez, Rafael**. 4 pag 447-466, s.l.: Blackwell Sinergy, 2003, Vol. 7. doi:10.1111/1467-9671.00158 .
- [22]. **Zimmer, Jürgen y Autexier, Serge**. The MathServe System for Semantic Web Reasoning Services. [aut. libro] Springer Berlin / Heidelberg. *Automated Reasoning*. s.l.: Springer Berlin / Heidelberg, 2006.
- [23]. **Tsai, Min-Jen ,Wang, Chen-Sheng**. *A computing coordination based fuzzy group decision-making (CC-FGDM) for web service oriented architecture*. Expert Systems with Applications; Elsevier. May2008, Vol. 34 Issue 4, p2921-2936, 16p.

#### SOBRE EL AUTOR



**Carlos Andrés Morales**. Ingeniero de Sistemas Egresado de la Universidad Autónoma de Colombia. Realiza sus estudios de Maestría en Ingeniería de Sistemas y Computación en la Universidad Nacional de Colombia – Sede Bogotá. Desempeña funciones concernientes a los Sistemas de Información en la Caja de Previsión Social de la Universidad Nacional de Colombia – Nivel Nacional.