

Universidad Autónoma Metropolitana Unidad Azcapotzalco

División de Ciencias Básicas e Ingeniería

Licenciatura en Ingeniería en Computación

Propuesta de Proyecto Terminal

Extracción automatizada y representación de servicios Web mediante ontologías

Jorge Pascual Martínez 207202192

Trimestre 11-O

Fecha de entrega: 11 de noviembre del 2011

Versión: Primera

Asesor: Dra. Maricela Claudia Bravo Contreras

Profesor titular C

Departamento de Sistemas

Objetivo General

- Diseñar e implementar un sistema que permita extraer los elementos más relevantes de múltiples descripciones de servicios Web y que los represente mediante un modelo ontológico.

Objetivos Específicos

- Diseñar un modelo ontológico para la representación de los servicios Web
- Diseñar e implementar un modulo que ofrezca una interfaz de conectividad entre el sistema y el modelo ontológico.
- Diseñar e implementar el modulo de conversión de WSDL¹ 1.0[1] a OWL²[2]
- Diseñar e implementar el modulo de conversión de WSDL 2.0[3] a OWL
- Diseñar e implementar el modulo de conversión de OWL-S³[4] a OWL

Introducción

Para describir la interfaz pública de los servicios Web se utilizan distintos lenguajes de descripción de servicios (SDL⁴), lo que ocasiona una gran dificultad para la búsqueda e invocación automatizada de servicios Web. Estos SDLs utilizan una estructura distinta, sin embargo, contienen elementos comunes, los cuales proporcionan la interfaz de comunicación que utiliza el cliente para crear un proxy que invoque al servicio de forma remota.

Un estudio realizado a los siguientes SDLs: WSDL, OWL-S y WSML⁵, de acuerdo con [5], se obtuvo como resultado que la descripción funcional (*functional description*) de los servicios Web se dividen en la transformación de la información (*information transformation*) realizada por el servicio y el cambio de estado como consecuencia de la ejecución del servicio. La transformación de la información se correlaciona con los parámetros de entrada y de salida, mientras que el cambio de estado (*state change*) describe una condición de modificación después de la ejecución del servicio.

¹ WSDL *Web Services Description Language*.

² OWL *Web Ontology Language*.

³ OWL-S *Semantic Markup for Web Services*

⁴ SDL *Service Description Language*.

⁵ WSML *Web Services Modeling Language*

La tabla 1 muestra la comparación de la descripción funcional entre los SDLs analizados

		WSDL	OWL-S	WSML
<i>Functional Description</i>		Operation	Profile: serviceName	Capability
<i>Information transformation</i>	Input	Input	Profile: hasInput	Capability: Preconditions
	Output	Output	Profile: hasOutput	Capability: Postconditions
<i>State Change</i>	Previous state		Profile: hasPrecondition	Capability: Assumptions
	Posterior state		Profile: hasResult	Capability: Effects

Tabla 1

Usando la correlación que se describe en la tabla 1, entre estos modelos de descripción de servicios, es posible definir un modelo general para describir a los servicios Web.

Este modelo general será representado mediante el diseño de una ontología [6], se utilizara OWL como lenguaje para definir el modelo ontológico, la motivación acerca del uso de OWL es por ser una recomendación [7] de la W3C⁶ y por ser actualmente el lenguaje más popular para la creación de ontologías.

Esta propuesta toma como punto de partida los resultados expuestos anteriormente e implementara una aplicación que dado un archivo de descripción de un servicio Web escrito en WSDL (versión 1.0 y 2.0) y OWL-S lo transforme a su correspondiente representación en OWL.

Justificación

Los desarrolladores de software a menudo buscan en los repositorios de servicios Web públicos, aquellos que cumplan con ciertos requerimientos para integrarlos a sus aplicaciones. Sin embargo, la implementación de las interconexiones de los servicios Web debe realizarse de forma manual o semiautomática.

Además la falta de una buena documentación de los servicios, hace más complicado su invocación, para mejorar la documentación se han propuestos diferentes soluciones. Una de ellas se basa en la Web semántica y el uso de Ontologías.

⁶ W3C World Wide Web Consortium

Los beneficios de utilizar una ontología como modelo de representación son los siguientes:

- Definen de forma estándar los términos y sus relaciones dentro de un determinado dominio (área de conocimiento)
- Permiten el uso de restricciones y reglas de inferencia para hacer entender a las máquinas los conceptos que se manejan en ese dominio.
- Provee una forma para codificar el conocimiento y la semántica de tal manera que las máquinas puedan entenderlas.

Derivado de estos problemas la relevancia de este proyecto es que colabora en la solución de invocar a los servicios Web de forma automatizada, ya que al finalizar el proyecto se obtendrá una aplicación que permita, a partir de distintos archivos de descripción escritos en diferentes SDLs, encontrar los elementos más relevantes y llevarlos a un modelo ontológico general, a partir del cual se podrá generar el proxy cliente para invocar al servicio de forma remota.

Al usar un modelo ontológico se logra documentar de una mejor manera la descripción de los servicios ya que se asocian conceptos y relaciones del modelo ontológico con parámetros y operaciones de un servicio Web.

Antecedentes

Antecedentes internos:

Se encontró una propuesta de proyecto terminal el cual trabaja con servicios Web, este proyecto aun no está concluido.

Propuestas de proyecto terminal	Similitudes	Diferencias
Plataforma de archivos de acceso remoto mediante servicios Web[8]	El trabajo se fundamenta en servicios Web	No es una aplicación que pretenda resolver los problemas relacionados con la invocación de servicios Web.

Antecedentes externos:

Artículo de revista.

Titulo	Similitudes	Diferencias
"On the functional quality of service (FQoS) to discover and compose interoperable Web services[5]	Estudio de similitudes y comparación de SDLs. Planteamiento de un modelo general.	1.-Diseño del modelo general 2.-Implementación en java de la aplicación que permita representar a un servicio Web en base al modelo diseñado.

Documento consultado en línea

A Conceptual Comparison between WSMO and OWL-S[9]	Analiza y compara las similitudes de dos SDLs, plantea la correlación que existe entre las ellos.	1.- Implementación en java de la aplicación. 2.- Soporte para OWL-S, WSDL 1.0 y 2.0
---	---	--

Descripción técnica

Este proyecto tiene la finalidad de construir un sistema que permita la conversión de un archivo escrito en cualquiera de los siguientes SDLs: WSDL, OWL-S y WSML, hacia una representación ontológica descrita en OWL, con el objetivo de que al poder tener una representación general de los archivos de descripción de los servicios Web se tenga la posibilidad de construir automáticamente un cliente para la invocación de los servicios de forma remota.

El proyecto está conformado por los módulos descritos a continuación:

- Módulo WSDL1.0: Este elemento de la aplicación será el encargado encontrar los elementos más relevantes de los archivos de descripción de servicios escritos en WSDL 1.0, extraerlos y crear a partir de estos el archivo de descripción en OWL, correspondiente con el modelo ontológico elaborado.
- Módulo WSDL 2.0: Este componente recupera toda la información requerida para la creación de la representación ontológica, a partir de un archivo con formato WSDL 2.0 de de tal manera que se construya un archivo de descripción en OWL.
- Módulo OWL-S: Éste modulo tomara como origen un archivo escrito en OWL-S y lo llevara a su correspondiente OWL de acuerdo con el modelo ontológico, recuperando todos los elementos necesarios.

- **Módulo de conectividad:** Este módulo es la encargada de ofrecer la conectividad entre el sistema y el modelo ontológico, es la que permite crear un repositorio de archivos de descripción en OWL.

La ilustración 1 muestra los módulos que conforman la aplicación y el repositorio del modelo ontológico.

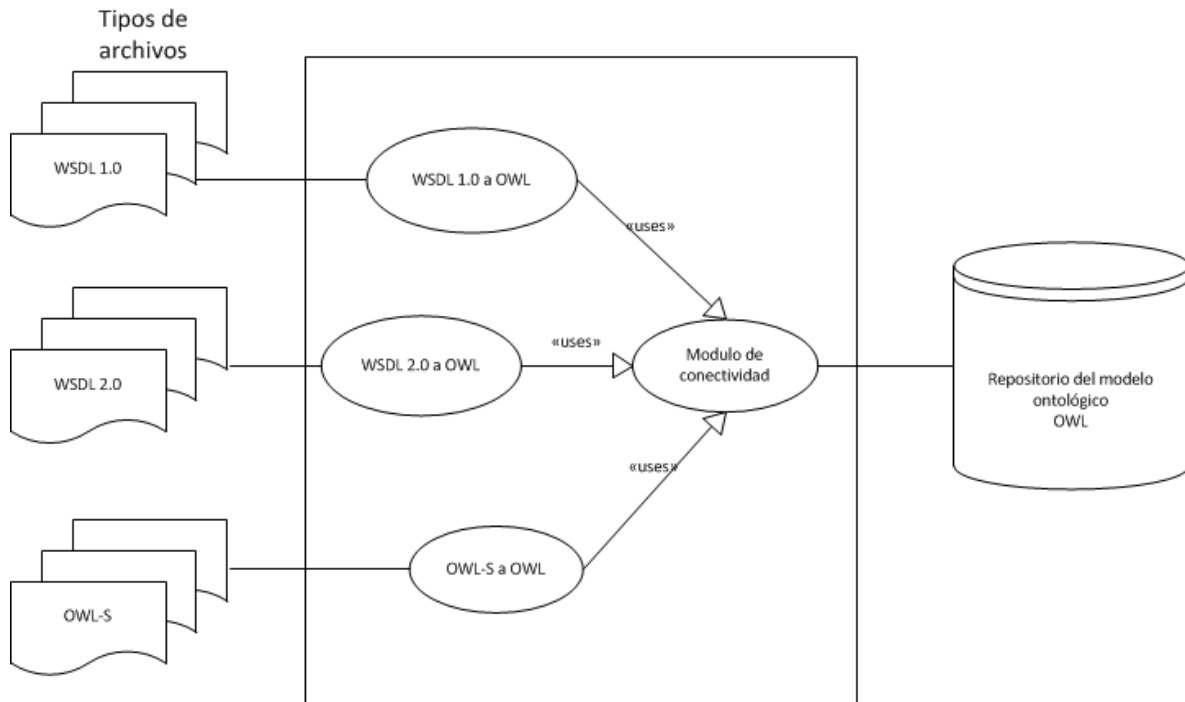


Ilustración 1

Especificación técnica

El presente proyecto plantea realizar una aplicación que una vez esté terminado, pueda ser usado como un API⁷ de java y así integrarse a una aplicación de mayor envergadura.

Herramientas de desarrollo

1. La aplicación será programada en lenguaje java.
2. Se utilizara el entorno de desarrollo integrado Eclipse Enterprise.

Este proyecto se dará por concluido cuando el sistema permita la conversión de diversos archivos escritos en WSDL (1.0 y 2.0) y OWL-S y los transforme a su correspondiente archivo de descripción OWL.

Además deberán de ser probados con un mínimo de 200 archivos de los tres diferentes SDLs que se usan, con lo que se generará la población automática de la ontología. La cual funcionara como repositorio para poder guardar y posteriormente consultar los archivos de descripción con el formato OWL.

Como continuación del proyecto se sugiere incluir más módulos de conversión entre SDLs que no se consideraron aquí, pero que son utilizados comúnmente para la representación de los servicios Web. Como pueden ser SAWSDL⁸, WSMO⁹ etc.

Los entregables del proyecto son los que se listan a continuación:

- Cd con todos los componentes para hacer funcional el proyecto.
- Manual de instalación
- Código fuente documentado
- Repositorio generado por el programa (Poblado de la ontología)
- Repositorio de archivos de origen (Archivos con extensión WSDL,OWL-S)

⁷ API Application Programming Interface

⁸ SAWSDL Semantic Annotations for WSDL

⁹ WSMO Web Service Modeling Ontology

Calendario de trabajo

El desarrollo del proyecto tendrá una duración de dos trimestres, el proyecto terminal I en el trimestre 12-I y el proyecto terminal II en el trimestre 12-P, las tablas 2 y 3 muestran las actividades a realizar en cada trimestre.

Calendario de actividades para el proyecto terminal I correspondiente al trimestre 12-I (99 hrs)

Semana Actividades	1	2	3	4	5	6	7	8	9	10	11	Horas a la semana
Diseño del modelo ontológico	■	■	■									27
Diseño del modulo WSDL 1.0 a OWL				■	■							18
Diseño del modulo WSDL 2.0 a OWL						■	■					18
Diseño del modulo OWL-S a OWL								■	■			18
Diseño del modulo de conectividad										■	■	18
Documentación	■	■	■	■	■	■	■	■	■	■	■	
	Total											99

Tabla 2

Calendario de actividades para el proyecto terminal II correspondiente al trimestre 12-P (198 hrs)

Semana Actividades	1	2	3	4	5	6	7	8	9	10	11	Horas a la semana
Implementación del modulo WSDL 1.0 a OWL	■	■	■									45
Implementación del modulo WSDL 2.0 a OWL			■	■	■							45
Implementación del modulo OWL-S a OWL						■	■	■				54
Implementación del modulo de conectividad									■	■	■	54
Pruebas a cada modulo		■	■	■	■	■	■	■	■	■	■	
Documentación	■	■	■	■	■	■	■	■	■	■	■	
	Total											198

Tabla 3

Recursos

El alumno cuenta con los siguientes recursos:

- Computadora Acer con procesador Intel Pentium DualCore E5400, 2 GB de memoria RAM, disco duro de 320 GB.
- Con sistemas operativos Linux Debian 6.0 y Windows 7 professional.

Estos recursos son suficientes para concluir el proyecto en tiempo y forma

Bibliografía

[1] E. Christensen et al. (2011, Noviembre 4). *Web Services Description Language*. [En línea]. Disponible: <http://xml.coverpages.org/wsdl20000929.html>

[2] D. McGuinness, F. van Harmelen.(2011,Octubre 26) *OWL Web Ontology Language*. [En línea]. <http://www.w3.org/TR/owl-features/>

[3] D. Booth. (2011, Noviembre 5). *Web Services Description Language (WSDL) Version 2.0*. [En línea]. <http://www.w3.org/TR/wsdl20-primer/>

[4] D. Martin. (2011, Noviembre 5). *OWL-S Semantic Markup for Web Services*. [En línea]. <http://www.w3.org/Submission/OWL-S/>

[5] B. Jeong, H. Cho, C. Lee, "On the functional quality of service (FQoS) to discover and compose interoperable Web services", *International Journal of Expert Systems with Applications*, vol. 33, no 3, pp 5411-5418, Abril 2009

[6] J. Heflin. (2011, Noviembre 6). *OWL Web Ontology Language Use Cases and Requirements*. [En línea]. <http://www.w3.org/TR/2004/REC-webont-req-20040210/>

[7] S. Benchhofer et al.(2011, Noviembre 4). *OWL Web Ontology Language Reference*. [En línea]. Disponible: <http://www.w3.org/TR/owl-ref/>

[8] M.M Ramírez, "Plataforma de archivos de acceso remoto mediante servicios Web", Propuesta de proyecto terminal, Universidad Autónoma Metropolitana Azcapotzalco, D.F., México, 2008

[9] R. Lara et al. (2011, Octubre 20). *A Conceptual Comparison between WSMO and OWL-S*. [En línea]. Disponible: http://www.wsmo.org/2004/d4/d4.1/v0.1/20050106/d4.1v0.1_20050106.pdf