

---

# Servicios Web Semánticos

Pablo Valledor Pellicer  
UO80572@uniovi.es  
*Universidad de Oviedo*

## Resumen

Con la aparición de los servicios Web se abre la posibilidad de conseguir la automatización de tareas entre plataformas y servicios diferentes. Para poder llevar a cabo este cometido, se requiere de una plataforma con una semántica bien definida, que permita el descubrimiento y la composición de manera automática de los servicios. La siguiente generación de Web (llamada Web extendida), proporcionará la infraestructura necesaria para lograr este objetivo mediante la utilización de los servicios Web semánticos (denominados SWS). Este artículo tiene como objetivo servir de introducción a los SWS, analizando los frameworks actuales, que permiten la integración de este tipo de servicios en el entorno Web, y las características y funcionalidades aportadas por estos servicios.

Palabras claves: Servicios Web, Web semántica, SWS (Servicios Web semánticos), OWL-S, WSMOL, WSML, WSMX, SAWSDL.

## 1 INTRODUCCIÓN

Actualmente, dentro del campo empresarial, las nuevas aplicaciones para Internet se basan, fundamentalmente, en el software que surgió como alternativa a las páginas estáticas que proporcionan catálogos o simples formularios. En realidad, son una respuesta a las demandas de los usuarios que buscaban dinamismo y a las necesidades de la empresa cuyo objetivo es la globalización.

Las aplicaciones empresariales han ido evolucionando en paralelo con el desarrollo masivo de los procesos de negocio y de Internet en estos últimos años, aumentando las dimensiones y la complejidad de las mismas. Cada vez es más importante desarrollar arquitecturas que sean escalables y robustas. Para ello se aplican las nuevas tecnologías que reducen la complejidad del desarrollo.

Estas aplicaciones ya no se centran, solamente, en proporcionar ciertas funcionalidades sino que, cada vez, resulta más necesario que incorporen una serie de propiedades que les permitan diferenciarse unas de otras. Por otra parte, las empresas buscan resultados consistentes y productivos que se obtengan en poco tiempo y cuyos procesos sean eficientes y cuenten con un alto nivel de inteligencia.

Por tanto, las características fundamentales que normalmente se persiguen en las aplicaciones empresariales, son:

- Seguridad y fiabilidad.
- Adaptación al tipo y número de usuarios a los que irá dirigida la aplicación.
- Componentes que ofrezcan la posibilidad de reutilización.
- Mantenimiento sencillo.
- Almacenamiento de la información eficaz, y preferiblemente un modelo de datos unificado en una sola base de datos.
- Inteligencia en los procesos de negocio.
- Tiempo mínimo de inactividad de las aplicaciones.
- Alto rendimiento en las aplicaciones en la red (minimizar la latencia de la WAN, la gran carga de los servidores y la congestión de las conexiones con poco ancho de banda)
- Interoperabilidad con otras aplicaciones

La última de las características no es por ello la menos importante. Suele resultar imprescindible que una aplicación empresarial interopere con otros sistemas. De este modo, las aplicaciones no trabajan de forma aislada, sino que colaboran entre ellas. Para que esto sea posible pueden utilizarse servicios Web.

Los servicios Web (Web services) son componentes software que permiten que varias aplicaciones se comuniquen con el fin de intercambiar información. Para llevar a cabo esta tarea se hace uso de protocolos estándares de Internet. En particular se utilizan principalmente tres protocolos:

- WSDL (protocolo XML utilizado para la descripción de los servicios Web)
- SOAP (protocolo basado en XML empleado para comunicar clientes y servidores que exponen servicios Web)
- UDDI (protocolo de descubrimiento de servicios Web)

Organizaciones tales como OASIS (Organization for the Advancement of Structured Information Standards) o W3C (World Wide Web Consortium) son las encargadas de definir la arquitectura y la reglamentación de dichos servicios Web y fomentar su utilización. Otro organismo, el WS-I, se encarga de desarrollar varios perfiles que definan dichos estándares de un modo más concreto.

Entre las características de los servicios Web, se pueden destacar las siguientes:

- Proporcionan interoperabilidad, es decir, varias aplicaciones con propiedades completamente diferentes, desarrolladas mediante lenguajes de programación distintos e independientemente del fabricante y de la plataforma sobre la cual se hayan instalado, pueden comunicarse en una red de ordenadores mediante los servicios Web.
- Además superan las barreras geográficas, ofreciendo la posibilidad de integrar servicios de compañías diferentes y distantes geográficamente.
- Proporcionan una gran flexibilidad de modo que no supone un problema grande que se realicen cambios en las aplicaciones que usan los servicios.
- Se basan en HTTP sobre TCP en el puerto 80. Se permite también que la comunicación sea segura vía SSL sobre HTTPS.
- Permiten a los programadores abstraerse de los procesos de negocio y centrarse en el entorno de programación de los servicios Web.

Sin embargo también soportan algunos inconvenientes:

- Puesto que adoptan estándares y protocolos con formato basado en XML (facilitando la comprensión y el acceso al contenido), ofrecen un bajo rendimiento en comparación con los modelos de computación distribuida. Algunos de ellos son RMI (Java Remote Method Invocation), CORBA (Common Object Request Broker Architecture) o DCOM (Distributed Component Object Model).
- En cuanto a la potencialidad para realizar transacciones (se caracterizan por las tres propiedades que se enumeran: atomicidad, serializables y permanentes), no pueden competir con los estándares abiertos de computación distribuida que hacen uso de múltiples máquinas en un entorno multiusuario. Podemos mencionar CORBA porque proporciona una plataforma para el desarrollo de sistemas distribuidos lo que permite la invocación de métodos remotos.

La automatización del consumo y de la interacción de servicios Web es un campo complejo y necesario para los sistemas B2B actuales. Mediante la especificación sintáctica de los servicios Web, es imposible automatizar procesos tales como descubrimiento, ejecución y composición de servicios. Para solventar este problema, se desarrollaron los servicios Web semánticos, que consisten en la fusión de los servicios Web tradicionales y de las tecnologías empleadas en la Web semántica, que permiten a las máquinas interpretar la información que almacenan, empleando las ontologías como modelo de datos (ver Figura 1-1).

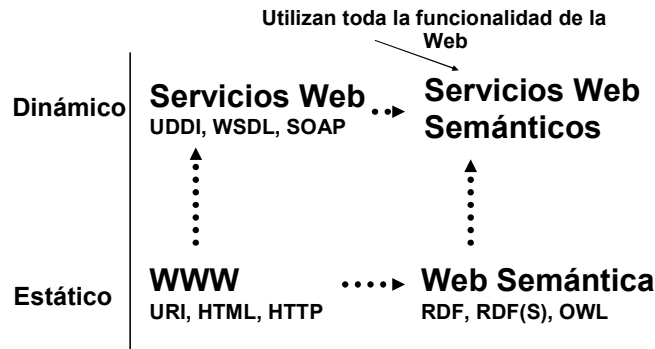


Figura 1-1: Visión de las tecnologías participantes en los SWS

## 1.1 Ontologías

Las ontologías son el concepto núcleo de la Web Semántica, que permite representar la información de manera procesable por un computador.

Según Fensel en [9], las ontologías representan de manera formal y consensuada especificaciones de conceptos, que proveen un conocimiento compartido y común del dominio como información semántica procesable por las máquinas e interoperable a través de agentes (organizaciones, individuos y software).

Una ontología se compone principalmente de los siguientes componentes:

- Conceptos: Elementos básicos de las tareas del dominio. Se suelen organizar en taxonomías.
- Instancias: Son elementos específicos de los conceptos.
- Relaciones: Expresan relaciones entre los conceptos del dominio.
- Funciones: Métodos que pueden ser invocados en la instancia específica de un concepto.
- Axiomas: Sentencias del modelo que son siempre ciertas. Hechos.

El lenguaje de marcado OWL<sup>1</sup>, es la recomendación del W3C como metalenguaje descriptivo de ontologías. Un sencillo ejemplo de ontología en formato OWL podría ser el modelado del concepto de personas relacionadas con la Universidad, formado por la unión de los docentes, estudiantes y el personal PAS:

```
<owl:Class rdf:ID="personasUniversidad">
  <owl:unionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#personalDocente"/>
    <owl:Class rdf:about="#estudiantes"/>
    <owl:Class rdf:about="#PAS"/>
  </owl:unionOf>
</owl:Class>
```

<sup>1</sup> <http://www.w3.org/TR/owl-features/>

## 2 SERVICIOS WEB SEMÁNTICOS

Según [1], la infraestructura de servicio Web semánticos, se puede caracterizar en tres dimensiones: *actividades de uso, arquitectura y ontología del servicio* (véase la Figura 2-1).

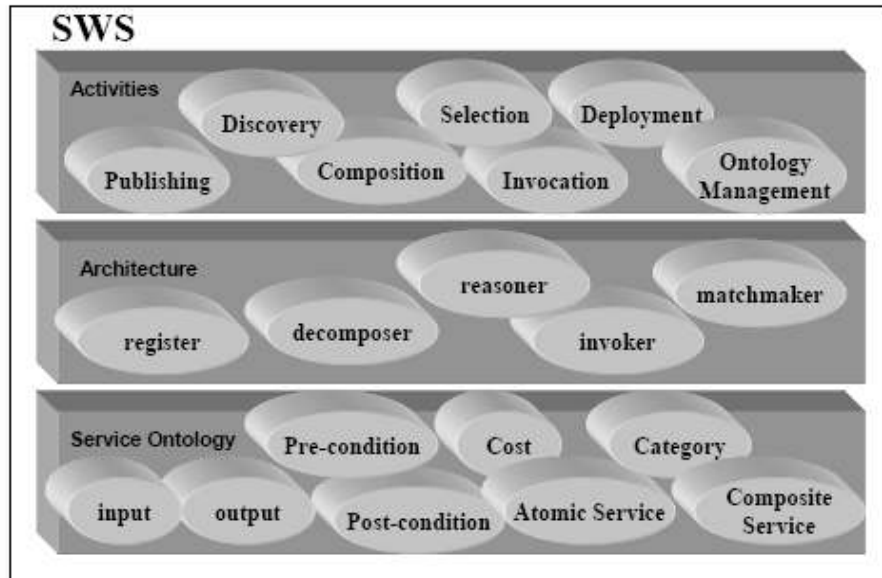


Figura 2-1: Dimensiones de la infraestructura de los SWS

Las actividades de uso del SWS identifican los requisitos funcionales que un framework de servicios Web semánticos debe soportar. Las principales actividades son:

- Publicación de servicios en un registro semántico.
- Descubrimiento automático en función de la petición de servicio y de la descripción semántica publicada del servicio.
- Selección de servicios cuando se pueda producir un conflicto entre varios servicios que satisfagan una petición.
- Composición: Un framework de SWS debe permitir la coreografía de servicios, de forma que se puedan componer los resultados de varios servicios para obtener una funcionalidad de un nivel más alto. Un lenguaje de descripción de procesos, empleado para la composición de servicios, podría ser el BEPL4WS.
- Invocación del SWS: validación de los parámetros de entrada contra la ontología del servicio y ejecución del servicio.
- Despliegue
- Gestión de las ontologías

La arquitectura de un framework de SWS define los componentes necesarios para cumplir las actividades de uso especificadas anteriormente. Estos componentes son:

- Un Registro que ofrezca los mecanismos necesarios para publicar y localizar los SWS.
- Un Razonador para las consultas e interpretación de la descripción semántica de los servicios.
- Matchmaker: Mediador entre el registro y el cliente del servicio en los procesos de descubrimiento y selección de servicios.
- Decomposer: Componente encargado de ejecutar la composición de servicios. Parte de la semántica de la petición realizada, tratando de descomponerla, de forma que se busca la colaboración de las funcionalidades ofrecidas por varios servicios para obtener el resultado deseado.
- Invocador: Mediador entre el cliente del servicio y el proveedor o entre el decompositor y el proveedor cuando se invoca un servicio.

Por último, la ontología del servicio representa las capacidades del servicio y sus restricciones de uso. Integra la semántica del servicio con su descripción. Consta de los siguientes elementos:

- Información funcional del servicio
  - Entradas
  - Salidas
  - Precondiciones
  - Postcondiciones
- Información no funcional
  - Categoría
  - Coste
  - Calidad de servicio
  - ...

En la actualidad se están desarrollando principalmente tres frameworks de SWS: OWL-S, WSMO y SAWSDL.

### 3 OWL-S

OWL-S es una ontología de servicios Web basada en OWL<sup>2</sup>. Añade un marcado de metadatos en los proveedores de los servicios Web, permitiendo la descripción de las propiedades y funcionalidades de cualquier servicio, de forma no ambigua, bien definida y computable.

El inicio de OWL-S<sup>3</sup> viene originado por la evolución de DAML-S, desarrollado por la empresa DARPA (agencia de investigación del departamento de defensa de los Estados Unidos), y que utiliza como metalenguaje, para especificar modelos de conceptos, DAML+OIL, lenguaje de marcado a partir del cual ha surgido el OWL, que es empleado como base para el desarrollo de las ontologías de OWL-S (véase Figura 3-1).

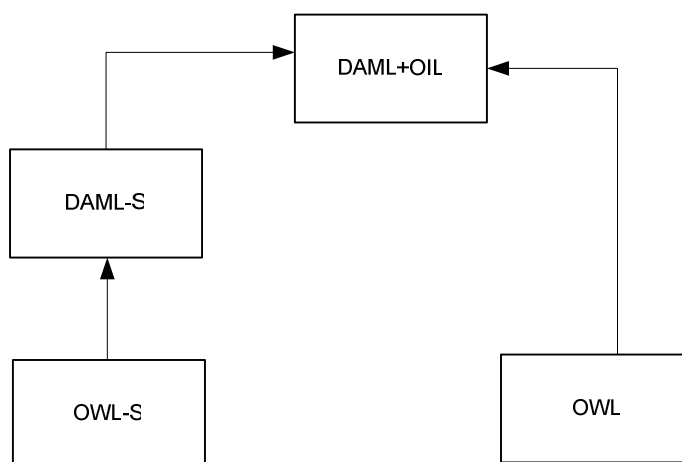


Figura 3-1: Evolución de metalenguajes

La ontología propuesta por OWL-S está formada por tres partes fundamentales (véase la Figura 3-2 obtenida de [8]):

- El *profile* del servicio, empleado para el descubrimiento y selección de servicios.
- *Process model*, ontología encargada de describir con detalle las operaciones procesadas por un servicio.
- El *grounding*, conocimiento base, que describe como interoperar con el servicio, a través de mensajes.

<sup>2</sup> <http://www.w3.org/TR/owl-features/>

<sup>3</sup> <http://www.daml.org/services/owl-s/1.0/owl-s.html>

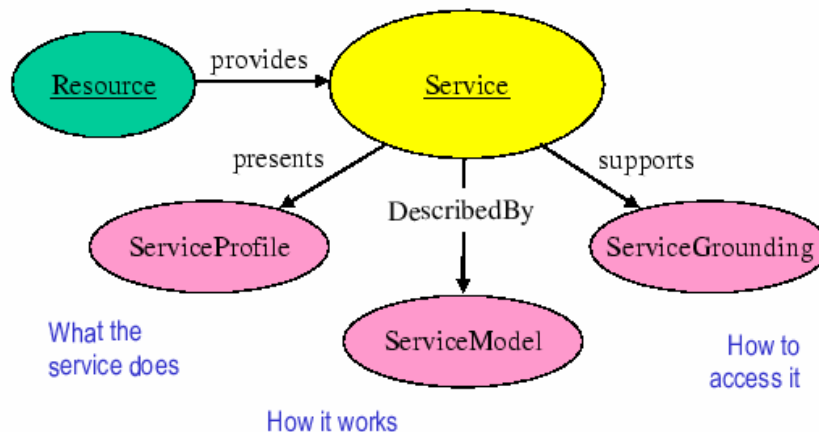


Figura 3-2: Ontologías de OWL-S

Cada servicio posee un conjunto de Service Profiles, de forma que pueda proporcionar diferentes funcionalidades a los clientes. El descubrimiento automático de servicios se hace mediante el matching de los perfiles definidos en el sistema. Cada servicio requiere de un único Service Model, que indica su funcionamiento, y se comunica con el cliente mediante un único Service Grounding.

### 3.1 Service Profile

El perfil de cada servicio consta de los siguientes elementos:

- Información sobre el proveedor del servicio: nombre del servicio, descripción textual breve, información de contacto, ...
- Características del servicio (propiedades): categoría del servicio y parámetros de libre disposición
- Funcionalidad del servicio: Transformación realizada entre los datos de entrada y de salida, precondiciones y efectos del servicio. Es decir, se describe el IOPE (Inputs-Outputs-Preconditions-Effects) del servicio.

### 3.2 Service Model

El servicio se modela como un proceso de interacción entre el cliente y el servicio. Existen tres tipos de proceso:

- Atómico: Formados por IOPE (Entradas, salidas, precondiciones y efectos). Pueden ser ejecutados directamente.
- Compuesto: Tienen asociados una estructura de control (Secuencia, Condición, Iteración, Repetir-Hasta...)
- Simple: Abstracción de procesos atómicos y compuestos.

### 3.3 Service Grounding

El conocimiento base (grounding) especifica cómo acceder e invocar al servicio. Realiza un descripción de las relaciones existentes entre la descripción del proceso realizado por el servicio y la descripción real de los elementos necesarios para interactuar con el servicio (como los protocolos de comunicaciones).

### 3.4 Herramientas desarrolladas para OWL-S

Actualmente, el estado de desarrollo de entornos basados en OWL-S es escaso. Algunas herramientas interesantes son:

- OWL-S Editor, desarrollado por la Universidad de Malta.
- OWL-S Java API, desarrollada por Mindswap, para ejecutar servicios OWL-S.
- OWL-S Editor basado en Protegé.

## 4 WSMO

WSMO es una ontología y un modelo conceptual para describir Servicios Web Semánticos, desarrollada por un conjunto de empresas dentro del SDK-Cluster, coordinadas por DERI<sup>4</sup>. Está basada en el WSMF (Web Service Modeling Framework), y en realidad consta de varias partes, tal y como se muestra en la Figura 4-1.

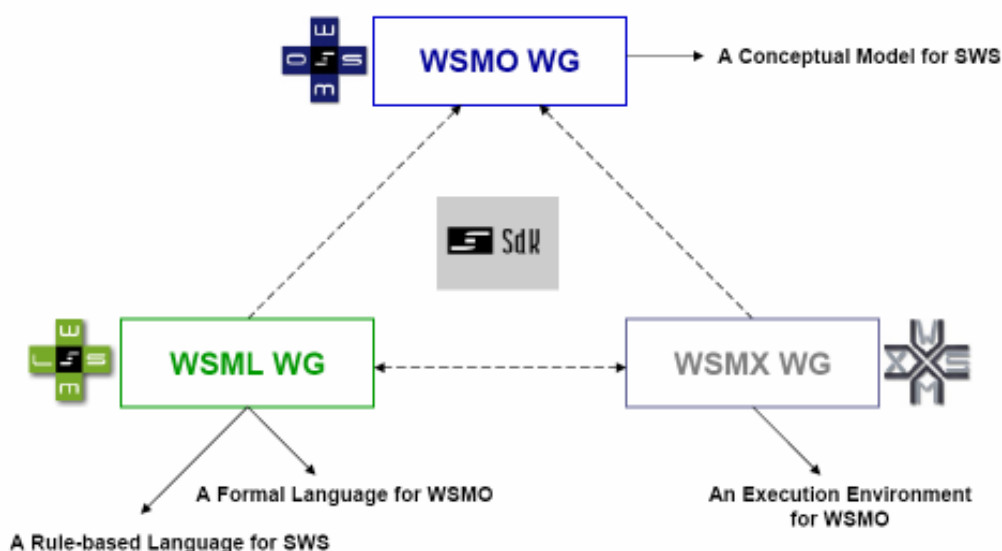


Figura 4-1: Visión general

La idea principal de WSMO es la de desarrollar un framework de SWS que permita resolver todos los problemas posibles en la integración de los SWS y de generar una implementación del mismo, capaz de servir como referencia. Todos estos aspectos se han de basar en dos principios:

- Desacomplamiento: Las aplicaciones han de ser lo más independientes posibles.
- Comunicaciones escalables de mediación, permitiendo que todo el mundo pueda

<sup>4</sup> <http://www.deri.org>

intercomunicarse con todo el mundo.

El proyecto WSMO se divide en tres partes: WSMO, WSML y WSMX.

## 4.1 WSMO

La ontología desarrollada como WSMO, expresada en WSML y publicada en el W3C como Member Submission, se basa en la utilización de cuatro componentes, tal como muestra la Figura 4-1.1:

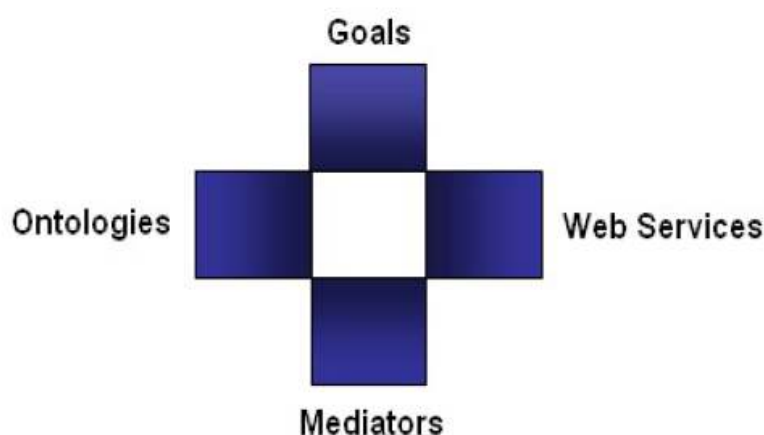


Figura 4-1-1: Componentes de WSMO

- Las ontologías, proporcionan la especificación formal de la información usada por todos los componentes.
- Los objetivos deseados por los clientes cuando invocan un servicio Web, describen cual es el problema a resolver.
- La semántica de los servicios Web, describen su interfaz de uso (utilizada para coreografía y orquestación de servicios) y sus funcionalidades (definidas mediante IOPE). También se especifican las propiedades no funcionales de los servicios, por ejemplo utilizado Dublin Core.
- Por último, los mediadores, son conectores entre componentes, permitiendo fácilmente la integración entre información heterogénea. Resuelven problemas de interoperabilidad y pueden ser de cuatro tipos:
  - Mediadores entre ontologías (OOMediators): Resuelven incompatibilidades.
  - Mediadores entre objetivos (GGMediators): Buscan el refinamiento y la compatibilidad entre objetivos.
  - Mediadores entre un servicio Web y un objetivo (WGMediator): Busca la satisfacción de las necesidades.
  - Mediadores entre servicios Web (WWMediator): Utilizado en composición de servicios.

## 4.2 WSML

El Web Service Modeling Language, es un nuevo lenguaje de modelado de ontologías desarrollado para modelar WSMO. Una de sus características principales es que no es un formato XML, aunque se puede mapear a XML mediante una especificación. Sus creadores decidieron desarrollar un formato plano en lenguaje natural, legible para los humanos, debido a su uso más sencillo para el modelado.



WSML es una familia de lenguajes, que se pueden dividir en las siguientes cinco categorías<sup>5</sup>:

- WMSL-Core: Se encuentra en la intersección entre la lógica descriptiva y la lógica de Horn. La semántica se define de forma similar a OWL-Lite, y tiene una sintaxis basada en marcos. Es la más sencilla.
- WMSL-Flight: Basada en OWL-Flight. Además de las capacidades de WMSL-Core, añade restricciones y características no monótonas. Es el lenguaje preferido para modelar ontologías.
- WMSL-Rule: Extiende WSL-Flight y se basa en la programación lógica. Es el preferido para modelar objetivos y servicios Web.
- WMSL-DL: Extiende WSML-Core con lógica descriptiva.
- WSML-Full: Unifica todas las variantes de WMSL bajo lógica de primer orden con extensiones no monótonas, lo que permite capturar la negación no monótona de WMSL-Rule.

### 4.3 WSMX

WSMX es una arquitectura orientada a servicios (SOA) basada en eventos, que sirve como entorno de ejecución para Servicios Web Semánticos. Es la implementación de referencia para WSMO.

Se encuentra en desarrollo y está publicada como software libre en Sourceforge<sup>6</sup>.

## 5 SAWSDL

SAWSDL<sup>7</sup> es una extensión, desarrollada por un grupo de trabajo del W3C, del lenguaje de descripción de servicios Web (WSDL). Está constituido por dos tipos básicos de anotaciones: la referencia al modelo y el mapeo del esquema.

Las anotaciones de referencia al modelo son las mismas que las empleadas en el modelo WSDL-S<sup>8</sup>, se utilizan para asociar interfaces, puertos, operaciones, entradas, salidas, elementos xml y atributos con conceptos semánticos.

En cuanto al mapeo del esquema, son atributos añadidos a la declaración del esquema XML de los elementos, para especificar mapeos entre la información semántica y XML. Se utilizan durante la invocación de los servicios, para formatear la información del cliente a XML, de forma que pueda ser entendida por el servicio Web, solucionándose el problema de la estructura de las entradas y salidas del servicio. Normalmente suele ir asociado a una hoja de transformación XSLT.

A diferencia de WSDL-S, permite especificar semánticamente el comportamiento de los servicios Web, mediante la utilización de lenguajes de descripción de ontologías como OWL, siendo útil para la realización de coreografías de servicios.

Permite el desarrollo de las siguientes funcionalidades:

- Clasificación de servicios
- Descubrimiento
- Matching
- Composición
- Invocación dinámica

---

<sup>5</sup> <http://www.wsmo.org/TR/d16/d16.1/v0.21/>

<sup>6</sup> <http://sourceforge.net/projects/wsmx/>

<sup>7</sup> SAWSDL: Anotaciones semánticas para WSDL

<sup>8</sup> <http://www.w3.org/Submission/WSDL-S/>

## 6 CONCLUSIONES

Con la aparición de la Web Semántica y el auge de las arquitecturas SOA (orientadas a servicios), aparecen los servicios Web Semánticos como tecnología de futuro. En la Web extendida, la información se encuentra bien definida, de forma que puede ser interpretada por las máquinas, permitiendo la automatización de tareas que en la actualidad solamente pueden realizarse en tiempo de compilación, como son la composición, el descubrimiento dinámico o la ejecución de servicios.

Actualmente existen diversos proyectos sobre SWS en fase de desarrollo, sin embargo, al ser una tecnología incipiente, nos encontramos en un punto inicial de la investigación en este campo, donde hay mucha formalización, pero pocas (o casi ninguna) plataforma capaz de ofrecer todas las funcionalidades requeridas por los servicios Web Semánticos.

En el futuro, la línea más optimista de trabajo en este campo, parece ser el enfoque realizado por SAWSDL, donde se añaden características semánticas al lenguaje de descripción de los servicios, permitiendo la rápida introducción de estos servicios en el mundo Web.

## BIBLIOGRAFÍA

- [1] L. Cabral, J. Domingue, E. Motta. *Approaches to Semantic Web Services: An Overview and Comparisons*. Proceedings First European Semantic Web Symposium (ESWS2004) The SemanticWeb: Research and Applications, Lecture Notes in Computer Science 3053(LNCS3053), pages pp. 225-239, Heraklion, Crete, Greece
- [2] D. Berrueta. *Una perspectiva de los Servicios Web Semánticos*. Abril 2006
- [3] D. Fensel, C. Bussler. *The Web Service Modeling framework WSMF*. Electronic Commerce Research and Applications, 1(2), 2002
- [4] E. Sirin, J. Hendler, B. Parsia. *Semi-automatic Composition of Web Services using Semantic Descriptions*.
- [5] Terry Payne. *Semantic Web Services*. Publicado en el IEEE Intelligent Systems, en 2004
- [6] Sheila A. McIlraith, Tran Cao Son, Honglei Zeng. *Semantic Web Services*. Universidad de Stanford. Publicado en el IEEE. Año 2001
- [7] Aaron Swartz. *MusicBrainz: A Semantic Web Service*. Publicado en el IEEE. Año 2002
- [8] The OWL Services Coalition. *OWL-S: Semantic Markup for Web Services*
- [9] D. Fensel. *Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce*, Springer-Verlag, Berlin, 2001. 2nd Edition, Springer-Verlag, Berlin, 2003
- [10] C. Bussler, S. Arroyo, M. Stollberg. *The Web Service Modeling Ontology – WSMO*. NetObjectDays2004. 27 de Septiembre del 2004.