

**OASEARCH: Modelo de aplicación
basado en web semántica para la
búsqueda de objetos de aprendizaje
mediante perfilado de consultas.**

Ing. DIEGO FERNANDO GÓMEZ ÁLVAREZ

**Trabajo de grado presentado como requisito parcial para
optar al título de Magister en Ingeniería.**

Asesor: Dr. JUAN GUILLERMO LALINDE

**MEDELLIN
UNIVERSIDAD EAFIT
ESCUELA DE INGENIERÍA
2014**

Nota de aceptación

Presidente del jurado

Jurado

Jurado

Medellín, Octubre 15 de 2014

A mis padres Javier y Beatriz que siempre estuvieron pendientes de lo que transcurriera con este trabajo.

A mis hermanos Catalina y Carlos Mario que con sus consejos hicieron que muchas cosas fueran posibles.

A mi novia Verónica que con su apoyo y comprensión logró que todo este proceso fuera más fácil.

AGRADECIMIENTOS

A la Línea de Investigación en Informática Educativa de la Universidad EAFIT por facilitar espacios para finalizar este trabajo.

A mis amigos y compañeros de Medellín Ciudad Inteligente que con sus consejos y apoyo permitieron que trabajar con ellos se convirtiera en la mejor excusa para dedicarme a este proyecto.

This work was conducted using the Protégé resource, which is supported by grant GM10331601 from the National Institute of General Medical Sciences of the United States National Institutes of Health.

OASEARCH: Modelo de aplicación basado en web semántica para la búsqueda de objetos de aprendizaje mediante perfilado de consultas.

Tesis Maestría en Ingeniería

Autor: Diego Fernando Gómez Álvarez dgomeza2@eafit.edu.co

Asesor: Juan Guillermo Lalinde jlalinde@eafit.edu.co

UNIVERSIDAD EAFIT

RESUMEN

Uno de los aspectos que determina la calidad de un Sistema Educativo Basado en la Web es la manera en que este proporciona a los estudiantes los objetos de aprendizaje que mejor se adaptan a sus condiciones específicas. Es por lo anterior que se hace necesario contar con mecanismos de búsqueda para estos objetos que puedan ser usados fácilmente por la aplicación o por los usuarios de la misma y que entreguen los resultados más relevantes posibles en un contexto determinado.

Múltiples mecanismos se han propuesto para la búsqueda de recursos digitales en la web. El enfoque fundamental de recuperación de la información ha sido utilizado por muchos años con bastante éxito pero con el advenimiento de las tecnologías que componen lo que se conoce como La Web Semántica es posible definir nuevas representaciones para los datos que permiten obtener una mayor precisión y relevancia de los resultados obtenidos.

El presente documento presenta un nuevo enfoque para la construcción de motores de búsqueda para objetos de aprendizaje basado en La Web Semántica que se apoya en la definición previa o perfilado de consultas de una manera desacoplada de la aplicación. Con este enfoque es posible construir Conceptos de Búsqueda que son especificados en una Ontología de Dominio de tal manera que luego puede ser utilizada por la aplicación para entregar sus resultados en un entorno distribuido de repositorios RDF que se basen en el estándar LOM para la descripción de metadatos.

PALABRAS CLAVE

Web Semántica, Objetos de Aprendizaje, Ontologías, Sistemas Educativos Basados en la Web, Recuperación de la Información, Descubrimiento de Información, LOM, Metadatos, OWL, RDF

Tabla de contenidos

| | |
|--|-----------|
| INTRODUCCIÓN..... | 1 |
| 1. OBJETIVOS..... | 4 |
| 1.1. Objetivo General: | 4 |
| 1.2. Objetivos Específicos:..... | 4 |
| 2. ALCANCE Y PRODUCTOS | 5 |
| 3. METODOLOGÍA..... | 6 |
| 4. MARCO TEÓRICO | 7 |
| 4.1. La web semántica y el descubrimiento de información. | 7 |
| 4.1.1. <i>Modelado de información</i> | 9 |
| 4.1.2. <i>Expresando la semántica</i> | 12 |
| 4.1.3. <i>Implementando aplicaciones para la web semántica</i> | 15 |
| 4.1.4. <i>Descubrimiento de información.</i> | 19 |
| 4.2. Los objetos de aprendizaje y metadatos | 20 |
| 4.2.1. <i>Los objetos de aprendizaje</i> | 20 |
| 4.2.2. <i>Metadatos para los objetos de aprendizaje.</i> | 20 |
| 4.3. Sistemas educativos basados en la web | 22 |
| 5. ESTADO DEL ARTE..... | 24 |
| 5.1. Introducción..... | 24 |
| 5.2. Web semántica y sistemas educativos | 24 |
| 5.3. Los objetos de aprendizaje y la Web Semántica..... | 27 |
| 5.4. Búsqueda de objetos de aprendizaje en una arquitectura de Web Semántica. | 30 |
| 6. PROPUESTA DE MODELO PARA LA BÚSQUEDA DE OBJETOS DE APRENDIZAJE | |
| 33 | |
| 6.1. Análisis del problema y propuesta de solución. | 33 |
| 6.2. Descripción general para el prototipo de la aplicación OASearch..... | 34 |
| 6.3. El modelo de datos..... | 36 |
| 6.3.1. <i>Seleccionando la Ontología para expresar los metadatos LOM</i> | 36 |
| 6.3.2. <i>Definiendo el patrón de expresión para el tipo de búsqueda.</i> | 37 |
| 6.3.3. <i>Construcción de la ontología de dominio para el patrón de expresión.</i> | 39 |
| 6.4. Construcción de la aplicación. | 45 |
| 6.4.1. <i>Descripción general de la aplicación</i> | 45 |
| 6.4.2. <i>Aspectos fundamentales de codificación de la aplicación.</i> | 46 |
| 7. CONCLUSIONES Y TRABAJO FUTURO..... | 51 |
| BIBLIOGRAFÍA | 53 |

Índice de ilustraciones

| | |
|---|----|
| Ilustración 1 Diagrama de capas de la Web Semántica..... | 9 |
| Ilustración 2 Ejemplo de grafo para representación de recursos..... | 11 |
| Ilustración 3 Interfaz inicial de OASearch..... | 34 |
| Ilustración 4 Resultados de una búsqueda en OASearch..... | 35 |
| Ilustración 5 Aplicación OASearch. Entradas y salidas de información. | 35 |
| Ilustración 6 Descripción general de la ontología lom2owl. | 36 |
| Ilustración 7 Ontología del modelo de la búsqueda | 39 |
| Ilustración 8 Ontología del modelo del estudiante | 40 |
| Ilustración 9 Ontología integrada del modelo del estudiante y la búsqueda | 40 |
| Ilustración 10 Definición en Protege de la clase no nombrada para el dominio de la relación belongsToSConcept..... | 41 |
| Ilustración 11 Ontología para el modelo del objeto de aprendizaje derivada de lom2owl | 42 |
| Ilustración 12 Ontología del modelo de la matriz | 42 |
| Ilustración 13 Ontología con todos los modelos integrados | 43 |
| Ilustración 14 Ontología integrada con el modelo de pesaje | 44 |
| Ilustración 15 Etapas de ejecución de la aplicación OASearch. | 45 |

Índice de tablas

| | |
|--|----|
| Tabla 1 Ejemplo de código en RDF/XML | 12 |
| Tabla 2 Relación entre el tipo de recurso LOM y las dimensiones de estilos de aprendizaje..... | 38 |

INTRODUCCIÓN

En el contexto de la informática aplicada a la educación sobresale el uso de los Objetos de Aprendizaje que se definen como recursos digitales que se pueden reutilizar en diferentes contextos para lograr un objetivo de aprendizaje particular [1].

Para especificar de una manera precisa las características y el objetivo de aprendizaje de estos objetos se utiliza la descripción a través de metadatos. Los metadatos que describen un objeto de aprendizaje son estandarizados en especificaciones como LOM (Learning Object Model) [2].

Los Metadatos que están asociados a los objetos de aprendizaje entregan información importante acerca de los mismos y permiten que estos sean recuperados con cierta facilidad desde los diferentes repositorios en que se encuentren ubicados siempre y cuando se conozca con certeza las características que el objeto buscado debe cumplir.

Como sucede con todos los recursos digitales publicados en la red Internet o en repositorios más o menos extensos, se hace necesario contar con mecanismos que faciliten que estos sean encontrados y que satisfagan las características que el usuario plantea a través de una consulta. Las consultas pueden hacerse sobre un metadato específico (Por ejemplo, todos los objetos de aprendizaje que en el título tengan la palabra planta) o pueden hacerse sobre varios metadatos combinados (Por ejemplo, todos los objetos de aprendizaje que en el título tengan la palabra planta, que tengan algún video y que correspondan al tema Ingeniería Industrial). En el primer caso, se obtendrá un conjunto amplio de resultados que satisfacen la consulta. En el segundo los resultados serán más restringidos en la medida que se ha especificado en mayor detalle la consulta utilizando varios metadatos.

En cualquiera de los casos anteriores, el usuario, al formular su consulta puede tener una idea más o menos clara del objeto de aprendizaje que busca y tratará de sintetizar esa idea en una serie de palabras clave u opciones de metadato específicas. Pero ¿Qué pasa cuando el usuario tiene una idea de lo que está buscando, pero no sabe a qué metadatos específicos corresponde o no le interesa saberlo? En este caso es posible proporcionarle al usuario un mecanismo sencillo mediante el cual solo seleccione el concepto que está buscando sin que tenga que trasladarlo a una combinación de metadatos determinada. Por ejemplo, sería trivial desarrollar un sistema de información en el cual, si un usuario está buscando objetos de aprendizaje que correspondan al concepto "Apto para estudiantes con déficit de atención" este pueda simplemente seleccionar este concepto de una lista en vez de seleccionar una combinación de metadatos como (Nivel de Interactividad=Alto, Tipo de recurso=Video, Duración del video < 1 minuto). Resumidamente, es posible entregar al sistema de

2 | OASEARCH: Modelo de aplicación basado en web semántica para la búsqueda de objetos de aprendizaje mediante perfilado de consultas.

información los datos que definen estos conceptos de búsqueda para que, una vez seleccionado el concepto, devuelva un conjunto de resultados relevantes. Sin embargo, todo esto implicaría que el administrador de la aplicación estuviera constantemente creando consultas predefinidas por cada concepto imaginable lo cual sería inviable desde el punto de vista del mantenimiento de la aplicación.

Sin embargo, si existiera un mecanismo que desacople la definición de las consultas de la aplicación en sí, sería posible entregarle esta labor a un conjunto de usuarios más amplio incrementando así la cantidad de consultas predefinidas susceptibles de ser creadas. De esta forma, el usuario que llega buscando objetos de aprendizaje tendría una oferta amplia de conceptos. En definitiva, la búsqueda la haría sobre conceptos ya elaborados y no sobre datos atómicos disgregados.

Este documento busca describir el trabajo realizado para definir un modelo de aplicación informática que demuestra la factibilidad de crear, de una manera desacoplada de la aplicación, consultas predefinidas que obedecen a conceptos específicos de búsqueda que luego el usuario puede utilizar para recuperar objetos de aprendizaje relevantes.

El modelo mencionado utiliza las tecnologías de la web semántica para desarrollar su propósito. Para lograr representar los conceptos de búsqueda se recurrió al uso de Ontologías, específicamente se utilizó la tecnología RDF/OWL para expresar el estándar LOM de metadatos para Objetos de Aprendizaje como una Ontología.

El uso de RDF y OWL para representar los datos de los objetos de aprendizaje así como los conceptos de búsqueda hace posible desacoplar la lógica de la aplicación del modelo de datos y permite obtener un mecanismo de búsqueda más conciso y útil desde el punto de vista del usuario.

A través de las siguientes secciones se mostrará cómo se llegó al resultado mencionado. En el marco teórico (sección 4.1) se hará una descripción de la web Semántica y cómo esta ofrece mecanismos que permiten no solo almacenar información sino también darle sentido (semántica) a un conjunto de datos. En la sección 4.2 se revisará el concepto de Objeto de Aprendizaje y la manera en que son usados los metadatos para describirlo. Luego, en la misma sección, se mostrará cómo es posible utilizar la web semántica y específicamente RDF-OWL para expresar los metadatos de un Objeto de Aprendizaje determinado. Así mismo, en la sección 4.3, se presentan los conceptos de Sistemas Educativos Basados en la Web y Sistemas Educativos Adaptativos Hipermedia como paso previo para introducir los elementos relacionados con la búsqueda de objetos de aprendizaje. En el estado del arte (sección 5) se define la manera en que en la actualidad se aborda la búsqueda de objetos de aprendizaje mediante el uso de ontologías. En la sección 6 se hace un análisis del problema presentado en la sección anterior, se define el modelo arquitectónico bajo el cual se plantea la construcción de la aplicación, se detalla la ontología construida con base en el modelo planteado en la sección anterior y por último se describe la manera en

que está construida la aplicación OASearch, la cual permite realizar búsquedas de objetos de aprendizaje con base en conceptos previamente definidos demostrando así mismo la tesis principal de este trabajo.

1. OBJETIVOS

1.1. Objetivo General:

Desarrollar un prototipo de aplicación informática (motor de búsqueda) que demuestre la factibilidad de crear, de una manera desacoplada de la aplicación, consultas predefinidas que obedecen a conceptos específicos de búsqueda que luego el usuario puede utilizar para recuperar objetos de aprendizaje relevantes en un Sistema Educativo Basado en la Web Semántica.

1.2. Objetivos Específicos:

- Establecer un modelo de aplicación con base en las tecnologías disponibles para la Web Semántica.
- Diseñar una ontología que permita modelar la manera en que deben ser creadas las consultas para su uso como conceptos de búsqueda dentro de la aplicación.
- Definir una instancia de concepto de búsqueda que se pueda inferir al interior de los metadatos de los objetos de aprendizaje y modelarlo a través de la ontología para realizar la prueba correspondiente de la aplicación.

2. ALCANCE Y PRODUCTOS

Este trabajo plantea la elaboración de un prototipo el cual se constituye en una prueba de concepto para lo descrito en los objetivos. No busca desarrollar una aplicación que pueda ser usada así como se entrega sino que sirva de base para un posible desarrollo más grande en el futuro.

Dentro de la construcción del prototipo se utiliza el concepto de “Estilos de aprendizaje” como el tipo de búsqueda predefinido en el que se basa la conceptualización. No se busca en este trabajo definir lo que es un estilo de aprendizaje ni definir los criterios para inferirlo a partir de los metadatos de un objeto de aprendizaje. En vez de esto se toman investigaciones y conceptualizaciones elaboradas en otras instancias, solo como ejemplo para la construcción de un modelo de datos. Una conceptualización distinta de lo que es un Estilo de Aprendizaje o de la manera en que se construye no afectará de ninguna forma lo que se expresa en este trabajo.

3. METODOLOGÍA

Para el desarrollo de este trabajo se definen inicialmente unas bases conceptuales que se expresan en el marco teórico luego de hacer un sondeo de los referentes más reconocidos en el campo de la web semántica.

Luego de esto, se realiza una revisión bibliográfica exhaustiva siguiendo un enfoque que va de lo general a lo particular para conformar un estado del arte. De esta forma inicialmente se indaga el tópico de la web semántica y los sistemas educativos, luego, yendo más hacia lo particular el tópico de objetos de aprendizaje en relación a la web semántica y por último, el tema concreto de este trabajo, relacionado con la búsqueda de objetos de aprendizaje en un entorno de web semántica.

Luego de tener un marco teórico y un estado del arte definido se aborda el problema particular de este trabajo mediante la definición de una ontología para el dominio concreto de los objetos de aprendizaje. Se busca hacer un reuso de ontologías existentes y no hacer un desarrollo desde cero. Es así como se selecciona una Ontología que luego es adaptada a las particularidades del problema planteado en este trabajo.

Finalmente, con la Ontología definida se trabaja en el desarrollo de un prototipo que es mejorado iterativamente hasta obtener los resultados esperados.

4. MARCO TEÓRICO

4.1. La web semántica y el descubrimiento de información.

Para abordar el tema de la búsqueda semántica mediante el uso de ontologías es necesario establecer una serie de fundamentos teóricos que en la actualidad se orientan principalmente a las tecnologías web y específicamente al uso de la web semántica como el repositorio de conocimiento por excelencia.

En la actualidad es posible pensar en múltiples formas de implementar y manipular bases de conocimiento, sin embargo, debido a las ventajas que se obtienen de utilizar una red como internet para manejarlas de una manera distribuida y a todas las bases que se han sentado para el tratamiento de información a través de la web y específicamente a través de la web semántica, aquí se abordará el tema de esta tesis desde ese punto de vista.

Aproximadamente desde el año 2001 y a raíz de un artículo[3] publicado en la revista Scientific American por Tim Berners-Lee, Ora Lassilla, y James Hendler comenzó a conocerse para un amplio público el concepto “Web semántica”. En este artículo se pusieron de manifiesto varios conceptos de importancia para la construcción de la web semántica como: XML, RDF, Ontologías, Reglas de inferencia y agentes.

Hoy en día, la web ha evolucionado a un punto en el que es posible implementar bases de conocimiento integradas a la web semántica mediante el uso de diferentes lenguajes y modelos de representación. En este capítulo se hará un recorrido rápido por todas esas tecnologías que permiten configurar la web semántica.

Cuando se habla de semántica se habla de significado. El objetivo de la web semántica es proveer de significado a los datos subyacentes, de esta forma se puede lograr un uso más efectivo de la información. Sin embargo, esta “provisión” de significado no se hace para que la información pueda ser entendida por los humanos sino por máquinas o agentes de software.

Lo anterior es un nuevo enfoque considerando que la gran mayoría de información presente en la web está pensada para ser leída por humanos [4, p. 24] incluso cuando el contenido es obtenido originalmente de bases de datos que por definición ya tienen alguna estructura y en consecuencia un significado de la información que contienen para un dominio muy específico.

Desde una perspectiva actual, podría decirse que los motores de búsqueda basados en palabras clave son la principal manera que tienen los usuarios de acercarse y utilizar la web[4]. No se puede negar que la web no sería lo que es si

no fuera por los motores de búsqueda, pero como el mismo Antoniu lo expone existen una serie de problemas asociados con su uso como el hecho de que proveen una alta recuperación de documentos pero una baja precisión, que los resultados son muy sensibles al vocabulario, que en ciertas ocasiones la recuperación es baja o nula, o que los resultados en sí son páginas web más que unidades de información.

En este campo de las búsquedas de información en la web (Recuperación de la información), la web semántica plantea mejoras o soluciones a la actual problemática. Y fue incluso desde que Berners-Lee planteo el primer plan de trabajo para la web en 1998 [5] que se perfilaban las posibilidades que este nuevo enfoque traería consigo. En esta fase de desarrollo de la web semántica se planteaba la posibilidad de utilizar los motores de búsquedas tal cual funcionan en la actualidad en unión con un motor de razonamiento. De esta forma un motor de búsqueda podría mejorar sus respuestas haciendo uso de la inferencia y así mismo, para cada respuesta se podría ofrecer una prueba del proceso de razonamiento que la produjo.

Los fundamentos teóricos en los que se basa la web semántica pueden trazarse desde el trabajo de Leonhard Euler en su famoso artículo “Los siete puentes de Konigsberg” en donde sienta las bases de la teoría de grafos. Así mismo, la web semántica hace uso de la lógica descriptiva desarrollada más recientemente y que deriva de la lógica de primer orden. La lógica descriptiva establece las reglas para la construcción de representaciones del conocimiento válidas y de utilidad, que sean decidibles, que puedan producir una respuesta [6, p. 24].

Así como la web semántica se basa en fundamentos teóricos avanzados de la teoría de la información, hace uso además de una pila de tecnologías probadas y con un buen nivel de maduración. La siguiente gráfica ilustra todas esas capas de tecnologías que conforman la web semántica.

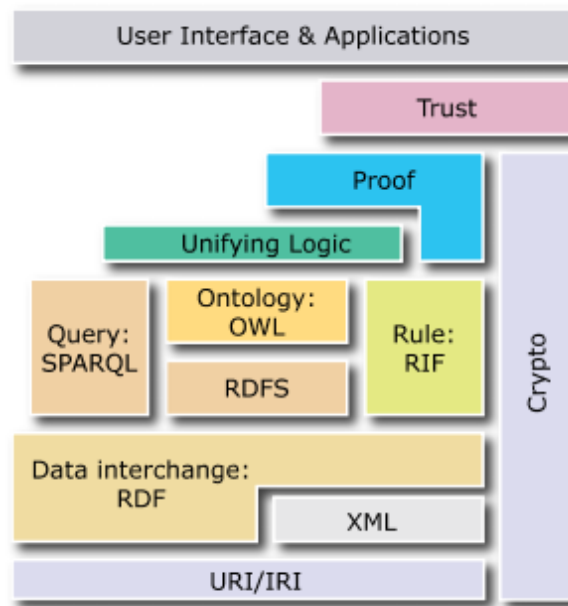


Ilustración 1 Diagrama de capas de la Web Semántica.

Como se puede ver en la gráfica anterior (Ilustración 1), la web semántica aborda temas como el manejo de relaciones de confianza y pruebas de razonamiento. Utiliza lenguajes basados en XML para el intercambio de reglas y la realización de consultas, para el manejo de ontologías y el modelado de la información. Tiene en cuenta además el manejo de espacios de nombres y utiliza el concepto de URI (Universal ResourceIdentifier) para dotar de una identidad a cada objeto presente en la web. Con todos estos elementos es posible construir aplicaciones para la web semántica. En las siguientes secciones de este marco teórico se abordará con más detalle las tecnologías enunciadas de acuerdo a la función que prestan.

4.1.1. Modelado de información

4.1.1.1. Principios de modelado de información

En los sistemas de software, cuando se implementa un modelo de información lo que se busca es capturar los objetos del mundo real al definir sus atributos los cuales son los que determinan las operaciones que con estos objetos se pueden llevar a cabo. En los sistemas semánticos este modelado de la información es especialmente importante porque es mediante un modelado riguroso de conceptos y objetos que es posible reusar datos entre diferentes aplicaciones.

Así como se podría pensar en los modelos de objetos y los modelos relacionales como formas de modelar la información, la web semántica plantea su propio modelo, el cuál a diferencia de los anteriores está diseñado para permitir un fácil intercambio de datos e interoperabilidad.

Cuando se habla de intercambio de datos se habla de dos componentes: Intercambio de la sintaxis e intercambio de la semántica[6, p. 65]. Este

intercambio se puede lograr en sistemas de software mediante el uso de diferentes formas de representación de la información[6, p. 66] como por ejemplo los objetos serializados, las bases de datos relacionales y el lenguaje extensible de marcado (XML). Cada una de las aproximaciones anteriores está en capacidad de abordar, con diferentes limitaciones, el problema del intercambio de sintaxis, pero el intercambio de la semántica plantea otros retos que han sido manejados al considerar los datos y los metadatos como una unidad [6, p. 68]

Retomando lo presentado en la ilustración 1, el intercambio de información es posible gracias a la serialización que provee XML y es sobre esta tecnología que se implementa RDF (Resourcedescription Framework) el cual es un modelo de datos optimizado para compartir e intercambiar información. En la sección 4.1.1.2 se ampliará más la manera en que funciona RDF.

4.1.1.2. Marco para la descripción de recursos RDF

Dentro del modelo de capas (Ilustración 1) de la web semántica se define una instancia en la cual se representa el modelo de información y se establecen las fundaciones para capas más altas. Se trata del marco de descripción de recursos RDF (Por sus siglas en inglés, ResourceDescription Framework). Se puede decir en general, que RDF es un lenguaje para representar información acerca de recursos en la web[7].

RDF provee un marco común de trabajo para expresar la información, de esta forma es posible intercambiar información entre aplicaciones sin pérdida de significado. Esta habilidad significa que la información puede estar disponible incluso para aplicaciones diferentes a esas para las que fue creada originalmente. [7]

RDF se fundamenta en la descripción de recursos. Un recurso es cualquier cosa que pueda ser nombrada[6, p. 69]. De esta forma, para hacer referencia a un recurso se utilizan los URIs (Uniform Resource Identifiers) o una extensión de este último que incluye el manejo de caracteres unicode y se conoce como IRI (Internationalized Resource Identifier). Algunos URIs en particular proveen no solo una manera de nombrar un recurso sino también información acerca de cómo accederlo. Es allí donde surgen los más familiares URL (Uniform Resource Locator).

Con el uso de URIs y describiendo los recursos en términos de propiedades es posible expresar conceptos simples sobre los recursos como un grafo de nodos y arcos representando los recursos, sus propiedades y sus valores. Miller propone un ejemplo simple para ilustrar lo anterior en el que utiliza un grafo (Ilustración 2) para representar las siguientes proposiciones [7]: Existe una persona que se identifica por 'http://www.w3.org/People/EM/contact#me', cuyo nombre es Eric Miller, cuya dirección de correo es em@w3.org, y cuyo título es Dr."



Ilustración 2 Ejemplo de grafo para representación de recursos

La ilustración 2 muestra entonces la manera en que RDF utiliza las URIs para identificar:

- individuos, e.g., Eric Miller, identificado por <http://www.w3.org/People/EM/contact#me>
- tipos de cosas, e.g., Person, identificado por <http://www.w3.org/2000/10/swap/pim/contact#Person>
- propiedades de esas cosas, e.g., mailbox, identificado por <http://www.w3.org/2000/10/swap/pim/contact#mailbox>
- valores de esas propiedades, e.g. <mailto:em@w3.org> como el valor de la propiedad mailbox

Proposiciones como las que se muestran en el anterior ejemplo son las que conforman los bloques de construcción de la web semántica [8, p. 19] y son conocidas como tripletas debido a que están compuestas por tres elementos fundamentales: El sujeto, el predicado y el Objeto.

Normalmente el sujeto en una triplete corresponde a una entidad. Los predicados son una propiedad de la entidad a la cual están ligados. Los objetos pueden ser de dos clases: Entidades, que pueden ser el sujeto en otras tripletas y valores literales como cadenas o números.

Con todo lo anterior en cuenta, un grafo como el de la ilustración 2 podría expresarse en RDF/XML[9] de la siguiente manera.

```
<?xmlversion="1.0"?><rdf:RDFxmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:contact="http://www.w3.org/2000/10/swap/pim/contact#"><contact:Personrdf:about="http://www.w3.org/People/EM/contact#me">
<contact:fullName>Eric
Miller</contact:fullName><contact:mailboxrdf:resource="mailto:em@w3.org"/><contact:personalTitle>Dr.</contact:personalTitle></
contact:Person></rdf:RDF>
```

Tabla 1 Ejemplo de código en RDF/XML

RDF/XML no es el único formato en el que es posible modelar información para la web semántica de tipo RDF, existen otros como el formato Turtle (Terse RDF triple lenguaje[10] y N-Triples[11, p. 3].

El World Wide Web Consortium es la organización que se ha encargado de desarrollar y publicar todas las recomendaciones referentes a la web semántica y en el caso de RDF estas recomendaciones se componen de la definición [7], los conceptos y la sintaxis [12], la semántica[13], Vocabulario[14], Casos de prueba[11] y la especificación sintáctica para RDF/XML [9].

4.1.2. Expresando la semántica

La comunicación de una información incluye la sintaxis y la semántica. Proveer de semántica a un documento implica disponer de un mecanismo para definir conceptos y relaciones de una manera precisa y yendo más allá de la sintaxis.

Retomando los fundamentos de la web semántica, los factores preponderantes que motivan su construcción son [6, p. 94]:

- Hacer que la información disponible en internet sea entendida por máquinas de software.
- Proveer un modelo semántico lo suficientemente rico como para expresar un dominio del conocimiento.
- Habilitar el intercambio de información entre dominios.

Lograr estos objetivos implica realizar una serie de construcciones que permitan expresar la semántica para un modelo de información específico.

Si bien, como se ve en la sección 4.1.1.2, RDF es una herramienta poderosa para describir información, si se usa aisladamente carece de un soporte explícito para expresar los significados o la semántica detrás de las descripciones.

Siguiendo con lo mostrado en la ilustración 1, la arquitectura de la web semántica define un par de niveles por encima de RDF con los cuales es posible proveer semántica. Esto se hace utilizando RDFS[14] y OWL[15] junto con RDF como se verá en la sección 4.1.2.1.

4.1.2.1. Formas de expresar la semántica en la WEB

Existen varias formas de expresar el significado de la información. Ejemplos de ello son los vocabularios, las taxonomías y las ontologías. La diferencia entre cada una de estas herramientas radica en su expresividad, o qué tanto significado pueden aportar a los términos que describen [6, p. 99]

Un vocabulario es una colección de términos definidos sin ambigüedad. Una taxonomía organiza los términos de una manera jerárquica. Las ontologías tienen un vocabulario de términos predefinido para definir conceptos y relaciones para un área específica de interés o dominio.

Con RDFS es posible proveer un vocabulario específico para RDF que puede usarse para definir taxonomías de clases y propiedades y especificaciones simples de dominio y rango para las propiedades. Sin embargo, se han identificado cierto número de casos de uso característicos en los que se requiere mucha más expresividad de la que RDF junto con RDFS puede ofrecer[4]

Por lo anterior fue que diferentes grupos de investigación alrededor del mundo comenzaron a desarrollar un lenguaje de modelado de ontologías más poderoso que llevó a la definición de DAML+OIL el cual derivó en el que se ha convertido en el estándar de los lenguajes para ontologías OWL que a la fecha de este escrito va en su versión 2 [16]. OWL se define entonces como un lenguaje de ontologías para la web semántica con un significado formalmente definido [16].

En la sección 4.1.2.2 se ahondará en la función de las ontologías y la manera en que estas pueden construirse.

4.1.2.2. Ontologías

El término “Ontología” deriva de la filosofía en donde se utiliza para referirse a la disciplina que estudia “la naturaleza de la existencia” es una rama de la metafísica que se preocupa por identificar, en los términos más generales, la clase de cosas que actualmente existen y cómo describirlas [4, p. 10].

Las ciencias de la computación tomaron prestado este término para darle un significado técnico en lo específico. Según T.R. Grubers’s, *una ontología es una especificación formal y explícita de una conceptualización.*

En esencia, una ontología sirve para describir formalmente un dominio del discurso. Normalmente, una ontología consiste en un conjunto de términos y las relaciones entre esos términos[4, p. 11]. Dado lo anterior, es posible realizar una serie de razonamientos inherentes a la estructura ontológica y derivar nueva información de una representación de información.

Lenguajes como el OWL pueden expresar ontologías que pueden ser usadas en la web semántica teniendo en cuenta una serie de propiedades que derivan principalmente de la naturaleza distribuida de la web como se verá a continuación.

4.1.2.2.1. *Ontologías en la web*

Como se dijo en la introducción a este marco teórico (sección 4.1) , los recursos en la internet tienen una naturaleza distribuida y como resultado, las descripciones de recursos contenidas en la web semántica también lo son. OWL soporta este modelo distribuido porque está construido sobre RDF, el cual permite declarar y describir recursos localmente y llamarlos remotamente. OWL también provee un mecanismo para importar y reusar ontologías en un ambiente distribuido (Ver “Encabezado de la ontología” en sección 4.1.2.2.2.).

Para proporcionar una base en la cual se puedan hacer inferencias validas en el modelo distribuido de la web semántica deben definirse dos supuestos fundamentales: El supuesto del mundo abierto y el supuesto de los no nombres únicos [6, p. 102]

El supuesto del mundo abierto establece que la verdad de una sentencia es independiente de si esta es conocida. Este supuesto permite que la nueva información siempre será aditiva, puede que sea contradictoria pero no puede remover información declarada previamente[6, p. 103]. Esto implica que con OWL las inferencias solo pueden hacerse con la información que es conocida.

El supuesto de los no nombres únicos establece que a menos que se diga lo contrario explícitamente, no es posible asumir que los recursos que son identificados por URIs diferentes son diferentes.

4.1.2.2.2. *Elementos de las ontologías*

Las ontologías normalmente se manejan como documentos en la web y estos documentos están formados por un encabezado de la ontología, anotaciones, clases y definiciones de propiedades o axiomas, hechos sobre individuos y definiciones de tipos de datos.

Encabezado de la ontología. Es un recurso que representa la ontología en sí misma. Describe la ontología y normalmente contiene comentarios, marcas, información de versiones y las sentencias para importación que relacionan otras

ontologías que puedan ser necesarias para abarcar completamente los conceptos y las relaciones descritas en la misma ontología.

Clases e individuos. Una clase es un tipo especial de recurso que representa un conjunto de recursos que comparten características comunes. Un recurso que es miembro de una clase es un individuo y representa una instancia de esa clase.

Propiedades. Son usadas como predicados en las sentencias que describen los individuos. Hay dos tipos principales de propiedades en OWL: Las propiedades de objeto y las propiedades de tipos de objeto.

Anotaciones. Son similares las propiedades OWL normales pero no tienen una semántica asociada y se usan principalmente en la interfaz con el usuario humano.

Tipos de datos. Representan un rango de valores de datos. Ejemplos de ello son el tipo entero, string y tiempo. Con OWL 2 es posible definir tipos de datos complejos utilizando enumeraciones o restricciones de faceta.

4.1.3. Implementando aplicaciones para la web semántica

4.1.3.1. *Frameworks de desarrollo de aplicaciones para la web semántica y sus componentes.*

Para poder sacar todo el provecho a las posibilidades que ofrece OWL y RDF para especificar semántica y definir un modelo de conocimiento es necesario utilizar herramientas de software o frameworks con diferentes componentes integrados. Un ejemplo de framework es “Jena Semantic Web Framework”. Este framework permite realizar adaptaciones para crear mecanismos propios de almacenamiento RDF o motores de razonamiento. Otro ejemplo es “Sesame RDF framework”.

En general, un framework es un conjunto de herramientas que permiten crear y trabajar con una base de conocimiento[6, p. 143].

Normalmente un framework para la web semántica se compone de los siguientes elementos:

- El repositorio o Almacen RDF
- Una API de acceso o procesador de consultas
- Motor de razonamiento

En esencia, una base de conocimiento es una colección de hechos o proposiciones y un framework permite guardar, proveer acceso y realizar inferencias sobre esas proposiciones.

4.1.3.2. Almacenamiento y recuperación de información RDF

Los almacenes o repositorios RDF proveen de persistencia a las sentencias expresadas en un grafo RDF. Lograr un almacenamiento y acceso eficiente es todavía un área de investigación. Existen almacenes RDF basados en bases de datos relacionales las cuales usan tablas para almacenar las tripletas sujeto-predicado-objeto. También existen almacenes RDF basados en el modelo de grafos y han sido desarrollados para expresar de una manera más directa la estructura de un modelo RDF y mitigar los problemas de desempeño que surgen con los almacenes basados en bases de datos relacionales.

Para recuperar la información contenida en un repositorio RDF existen varias posibilidades. La más natural de ellas implica utilizar un browser para navegar el grafo de recursos interconectados a medida que se va por la web. Una herramienta para este tipo de navegación la proporciona el sesame framework. El anterior puede ser un método adecuado para que un usuario humano dé un vistazo a la información o realice una tarea de depuración en el repositorio pero cuando se pretende que un agente de software acceda al mismo repositorio para realizar una tarea de una manera más enfocada y precisa es necesario utilizar las interfaces de programación que ofrecen los frameworks (APIs) o usar el más común de los métodos: A través de una interfaz de consulta como SPARQL.

4.1.3.3. Haciendo uso de la semántica OWL para manejar inferencias.

El mayor beneficio de utilizar OWL es el de poder definir semántica que enriquece la información. Para que la información contenida en una base de datos pueda ser interpretada y pueda sacársele provecho es preciso hacer uso de algún motor de inferencias que puede lograr este propósito a través de una definición de reglas y el uso de un motor de reglas así como disparadores (triggers) en una base de datos o almacén RDF, árboles de decisión, algoritmos de tableau o inclusive a través de código para la lógica del negocio.

La inferencia basada en reglas es particularmente usada en muchos Frameworks. Los motores que usan este tipo de inferencia combinan las declaraciones contenidas en la base de conocimiento con un conjunto de reglas lógicas de tal manera que se puedan derivar nuevas declaraciones o realizar acciones. La ejecución de inferencias en una base de conocimiento se puede realizar de dos maneras: utilizando encadenamiento hacia adelante o encadenamiento hacia atrás. En el primer caso todas las declaraciones se derivan de lo contenido en la base de conocimiento y son agregadas a la misma base de conocimiento. En el segundo caso se identifica la declaración a la que se quiere llegar y se continúa identificando hacia atrás todas las declaraciones que pueden derivar en la actual.

El método de inferencia que se utilice depende de las limitaciones del sistema de computo y de los requerimientos que se tengan. En el caso del encadenamiento hacia adelante se prioriza el desempeño de las consultas más que el desempeño en inserción y remoción de declaraciones. En el encadenamiento hacia atrás se mejora el desempeño de inserción y remoción y puede hacerse indispensable en

sistemas distribuidos de razonamiento en los que no existe una base de datos centralizada.

4.1.3.4. Los perfiles OWL

Dentro de OWL se han definido sublenguajes con el objetivo de lograr eficiencia computacional. A estos sublenguajes se les conoce como perfiles, y básicamente definen una sintaxis restringida o modificada y así mismo, algunas restricciones no estructurales para el uso de OWL. Estas restricciones se definieron con el objetivo de lograr sublenguajes decidibles.

Los perfiles básicos que se manejan son OWL-DL y OWL-Lite. En el caso de OWL-DL se definió este perfil con el objetivo de proveer muchas de las capacidades de la “Lógica descriptiva” el cual, como se sabe, es un importante subconjunto de la lógica de primer orden. OWL-DL contiene todo el vocabulario de OWL completo, pero introduce la restricción de que su semántica no puede ser aplicada a un documento RDF que trate una URI como un individuo y una propiedad de clase. La anterior y algunas otras restricciones hacen a OWL-DL decidible. Específicamente, existen tres perfiles que se definen con base en OWL-DL: OWL-EL, OWL-QL, OWL-RL.

OWL-EL se diseñó para proveer un tiempo polinomial de computación para determinar la consistencia de una ontología y ajustar individuos a clases. OWL-QL fue diseñado para proveer satisficibilidad en las consultas conjuntivas dentro de un espacio logarítmico con respecto al número de declaraciones en la base de datos que está siendo consultada. OWL-RL fue diseñado para ser lo más expresivo posible sin sacrificar el uso de reglas y un sistema de procesamiento de reglas.

4.1.3.5. Manejo de Ontologías

4.1.3.5.1. Desacople del modelo de conocimiento y la aplicación.

Desacoplar el modelo de conocimiento de la aplicación es crítico para poder lograr el intercambio de información y el reúso entre usuarios y aplicaciones. Un modelo de datos muy acoplado a la aplicación limita al usuario para transferir información entre aplicaciones y ambientes.

Aunque fuera fácil traducir un modelo de conocimiento de una aplicación a otra, algunas complicaciones se mantienen que tienen que ver con la expresividad limitada de los modelos de conocimiento tradicionales como las bases de datos relacionales y los esquemas XML que obligan a implementar una “lógica del negocio” dentro de la aplicación y esto hace que los datos estén muy fuertemente acoplados a la aplicación. [6, p. 173]

No es posible hacer que todas las aplicaciones operen afuera de un modelo común de conocimiento, así como no se le puede pedir a todos los usuarios de varios dominios del conocimiento que compartan el mismo modelo de conocimiento. La solución es usar ontologías específicas a la aplicación como interfaces entre el modelo de conocimiento del usuario y de la aplicación. Así cada aplicación tiene su propio modelo de conocimiento y los desarrolladores pueden integrar sus propios modelos de conocimiento usando las posibilidades de mapeo de OWL y SWRL.

4.1.3.5.2. *Compartición de información entre dominios y límites de aplicaciones.*

Compartir información entre dominios y aplicaciones es posible mediante el establecimiento de los conceptos en cada uno [6, p. 174]. El esfuerzo necesario para integrar dos dominios de conocimiento (o aplicaciones) es directamente proporcional al número de conceptos diferentes y desconectados en cada uno.

El proceso puede incluir:

- Un mapeo simple entre conceptos idénticos
- Establecimiento de relaciones de subclase o subpropiedad.
- Operaciones complicadas como cálculos matemáticos o manipulación de caracteres.

Algunas de estas relaciones pueden ser establecidas utilizando construcciones ontológicas, pero otras requieren el uso de un lenguaje de reglas como SWRL.

Para facilitar esta integración de dominios lo adecuado es minimizar el número de conceptos desconectados entre dominios. Y para lograr esto, una de las mejores maneras, es hacer un reuso o extensión de conceptos ya existentes en otras ontologías. Una nueva ontología solo debe ser creada si esto es absolutamente necesario.

4.1.3.5.3. *Ontologías fundacionales*

Para lograr un intercambio de información entre dominios diferentes como se enunció en el apartado 4.1.3.5.2 es necesario recurrir a una ontología de base la cual debe ser extendida y así disminuir el número de conceptos desconectados entre dominios específicos.

Estas ontologías de base son las que se conocen como ontologías fundacionales. Cualquier ontología puede ser usada como ontología de base, pero las ontologías fundacionales normalmente son diseñadas pensando en el reuso y la extensión.

Las ontologías fundacionales crean un espacio en el cual diferentes dominios del conocimiento comparten una serie de conceptos raíz y estos dominios del conocimiento pueden enriquecer aún más la ontología fundacional adicionando conceptos más generales (En anchura) o más específicos (En profundidad) de acuerdo a lo requerido por cada aplicación específica. Cuando esto se logra, los diferentes dominios pueden compartir información a nivel fundacional sin importar como ha sido extendida la ontología.

4.1.3.5.4. *Registros, repositorios y buscadores de Ontologías*

En la red hay muchas ontologías que pueden ser utilizadas, pero antes de poder utilizarlas, es preciso conocer su ubicación. Para esto existen una serie de herramientas que se pueden clasificar en registros, repositorios y buscadores de ontologías.

Un registro es una aplicación web en la que cualquiera puede ir a registrar una ontología específica proporcionando los datos sobre cómo esta puede ser encontrada. Funciona básicamente como un directorio de ontologías.

Un repositorio es más centralizado debido a que almacena y gestiona un conjunto de ontologías permitiendo a los usuarios subirlas y manejar diferentes versiones de la misma ontología. Estos repositorios proporcionan además filtros y mecanismos para facilitar la selección de una ontología en particular.

En la red ya existe un número considerable de repositorios en los cuáles se pueden encontrar ontologías adecuadas a diferentes dominios y en los que cualquiera puede publicar las suyas.

Para el caso de los buscadores de ontologías no se requiere que un usuario suba activamente una ontología en alguna aplicación como sí sucede con los registros y repositorios. Los buscadores tienen un enfoque diferente al recorrer la web en búsqueda de ontologías que puedan ser agregadas a un índice que luego podrá ser consultado por el usuario. Uno de los buscadores de ontologías más usados es Swoogle (<http://swoogle.umbc.edu>)

4.1.4. *Descubrimiento de información.*

Dependiendo de qué se busca exactamente y dónde podría estar la información se conoce como está estructurada la información.

El descubrimiento de información se puede hacer a través de la navegación, la búsqueda y/o la consulta.

Navegación: Se hace recorriendo tripleta por tripleta sin un objetivo en particular. A través de la dereferenciación es que se pide y se recibe la

representación de un recurso URI. A este proceso se le conoce como descubrimiento libre.

Búsqueda: Se basa en la navegación. En este tipo de descubrimiento puede existir un índice previo de documentos web semánticos o información que luego es consultada usando palabras clave.

Consulta: Permite hacer preguntas explícitas, complejas y estructuradas. Se basa en una sintaxis y semántica formal. Usa el lenguaje de consulta SPARQL.

La navegación se puede volver problemática cuando se convierte en un trabajo rutinario, además, simplemente el tiempo no alcanza para navegar suficientes fuentes de datos para resolver preguntas que se conocen. En este caso es mejor recurrir a la búsqueda.

Cuando se hace búsqueda lo que se espera es poder sacar ventaja de algún tipo de pre procesamiento de datos como descubrimiento, ordenamiento, filtrado o indexado.

Para consultar la web semántica se requiere de un lenguaje que reconozca RDF como la sintaxis fundamental. SPARQL es uno de estos lenguajes.

4.2. Los objetos de aprendizaje y metadatos

4.2.1. Los objetos de aprendizaje

Muchas perspectivas han dado forma al concepto de objeto de aprendizaje. Desde la concepción de D. Wiley ya mencionada en la introducción, pasando por la definición más general que hace la IEEE como “cualquier entidad, digital o no, que pueda ser utilizada para el aprendizaje, la educación o la capacitación” [2] hasta llegar a definiciones más centradas en la estructura como la que hace Chiappe y compañía [17] "Una entidad digital reutilizable y auto-contenida, con un claro propósito educativo, con al menos tres componentes internos editables: contenido, actividades de aprendizaje y elementos de contexto. Los Objetos de Aprendizaje deben tener una estructura externa de información para facilitar su almacenaje y recuperación: los metadatos."

Es de la última definición dada por Chiappe que se puede obtener el concepto de metadato ligado al de Objeto de aprendizaje y para los efectos de este documento es la que mejor se ajusta.

4.2.2. Metadatos para los objetos de aprendizaje

Así mismo, existen diferentes estándares que pueden ser usados para la descripción de los objetos de aprendizaje a través de Metadatos. IMS Object Metadata [18], Dublin Core [19] o SCORM [20] son ejemplos de este tipo de estándares. Sin embargo, el estándar más ampliamente utilizado es LOM [2], el cual permite no solo describir los objetos de aprendizaje como un recurso digital,

sino también como un objeto con una intención educacional definida. En general LOM apunta a facilitar la compartición de descripciones de los recursos de aprendizaje. También determina los aspectos de un determinado objeto de aprendizaje que necesitan ser descritos por su diseñador y el vocabulario apropiado que debe ser usado.

Básicamente, LOM es una jerarquía de elementos agrupados en nueve categorías que son [21]

- General: Información general acerca del objeto de aprendizaje
- Ciclo de vida: La historia del objeto de aprendizaje, su estado actual y la información de los que aportaron en su construcción.
- Meta-metadatos: Información acerca del registro de metadatos
- Educacional: Características de tipo educacional , clasificaciones.
- Técnicos: Requerimientos técnicos y características.
- Derechos: Derechos de propiedad intelectual y condiciones de uso.
- Relación: Su relación con otros objetos de aprendizaje.
- Anotación: Comentarios acerca del objeto y el que comenta.
- Clasificación: Relación con un sistema de clasificación.

Aunque el estándar LOM es útil en muchas situaciones, hay algunas limitaciones que es preciso tener en cuenta [22]:

- Las lecciones completas son indexadas de la misma forma que los ejercicios únicos o las imágenes.
- La definición de metadatos y el significado de los términos no está completamente resuelto.
- La unidad de indexación es un archivo que representa una unidad técnica, pero no una de tipo educativo.
- Algunas ambigüedades en el modelo hacen difícil su uso

Para ayudar a resolver algunos de estos problemas el estándar LOM usa lo que se llama Perfiles de Aplicación. También, la indexación de LOM es completada con representación de contenido basada en una ontología temática.[22]

El uso adecuado de los metadatos educativos puede facilitar la reusabilidad y las diferentes combinaciones de los objetos de aprendizaje en diferentes contextos de aprendizaje y escenarios. De esta manera, los objetos de aprendizaje requieren ser descritos de una forma consistente y universal por toda la comunidad educativa, haciendo necesaria la formulación de ontologías, así como estándares y especificaciones globales [23].

Aunque el estándar LOM describe la estructura de los metadatos, no define cómo esos metadatos deben ser representados o usados en los sistemas de gestión del aprendizaje. Para usar este estándar es necesario expresarlo en un formato específico que pueda ser manejado por la maquina.

4.3. Sistemas educativos basados en la web

Según la definición de Davedsik [24, p. 1], informalmente hablando, la educación basada en la web involucra todos los aspectos y procesos de educación que tienen que ver con la *World Wide Web* como un medio de comunicación y tecnología de soporte.

De acuerdo a lo que expresan Keegan[25] y Paulsen [26], se puede decir que la educación basada en la web está caracterizada por:

- La separación entre profesores y estudiantes (Que lo distingue de la educación presencial)
- La influencia de una organización educativa (Que lo distingue del auto-estudio o tutorías privadas)
- El uso de tecnologías Web para ofrecer y/o distribuir algún contenido educativo.
- La provisión de una comunicación de doble vía a través de internet para que de esta manera los estudiantes se puedan beneficiar de la comunicación entre ellos, con los profesores y con el equipo involucrado.

Según el mismo Davedzik, hay un número importante de conceptos relacionados con la Educación Basada en la Web como el e-aprendizaje (e-Learning), la educación a distancia y el aprendizaje adaptado.

El e-aprendizaje (e-learning) se define como un aprendizaje interactivo en el cual los contenidos de aprendizaje están disponibles en línea y provee una retroalimentación automática a las actividades de aprendizaje del estudiante. Se diferencia de la educación basada en la web en la medida que esta última cubre un rango más amplio de servicios que el e-learning. Normalmente el e-learning se centra más en los contenidos de aprendizaje.

La educación a distancia, aunque muchas veces se usa como sinónimo de la educación basada en Web, es en realidad un concepto más general que incluye también el concepto de e-learning. La educación a distancia utiliza un amplio rango de tecnologías que no necesariamente son de tipo Web incluso tampoco de tipo digital.

Uno de los aspectos importantes que plantea el uso de la educación basada en la Web es la necesidad de hacer un mapeo entre esta tecnología y los flujos de trabajo educativos para poder alcanzar un aprendizaje efectivo. Los Flujos de trabajo educativo involucran la creación, el acceso, la enseñanza y/o la manipulación de materiales de aprendizaje y otros recursos así como la interacción y la comunicación entre diferentes categorías de usuarios de los sistemas de aprendizaje basados en la web. De otro lado, es importante resaltar

que los estudiantes demandan sistemas de este tipo que sean de alta calidad en términos de contenido, pedagogía y marco tecnológico [24, p. 16]

Dos formas efectivas de mejorar la calidad de los Sistemas de Aprendizaje Basados en la Web son introducir inteligencia en el sistema y hacer que se adapte a las necesidades particulares del estudiante. De las anteriores premisas surgen los conceptos de Sistemas Inteligentes de Tutoría (Intelligent Tutoring Systems) y sistemas educativos adaptativos hipermedia (Adaptive Educational Hypermedia Systems). [24, p. 21]

En los Sistemas Educativos Adaptativos Hipermedia (SEAH) la propiedad de adaptación viene en 3 niveles [27]:

Conectividad: Partes de los contenidos de aprendizaje están interrelacionados de varias formas, permitiendo a los navegantes navegar de diferentes maneras. Los SEAH presentan diferentes enlaces con diferentes indicadores visuales de concordancia y relevancia respecto al usuario, de acuerdo a su modelo de aprendizaje.

Contenido: El sistema muestra información adicional al estudiante para compensar sus vacíos de conocimiento (De acuerdo a su modelo de estudiante), u oculta información innecesaria si concluye que el estudiante ya conoce el tema y no se beneficia de verlo nuevamente.

Cultura: Los SEAH tienen en cuenta el trasfondo, las motivaciones, preferencias y estilos de diferentes estudiantes y adaptan las tareas educativas a cada uno.

El concepto de Sistemas Educativos Adaptativos Hipermedia será crucial, más adelante en el desarrollo de este trabajo en la medida que será a partir de él que se establecerá una prueba de concepto para el buscador de objetos de aprendizaje que se propone desarrollar.

5. ESTADO DEL ARTE

5.1. Introducción

Luego de revisar los conceptos que enmarcan el objeto de este trabajo, es preciso hacer un recorrido por los últimos avances en el campo de la búsqueda de objetos de aprendizaje mediante la web semántica.

En orden de especialización se revisará primero el estado del arte respecto a los sistemas educativos basados en web semántica, luego se revisará cómo un aspecto de estos sistemas, el cual tiene que ver con el uso y tratamiento de los objetos de aprendizaje, ha avanzado en los últimos años. Para finalizar, se abordará la preocupación fundamental de este trabajo mediante una clasificación de algunas de las últimas publicaciones en torno al tema de la búsqueda de objetos de aprendizaje en sistemas basados en la web semántica.

5.2. Web semántica y sistemas educativos

Cuando se habla de Sistemas Educativos Basados en Web Semántica (SEBWS), normalmente se entiende este concepto como una extensión a los avances ya logrados en el campo de la educación basada en sistemas web (Ver sección 4.3.) Debido a que la Web Semántica ofrece numerosas mejoras sobre el uso que se le da a la web tradicional y las aplicaciones que se construyen sobre ella, se espera que los sistemas educativos basados en la Web Semántica tendrán profundos efectos en todos los aspectos y procesos de la educación que tienen que ver con las tecnologías web [24, p. 71]

Las tecnologías para la web semántica han visto su uso en diferentes aspectos relacionados con el aprendizaje en línea: Sistemas Educativos Hipermedia Adaptables, Creación de contenido adaptable, búsquedas personalizadas y composición de contenidos, compartición de modelos de usuarios y captura del contexto [28] son algunos ejemplos.

Carapina presenta en [22, Sec. 3] otros ejemplos del uso de tecnologías para la Web semántica en sistemas de tipo educativo. Carapina define su clasificación en sistemas para, evaluación del conocimiento del alumno, manejo de varios tipos de colaboración, modelado del usuario como una parte importante en el proceso de adaptación de contenidos, análisis de redes sociales con recursos de aprendizaje y por último, sistemas para el manejo de interacciones entre grupos.

Como lo expresa Davedzik en [24]: “La fuerza que promueve los diferentes desarrollos en el campo de los Sistemas Educativos Basados en la Web

Semántica no es la tecnología en sí; Es la fuerte motivación que existe para satisfacer las crecientes necesidades de los estudiantes de una manera más confortable y activadas por la tecnología” Anderson y Whitelock llaman a esta visión *La Web Semántica Educativa* [29] y establecen una serie de recomendaciones para la masificación de su uso.

En [24, p. 72] Davedzik establece el ambiente o disposición necesaria para que exista la posibilidad de enseñar, aprender, colaborar, evaluar y realizar otro tipo de actividades educativas en la Web Semántica. Los actores que intervienen en un SEBWS se definen como Estudiantes, Profesores y Autores. El soporte que esto requiere incluye la definición del *Servidor Educativo* y el *Contenido Educativo* el cual se aloja y se gestiona desde el primero. Se define también el concepto de *Agentes Pedagógicos Inteligentes* los cuales se encargan de permitir el flujo de conocimiento y contenido entre el Cliente y los servidores. Además de esto, se define el concepto de *Servicio Educativo* como un servicio web diseñado específicamente para soportar objetivos de aprendizaje, enseñanza o creación de contenido.

Davedzik define también un marco en el cual clasifica los asuntos más importantes en los cuales se enmarcan los desarrollos de los SEBWS [24, p. 74]: Contenido Educativo, Ontologías, Agentes pedagógicos, Personalización, Lenguajes, Tecnología, Herramientas y Servicios.

El Contenido Educativo se entiende como los objetos de aprendizaje multimedia, ejemplos, preguntas, simulaciones y cosas relacionadas que prepara un autor determinado. El contenido normalmente se estructura en unidades de aprendizaje coherentes basadas en algunas metas u objetivos pedagógicos.

Las ontologías representan conocimiento esencial al definir la terminología, los conceptos, relaciones, jerarquías de conceptos y restricciones. Permiten el poder compartir y hacer reuso del contenido educativo además de propiciar la interoperabilidad de diferentes aplicaciones educativas.

Los agentes pedagógicos son de gran ayuda en localizar, explorar, seleccionar, organizar, integrar y en general, hacer uso del contenido educativo alojado en los servidores educativos.

La personalización se refiere a la posibilidad que debe ofrecer un SEBWS de adaptarse a las características individuales de cada usuario.

Los lenguajes son las diferentes formas de codificar y presentar la información contenida en un material de aprendizaje así como la comunicación entre agentes pedagógicos.

La tecnología se refiere a todo el marco tecnológico que soporta un SEBWS. Aunque no se define como el objetivo final, se reconoce como un activador importante.

Las *herramientas* son usualmente aplicaciones de software que apoyan los procesos de Aprendizaje, enseñanza y creación de contenido. Ejemplos de ello son Los ambientes integrados de aprendizaje, Los Sistemas de Gestión del Aprendizaje, Herramientas de creación de contenidos y similares.

Los *servicios web semánticos* permiten ofrecer a los profesores, estudiantes y autores de contenido acceso a contenidos orientado a servicios en dominios específicos de interés.

A continuación se exponen algunos ejemplos de Sistemas Educativos Basados en la Web Semántica que se enfrentan en diferente medida a uno más de los asuntos expuestos anteriormente:

En [30] Vargas-Vera propone un enfoque semántico para la construcción de un portal educativo que vaya más allá del enfoque centrado en los objetos de aprendizaje. Su propuesta incluye el uso de ontologías de personalización y tecnologías que ofrezcan soporte para el trabajo en red, la localización del conocimiento adecuado, la posibilidad de involucrarse en discusiones de análisis crítico y la posibilidad de dar sentido a diferentes narrativas pedagógicas. Desde el punto de vista arquitectónico se propone un enfoque orientado a servicios donde un *Service Broker* se encargaría de gestionar el registro de los mismos.

Enmarcado en el asunto de Servicios Web Semánticos, Barros [31] propone una forma sistemática de facilitar el proceso de desarrollo al proveer un conjunto de pasos guía para los desarrolladores de sistemas en el sistema Grinv el cual es un *middleware* para el descubrimiento, composición e invocación automática de servicios web semánticos.

Relacionado con el asunto de personalización y ontologías se puede citar el trabajo de Barla et al. [32] Quienes describen una aproximación de modelado del usuario basada en la adquisición automática de su comportamiento y su transformación en un modelo de usuario ontológico.

Respecto al asunto de los agentes pedagógicos, Gladun et al. [33] Usan la web semántica y los sistemas multiagente con el propósito de evaluar el conocimiento del estudiante. Utilizan para esto un método mediante el cual el estudiante construye su propia ontología de dominio que luego es comparada con una ontología de referencia con el objetivo de mostrar aquellas partes del dominio de conocimiento que no han sido correctamente asimiladas.

El sistema blikiBook [34] demuestra el uso de ontologías y mecanismos de marcado semántico para soportar diferentes tipos de conocimiento y aprendizaje adaptado. Al usar descripciones semánticas, el sistema provee a los usuarios la posibilidad de organizar sus objetivos y rutas de aprendizaje que luego pueden ser revisadas colaborativamente.

Relacionado con el asunto de las herramientas y el soporte que estas puedan prestar en el aprendizaje colaborativo puede encontrarse el trabajo de Vega-Gorgojo [35] en el cual se presenta el sistema OntoolSearch el cual busca herramientas apropiadas para diferentes situaciones con el objetivo de ayudar a los tutores. Otro sistema relacionado que Vega-Gorgojo junto con Ruiz-Calleja et al. presenta más recientemente es la herramienta SEEK-AT-WD la cual ofrece un sistema de registro de herramientas ICT basado en un enfoque social-semántico [36]

En un contexto similar, Isotani et al. [37] presentan una herramienta llamada CHOCOLATO que ayuda a los profesores en la planeación de escenarios colaborativos mediante el uso de ontologías que sirven para expresar el conocimiento sobre diferentes pedagogías y prácticas relacionadas a la colaboración.

Los anteriores ejemplos son solo algunos en el amplio universo de los SEBWS. En la actualidad, el concepto sigue evolucionando y es normal comenzar a verlo relacionado con el concepto de E-learning 3.0 [38, p. 0] [39] que sin ser lo mismo enmarca los Sistemas Educativos Basados en la Web Semántica en un sitio de gran importancia para el desarrollo de los futuros sistemas de aprendizaje.

5.3. Los objetos de aprendizaje y la Web Semántica

En la sección anterior se ofrecía un marco para clasificar los asuntos que conforman un Sistema Educativo Basado en la Web Semántica. Uno de esos asuntos consiste en *El Contenido Educativo* dentro del cual se enmarca a su vez el concepto de objeto de aprendizaje de la manera en que se expuso en el marco teórico (Sección 4.2.1). En la presente sección se hará un recorrido por la evolución que han tenido los SEBWS hasta llegar a un estado del arte sobre la manera que manejan los objetos de aprendizaje y definen nuevos usos para los mismos.

Como se expresa en [40], para el año 2003, la mayoría de desarrollos que tenían que ver con los objetos de aprendizaje estaban enfocados hacia los aspectos del manejo de metadatos y al empaquetado de los mismos y no se reconocía un trabajo significativo en los procesos de automatizar el descubrimiento de objetos y el empaquetamiento basado en variables como Objetivos de Aprendizaje o Resultados de Aprendizaje. No se reconocía tampoco un avance significativo en la personalización del aprendizaje basado en objetos desarrollados y ubicados en sitios arbitrarios de Internet. Esta situación se explicaba entonces como una consecuencia de la reciente introducción de los objetos de aprendizaje y en el hecho de que lograr una automatización de esos procesos requeriría una aplicación de técnicas de inteligencia artificial para manejar la ingente cantidad de conocimiento que debía ser representado con relación a los objetos de aprendizaje.

Para ese entonces, ya se preveía la necesidad de desarrollar ontologías para expresar conceptos del dominio de conocimiento y la estructura de los objetos de aprendizaje. [40, Sec. 3], esto permitiría facilitar su descubrimiento y reúso en la red.

Con el advenimiento de la web semántica, estos aspectos comenzaron a ser manejados cada vez en mayor medida. Ya para el año 2011 se sigue reconociendo la necesidad de establecer ontologías para describir los objetos de aprendizaje como una de las mayores preocupaciones a tener en cuenta [21] pero esta vez el enfoque está dado en la necesidad de unificar vocabularios y detener la proliferación de diversas ontologías.

Un ejemplo de modelo ontológico para la descripción de objetos de aprendizaje puede verse en [20] el cual unifica las diferentes facetas de descripción por metadatos que ofrecen estándares como SCORM, LOM, IMS-LD en combinación con un modelo para especificar la teoría pedagógica del objeto.

En general, Davedzik [24, p. 167] define una serie de asuntos relacionados con los estándares para objetos de aprendizaje que deben ser tenidos en cuenta a la hora de diseñar un SEBWS:

- El desarrollo de métodos simples y herramientas para la anotación (Diligenciamiento de la información) de los Objetos de Aprendizaje.
- Diferenciar entre metadatos objetivos y subjetivos.
- Combinar conjuntos de metadatos y esquemas desde múltiples fuentes.
- Fácil integración de la producción de objetos de aprendizaje y su anotación.
- Anotación avanzada y automática.
- Introducción de semánticas formales para los estándares existentes.
- Flexibilidad y dinámicas de asociación entre metadatos y Objetos de Aprendizaje.

Al hacer un barrido del estado del arte en lo referente a la manera en que son tratados los objetos de aprendizaje dentro de los Sistemas Educativos Basados en Web Semántica se reconocen una serie de asuntos o preocupaciones de investigación fundamentales:

- La personalización en la entrega de objetos de aprendizaje.
- El soporte y automatización en la construcción de los metadatos (Anotación) de los objetos de aprendizaje.
- Construcción automática de Objetos de aprendizaje.
- Construcción de repositorios
- Arquitecturas basadas en Servicios Web Semánticos
- El desarrollo de Ontologías para la anotación de los objetos de aprendizaje.
- Búsqueda de objetos de aprendizaje

A continuación se presentan algunos ejemplos relacionados con los asuntos anteriores:

En [41] se presenta una aplicación de la web semántica para apoyar el proceso de creación de los objetos de aprendizaje. Se hace énfasis en la propiedad que tienen los objetos de aprendizaje de ser reutilizados y se define un nuevo enfoque en el cual se crean nuevos objetos desde un conjunto base. Lo interesante de este modelo consiste en que el uso de ontologías para expresar los metadatos LOM permite inferir el conjunto de metadatos que debe tener el objeto derivado. En este mismo trabajo se introducen además una serie de indicadores para describir los objetos de aprendizaje desde una perspectiva semántica. Los indicadores son: Complejidad del contenido, Continuidad Semántica y Desbalance Semántico del Contenido.

En [42] se presenta un modelo por medio del cual se extrae información de documentos semiestructurados para construir automáticamente los metadatos de objetos de aprendizaje.

El sistema LOCO [28] permite realizar análisis sobre objetos de aprendizaje y su uso dentro de plataformas gestoras del aprendizaje permitiendo obtener una retroalimentación útil para los autores de contenido. Todo este proceso se lleva a cabo mediante el uso de diferentes Ontologías de dominio y de perfilado del usuario.

En [43] se presenta un modelo para la personalización de los objetos de aprendizaje para los ambientes virtuales de aprendizaje. Este modelo analiza las interconexiones entre los estilos de aprendizaje de los estudiantes, sus actividades de aprendizaje preferidas, los métodos de enseñanza/aprendizaje relevantes y los tipos de objetos de aprendizaje.

Un *framework* basado en ontologías para la gestión del cambio y evolución de los contenidos de aprendizaje es presentado en [44]. Este *framework* se desarrolla con base en un modelo propio que soporta estructuras de contenido de aprendizaje multi-capa.

Casali et al. [45] Presentan un asistente para la anotación de objetos de aprendizaje los cuales son validados mediante el uso de una Ontología propia desarrollada para el caso la cual establece una capa intermedia que ofrece un vocabulario compartido que permite especificar restricciones y entrega una semántica común para cualquier aplicación que use los metadatos de los objetos.

En lo referente a la construcción asistida o automatizada de objetos de aprendizaje pueden encontrarse ejemplos como los relacionados en [42] y en [46]. En este último se introduce un asistente que apoya a los diseñadores instruccionales en la creación de objetos de aprendizaje partiendo de conjuntos de recursos bien definidos.

Relacionado con la construcción de repositorios para objetos de aprendizaje existen ejemplos como el presentado en [47] el cual ofrece un marco de trabajo para un dominio de conocimiento muy específico.

En los últimos años, el desarrollo de los SEBWS se ha visto influenciado por el enfoque arquitectónico para la construcción de software basado en servicios. La web semántica ofrece su propio modelo para la creación de servicios web semánticos y en lo referente al manejo de objetos de aprendizaje pueden encontrarse ejemplos como el relacionado en [48] el cual propone una arquitectura llamada CROP para construir lo que se denomina *Servicios de Aprendizaje*. Así mismo, en [49] se enuncia otra arquitectura basada en Servicios Web que se enfoca más en el reuso de los objetos de aprendizaje.

En el caso de la búsqueda de objetos de aprendizaje existen múltiples desarrollos al respecto. Este tema se constituye en el foco mismo de esta investigación, por tal motivo se definirá un estado del arte detallado en la siguiente sección.

5.4. Búsqueda de objetos de aprendizaje en una arquitectura de Web Semántica.

Los objetos de aprendizaje normalmente se encuentran organizados en repositorios dentro de la web. Como lo dice Davedzik [24, p. 108] “El propósito de los repositorios de objetos de aprendizaje es permitir a los usuarios un acceso estructurado a los objetos, por ejemplo, la posibilidad de buscar, descubrir, recuperar, navegar e intercambiar objetos”. En la mayoría de los casos, estos repositorios guardan los objetos adaptándose al estándar de metadatos LOM (Ver sección 4.2.2). Son esos metadatos los que permiten recuperar los objetos de acuerdo a las necesidades del aprendiz.

La búsqueda de objetos de aprendizaje se presenta entonces como un campo de investigación en lo relacionado con la construcción de SEBWS. Haciendo un mapeo de las publicaciones recientes en este tema se pueden encontrar diferentes enfoques: Desde el punto de vista de la forma de consulta el enfoque puede ser el de la entrega personalizada de Objetos de Aprendizaje basada en perfil de usuario o búsqueda basada en consultas de usuario. Desde el punto de vista de la ubicación de los objetos, la búsqueda puede ser en repositorios distribuidos (Aquí suele utilizarse la búsqueda mediante agentes) o en repositorios centralizados. Desde el punto de vista de la estructura del objeto, la búsqueda puede ser mediante indexado de metadatos (Information retrieval), Basada en Ontologías o Mixta.

La búsqueda para entrega personalizada de objetos de aprendizaje consiste en proporcionar al estudiante de una manera automática los materiales que mejor se ajustan a su perfil. El perfil del usuario suele modelarse mediante el uso de una ontología de usuario que luego es comparada con la ontología que describe los objetos de aprendizaje para obtener un conjunto de resultados relevantes. Respecto a este tema se pueden citar tres ejemplos de trabajos relacionados:

- En [50] se presenta una arquitectura que utiliza servicios web semánticos basada en WSMO (Web Service Modeling Ontology) y que además usa LOM como el estándar de descripción de los objetos de aprendizaje. El objetivo de esta arquitectura es el de permitir la compartición, descubrimiento, filtrado y recomendación de objetos de aprendizaje relevantes basándose en el perfil del usuario que entregan los proveedores de cursos.
- Una propuesta de Ontología para la búsqueda efectiva de objetos con contenido compartibles (SCO bajo el estándar SCORM) que hace énfasis en objetivos de aprendizaje es presentada en [51].
- En [52] se describe el proyecto BROAD el cual se fundamenta en una red de ontologías llamada BROADnet que cubre diferentes dominios y su objetivo es ofrecer un framework de trabajo para las búsquedas y semánticas y apoyar el proceso de composición de los objetos de aprendizaje.

Muchos desarrollos relacionados con la búsqueda de objetos de aprendizaje tienen el enfoque de la búsqueda basada en consultas de usuario. A diferencia del enfoque en personalización aquí se espera que el usuario tenga una idea básica del objeto que está buscando y de acuerdo a esto formula una consulta que puede ser simplemente una sucesión de palabras clave o un conjunto de reglas de búsqueda más avanzado. Algunos ejemplos:

- [53] Presenta el uso de ontologías para la búsqueda y compara esta aproximación con el enfoque clásico de recuperación de la información.
- [54] presenta un sistema de búsqueda que ofrece un servicio para formular consultas semánticas.

Cuando se tiene en cuenta la ubicación de los objetos en un repositorio centralizado se espera que las búsquedas se ejecuten sobre un mismo repositorio. De entrada, la web semántica soporta este tipo de arquitecturas sin ningún problema, sin embargo, dada la naturaleza de la misma, el mayor provecho se obtiene en ambientes con repositorios distribuidos. Un ejemplo que ilustra una arquitectura de este tipo se puede encontrar en [55] donde se presenta una aproximación mixta basada en palabras claves y ontologías. Usualmente, la búsqueda en repositorios distribuidos se apoya en el uso de agentes. Por ejemplo, en [56] se expone el desarrollo de una arquitectura basada en agentes para lograr la personalización de un plan de documentos de estudios dentro de un LMS.

En la mayoría de los casos consultados el uso de Ontologías es fundamental para lograr la precisión y recuperación deseada así como la sensibilidad al contexto, sin embargo existen aproximaciones a la búsqueda de objetos de aprendizaje que en alguna medida se desligan de este enfoque y utilizan técnicas clásicas de recuperación de la información (Information Retrieval). Un ejemplo de enfoque mixto para la búsqueda de objetos de aprendizaje se puede encontrar en [57] donde se propone una aproximación multicapas para definir los Metadatos LOM

de tal manera que se pueda trabajar con el esquema tradicional y con un esquema basado en ontologías. De esta manera se logra una compatibilidad con repositorios que no implementan tecnologías de web semántica.

En general, puede concluirse que el tema de la búsqueda de objetos de aprendizaje en sistemas educativos basados en la web semántica sigue desarrollándose y en la actualidad muchos de los esfuerzos se centran en el desarrollo de arquitecturas que implementen una búsqueda eficiente y relevante para diferentes contextos.

En la próxima sección se expone el enfoque propuesto para la búsqueda de objetos de aprendizaje en sistemas educativos basados en web semántica.

6. PROPUESTA DE MODELO PARA LA BÚSQUEDA DE OBJETOS DE APRENDIZAJE

6.1. Análisis del problema y propuesta de solución.

Dado el objetivo de este trabajo, se desarrolló un prototipo de motor de búsqueda para objetos de aprendizaje que se vale de una representación ontológica de “Consultas predefinidas” o “Conceptos de Búsqueda” para entregar sus resultados.

Un “Concepto de búsqueda” se entiende como una construcción que se hace en torno a los metadatos de un Objeto de aprendizaje. Por ejemplo, un concepto de búsqueda es “Apto para estudiantes con déficit de atención”. Este concepto de búsqueda se puede construir en torno a metadatos LOM de un objeto de aprendizaje, por ejemplo: (Nivel de Interactividad=Alto, Tipo de recurso=Video, Duración del video < 1 minuto).

En este caso, una aplicación que pueda buscar todos los objetos aptos para estudiantes con déficit de atención debe “entender” antes que nada qué es un objeto apto para estudiantes con déficit de atención y esto último es posible de hacerse, al darle semántica a los datos mediante el uso de Ontologías.

Lo anterior pone como primera condición para el desarrollo de la aplicación la necesidad de contar con algún mecanismo ontológico para representar los metadatos de un objeto de aprendizaje.

El estándar de metadatos que se escoge para profundizar en su representación ontológica es el estándar LOM dado su amplio uso en diferentes repositorios de objetos de aprendizaje. Para el caso del presente desarrollo se utilizó la ontología LOM2OWL definida por García y compañía en [58] (más adelante, en la descripción de la aplicación se explicará el uso que se hizo de esta ontología). LOM2OWL es una ontología expresada en el lenguaje OWL que mapea cada uno de los elementos que constituyen el estándar LOM.

Una vez seleccionada la Ontología con que se expresarán los metadatos para los objetos de aprendizaje es posible comenzar a diseñar la estructura de datos que ampliará la ontología y permitirá definir diferentes “Conceptos de búsqueda”. Esta estructura de datos se construye con base en un patrón de expresión particular (este concepto se explica en detalle en la sección 6.3.2) que permite construir el concepto de búsqueda de “Estilos de aprendizaje” el cual se define con base en una combinación de metadatos LOM.

Al final de este capítulo se describen cada una de las tecnologías y herramientas que fue necesario utilizar para la construcción de la aplicación así como la arquitectura general de la misma. De esta manera se logra cumplir con el objetivo planteado para este trabajo.

6.2. Descripción general para el prototipo de la aplicación OASearch.

Como se enunció anteriormente, la aplicación OASearch se construyó para demostrar la factibilidad de implementar un mecanismo de búsqueda basado en conceptos previamente definidos que pueden ser creados independientemente de la lógica de la aplicación. Esto es posible gracias a la facultad de las ontologías de desacoplar el modelo de conocimiento de la aplicación [6, p. 173] (En este caso, la metadata asociada a los objetos de aprendizaje de la lógica que implementa la búsqueda de los mismos).

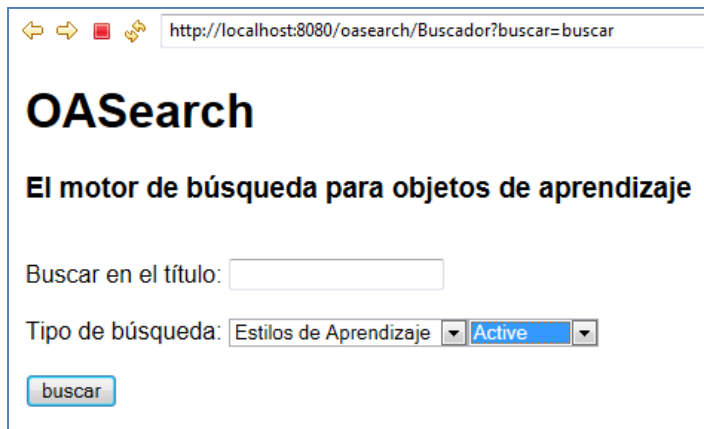


Ilustración 3 Interfaz inicial de OASearch

De esta manera, como se ve en la Ilustración 3, cuando el usuario ingresa a la aplicación tendrá la opción de seleccionar el concepto deseado desde una lista desplegable. Estos conceptos (Que para el ejemplo, son dos: Estilos de aprendizaje, y nivel de interactividad) pueden verse como tipos diferentes de búsqueda. Una vez seleccionado uno de los Conceptos/Tipos de búsqueda se visualiza un nuevo listado con los aspectos específicos relacionados al concepto, al seleccionar cualquiera de ellos podrá ejecutar la búsqueda y como se ve en la Ilustración 4, obtener un conjunto de resultados ordenados de acuerdo a los objetos de aprendizaje que mejor cumplan con el concepto expresado.

En el ejemplo, la consulta es equivalente a decir “Encuéntrame todos los objetos de aprendizaje que han sido creados para alumnos que tienen un estilo de aprendizaje Activo”.

Como se verá más adelante, el concepto “Estilo de aprendizaje” no es más que uno de tantos que pueden ser construidos mediante los metadatos LOM. Se seleccionó este concepto en particular para prototipar la aplicación, basándose en la investigación de mejía y compañía en [59], pero de la misma forma podría

expresarse cualquier concepto que cumpla con las mismas características estructurales (Patrón definido), las cuales se explicitarán más adelante.



Ilustración 4 Resultados de una búsqueda en OASearch

En esencia, lo que el usuario acaba de efectuar es una búsqueda en diferentes metadatos LOM asociados a los objetos de aprendizaje, solo que no tuvo que preocuparse por esto, ya que la estructura de la consulta ya fue expresada en la ontología. Él solo tuvo que preocuparse por seleccionar el concepto de su interés. Un mecanismo como este puede ser fácilmente desarrollado al interior de la aplicación, pero la diferencia aquí, y en esencia, el aporte de esta investigación radica en que los conceptos fueron creados fuera de la aplicación lo cual permite pensar en una dinámica diferente para la construcción de este tipo de motores de búsqueda para objetos de aprendizaje.

La siguiente gráfica muestra cada uno de los componentes de la aplicación OASearch que se describirán en las secciones subsecuentes.

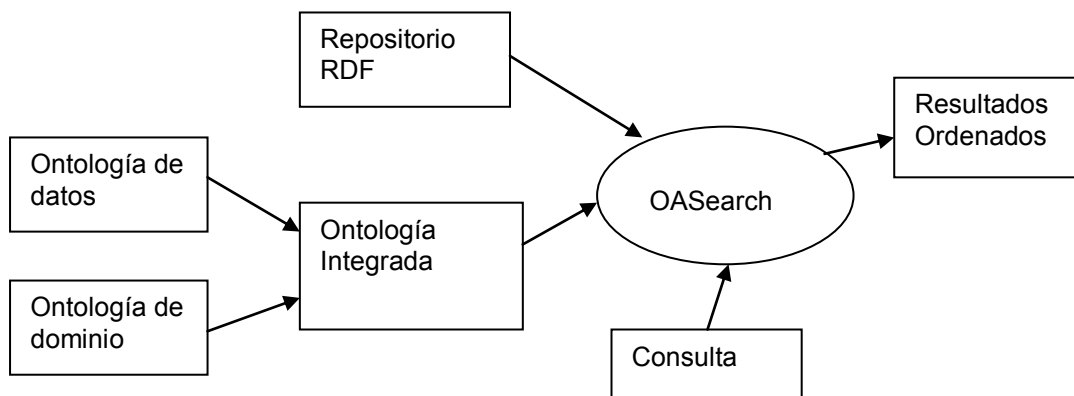


Ilustración 5 Aplicación OASearch. Entradas y salidas de información.

6.3. El modelo de datos

6.3.1. Seleccionando la Ontología para expresar los metadatos LOM

Uno de los principios fundamentales del trabajo con ontologías en la web semántica se refiere al reuso que debe hacerse de las mismas [6, p. 175]. En esencia, se promueve la reutilización de ontologías que ya han sido definidas previamente para un dominio específico de conocimiento. Es en este sentido que se tomó la decisión de usar la ontología LOM2OWL disponible en [60, p. 2] para modelar los datos de los objetos de aprendizaje para la aplicación OASearch.

Una descripción completa de la ontología lom2owl se puede encontrar en [58]

Las clases y relaciones básicas que son utilizadas por la ontología LOM2OWL se muestran en la siguiente ilustración.

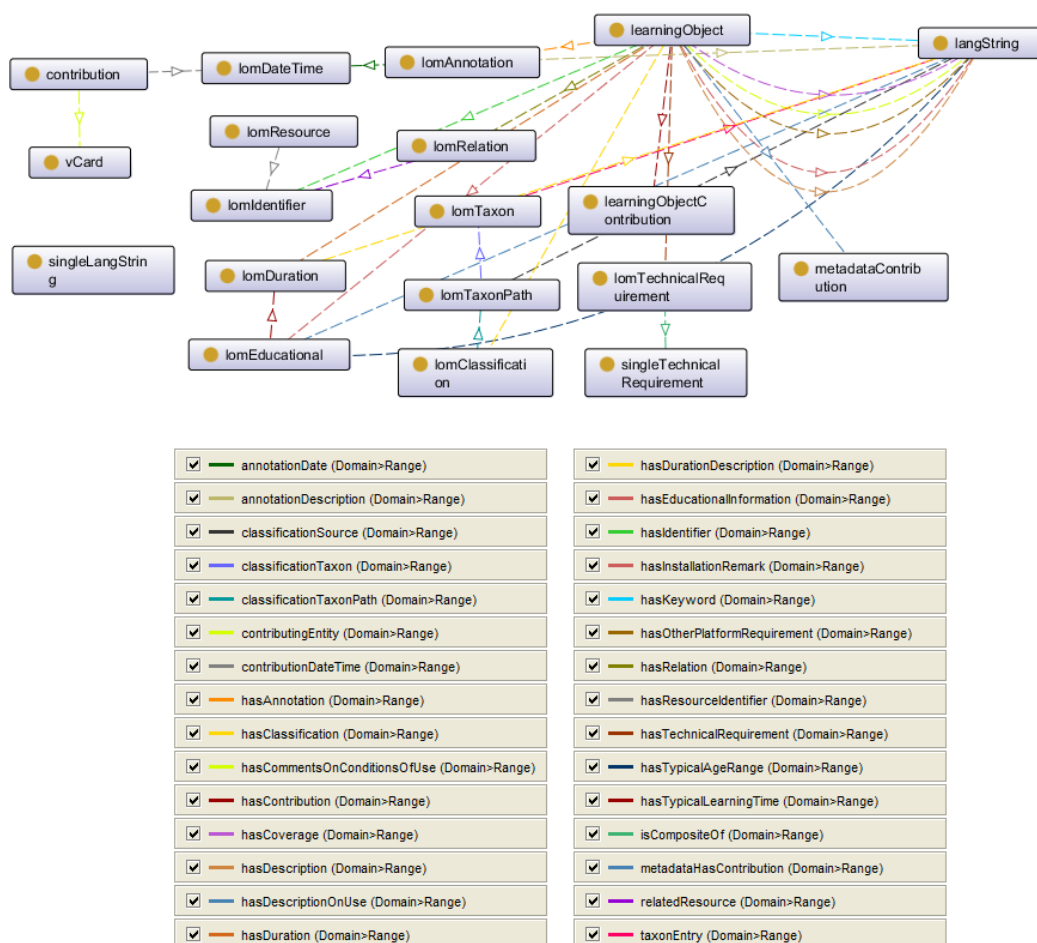


Ilustración 6 Descripción general de la ontología lom2owl.

6.3.2. Definiendo el patrón de expresión para el tipo de búsqueda

Una vez definida la ontología a usar, el siguiente paso es definir el patrón que debe seguirse para expresar los conceptos de búsqueda. “El patrón de expresión” es en esencia, la manera en que los usuarios externos deben definir los conceptos utilizando OWL de tal manera que la aplicación pueda incluirlos dentro del listado disponible para la búsqueda de objetos de aprendizaje. Esta definición de “El Patrón de expresión” equivale a la construcción de una Ontología de Aplicación [6, p. 361] . Si bien, esta Ontología de Aplicación podría definirse como un objeto OWL independiente y luego alinearse (Integrarse) con la Ontología de Dominio (lom2owl) utilizando la aplicación, para efectos del experimento se integraron ambas ontologías en un solo objeto que se denominó la “Ontología Integrada”

La edición de la Ontología Integrada se realizó utilizando la herramienta Protege [61] la cual facilita la creación de todas las estructuras OWL además de las entidades de prueba y la inclusión de razonadores. Cada uno de estos elementos fue necesario para poder expresar, mediante la Ontología Integrada, el tipo de búsqueda “Estilos de aprendizaje” que corresponde al patrón de expresión que se denominará “Clasificación por Aspectos”.

Como se dijo anteriormente, para definir un primer patrón de expresión que soportaría la aplicación y al mismo tiempo crear un tipo específico de búsqueda que sirviera como ejemplo, se partió del trabajo de Mejía y compañía en el cual se define un modelo que permite inferir el Estilo de Aprendizaje al que obedece un Objeto de aprendizaje basado en una combinación de sus metadatos LOM.

Específicamente, este modelo hace una clasificación de los objetos de aprendizaje en los diferentes estilos de acuerdo a los tipos de recursos que estos contienen. Los tipos de recursos determinan el estilo de aprendizaje y esta relación se expresa en una tabla que surge de un cuestionario a diversos expertos temáticos en el que se les pregunta qué tanta importancia le da un estudiante que tiene un determinado estilo de aprendizaje al hecho de que ese objeto de aprendizaje contenga un tipo de recurso determinado. Los resultados se expresan en una tabla de la siguiente manera:

| Resource Types | Learning Styles | | | | | | | |
|-------------------|-----------------|------------|------------|---------|---------------|--------|--------|--------|
| | Processing | | Perception | | Understanding | | Input | |
| | Active | Reflective | Intuitive | Sensory | Sequential | Global | Visual | Verbal |
| Exercise | G | G | VG | G | G | I | G | VG |
| Simulation | VG | I | I | VG | I | G | VG | I |
| Questionnaire | G | I | I | I | G | I | I | I |
| Diagram | I | G | G | G | G | VG | VG | I |
| Figure | I | G | G | G | G | VG | VG | I |
| Graph | I | G | G | G | G | VG | VG | I |
| Index | I | I | I | I | G | VG | I | G |
| Slide | I | G | G | G | G | VG | VG | G |
| Table | I | I | I | G | G | G | G | G |
| Narrative text | G | VG | VG | I | I | I | I | VG |
| Exam | G | I | I | I | G | I | I | I |
| Experiment | VG | G | G | VG | G | I | VG | I |
| Problem statement | VG | G | G | VG | I | I | G | VG |
| Self assessment | G | I | I | I | G | I | I | I |
| Lecture | I | VG | VG | I | I | I | I | VG |

Tabla 2 Relación entre el tipo de recurso LOM y las dimensiones de estilos de aprendizaje.

Básicamente, lo que se expresa en esta tabla es el conocimiento que tienen los expertos temáticos sobre cómo los alumnos responden a un objeto de aprendizaje de acuerdo a su estilo de aprendizaje. De esta manera, se puede definir luego, que el estilo de aprendizaje al que mejor se adapta un objeto de aprendizaje está en función de los tipos de recursos que contiene.

El problema que surge ahora es cómo expresar este conocimiento en una ontología OWL de tal manera que al repositorio RDF asociado se le puedan hacer consultas del tipo “¿Cuáles objetos corresponden a un tipo de aprendizaje Activo y en qué medida?”, esto respecto al ejemplo específico de los estilos de aprendizaje, pero generalizando se puede definir un Patrón de Expresión y en consecuencia, un tipo de búsqueda en el que se puedan hacer preguntas del tipo “¿Cuáles objetos corresponden a una clasificación particular y en qué medida?” Así como en este caso la clasificación es el Estilo de Aprendizaje, se puede pensar en otros Tipos de clasificación como por ejemplo “Nivel de atención”, “Contexto geográfico”, etc.

Tenemos entonces un Patrón de expresión que se podría denominar “Clasificación por Aspectos” o “Matricial” el cuál será el que definirá el alcance de la aplicación y será suficiente para demostrar la tesis planteada en la introducción. La aplicación podría desarrollarse con la capacidad de manejar otros patrones de expresión, pero eso sería objeto de otra investigación.

6.3.3. Construcción de la ontología de dominio para el patrón de expresión.

Para la construcción de una ontología de dominio que permita modelar el patrón de expresión matricial expuesto en la sección anterior se siguió una aproximación en la cual se definieron ontologías independientes que luego fueron integradas.

El primer paso consiste en modelar la manera en que la aplicación obtendrá los conceptos de búsqueda y las categorías relacionadas con cada concepto de búsqueda. Como se ve en la ilustración ¹, se definen tres clases: La búsqueda en sí, los conceptos de búsqueda que estarán disponibles y las categorías que están asociadas a conceptos de búsqueda.

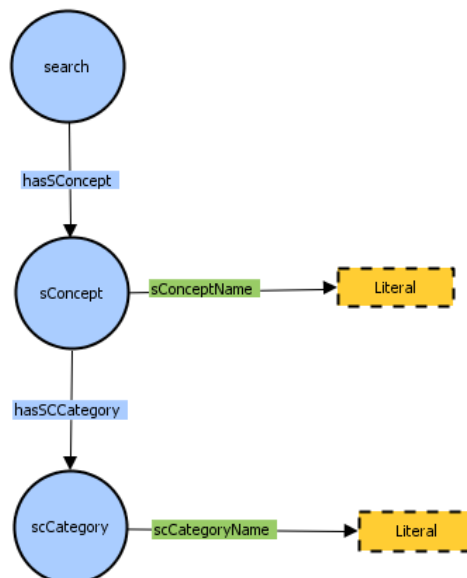


Ilustración 7 Ontología del modelo de la búsqueda

Para nombrar los conceptos y las categorías se utilizan Propiedades de Datos (Data Properties) que apuntan a literales de tipo String.

Con una representación de este tipo, es posible que la aplicación de búsqueda obtenga un listado de los conceptos disponibles y las categorías asociadas a cada concepto, pero estos conceptos dependen de la información que pueda obtenerse de otra ontología que defina un modelo del estudiante. La ontología que modela al estudiante y las características asociadas a él es la siguiente:

¹ Se utiliza aquí la notación VOWL definida en [62]

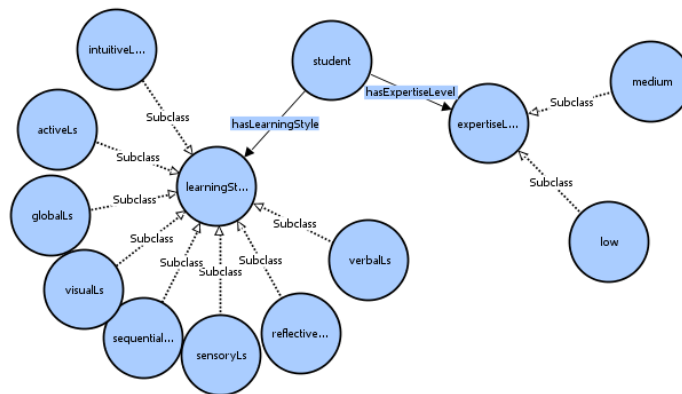


Ilustración 8 Ontología del modelo del estudiante

Como se ve, esta ontología permite modelar el hecho de que un estudiante posee una característica llamada “Estilo de Aprendizaje” la cual tiene como subclases los diferentes tipos de estilos de aprendizaje como se definieron en la sección 6.3.2. Además, a manera de ejemplo se presenta otra característica que ha sido llamada “Nivel de experiencia”. Así como se han construido estas dos características el administrador de la ontología podría definir cualquier cantidad de características adicionales.

El problema que surge ahora es cómo alinear estas dos ontologías para lograr que cada característica del estudiante se vea reflejada en la ontología de búsqueda. Para esto, se propone el uso de una relación que asocie la unión de cada característica con los conceptos de búsqueda.

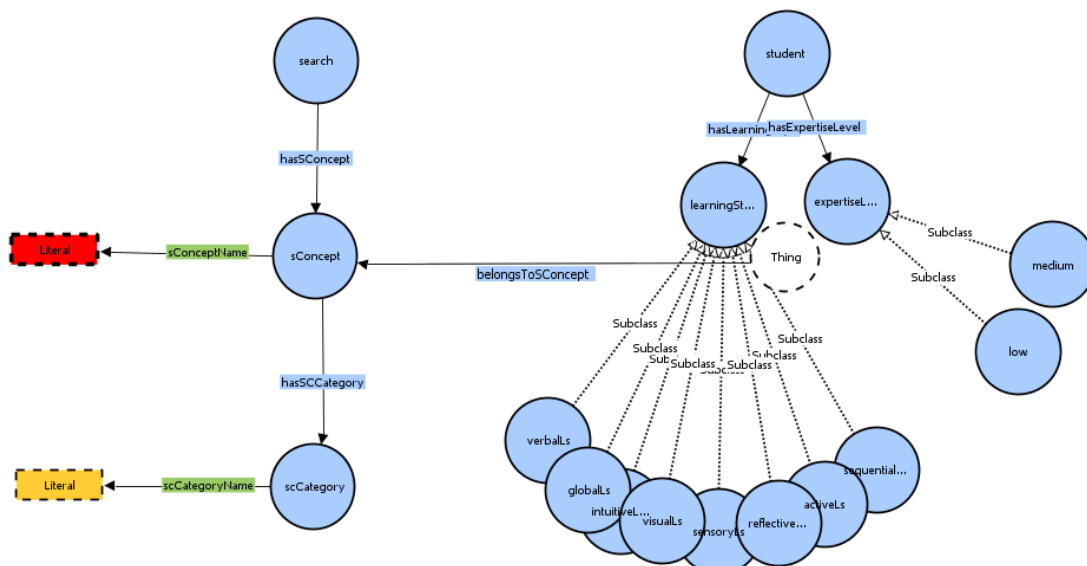


Ilustración 9 Ontología integrada del modelo del estudiante y la búsqueda

Esta relación, llamada “belongsToSConcept” parte de la unión de las características existentes del estudiante (En el momento learningStyle y experienceLevel) y finaliza en la clase sConcept de la ontología de búsqueda. El

punto de partida equivale a lo que se conoce como una Clase No Nombrada y se define en Protege en el dominio de la relación de la siguiente forma:

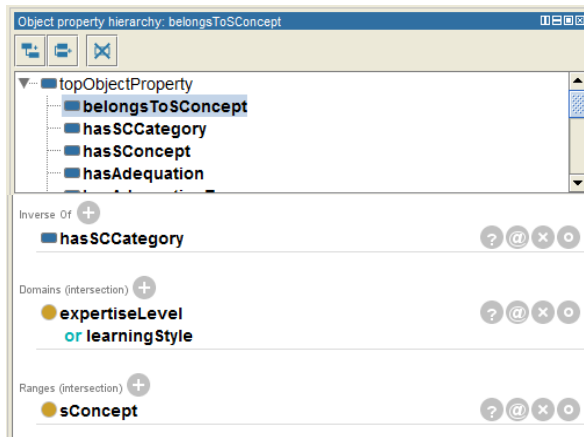


Ilustración 10 Definición en Protege de la clase no nombrada para el dominio de la relación belongsToSConcept.

Además de definir el dominio, se define la relación belongsToSconcept como la inversa de la relación hasSCategory logrando de esta manera que en el momento de ejecutar un razonador, las características del estudiante se equiparen a las categorías de búsqueda.

Luego de tener definida la integración entre el modelo del estudiante y el modelo de la búsqueda es preciso definir la manera en que se representarán los recursos de aprendizaje en relación con los objetos de aprendizaje. La siguiente imagen muestra dos clases que están contenidas en la ontología base lom2owl: La clase del objeto de aprendizaje (learningObject) y la clase que refleja el conjunto de metadatos de tipo educacional (lomEducational) definida en el estándar LOM.

Adicionalmente, se define la clase learningResource la cual se constituye en una modificación a la ontología lom2owl en la medida que está utiliza propiedades de datos (literales) para la representación de los recursos en vez de clases. Este cambio se realiza con el objetivo de poder generar instancias para cada recurso de aprendizaje que luego será relacionado con las características del estudiante (ej. El estilo de aprendizaje) a través de una representación de la matriz de pesos. Así mismo, se definen una clase por cada recurso de aprendizaje con el objeto de lograr una mayor especificidad que puede ser útil para la manipulación de la ontología y la posibilidad de realizar inferencias sobre la misma.

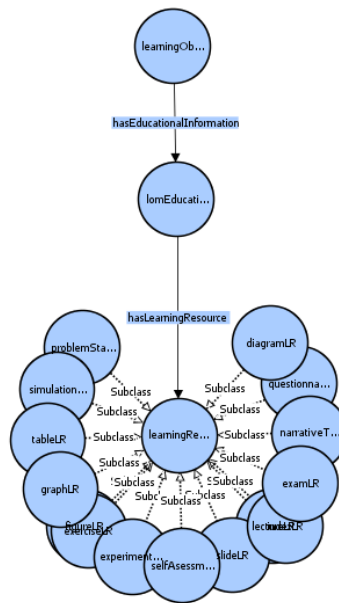


Ilustración 11 Ontología para el modelo del objeto de aprendizaje derivada de lom2owl

El siguiente paso es definir el modelo matricial que permitirá expresar la relación de peso que existe entre una categoría de búsqueda (clase scCategory) y los recursos de aprendizaje que tenga un objeto de aprendizaje. (clase learningResource). La siguiente imagen muestra esta ontología:

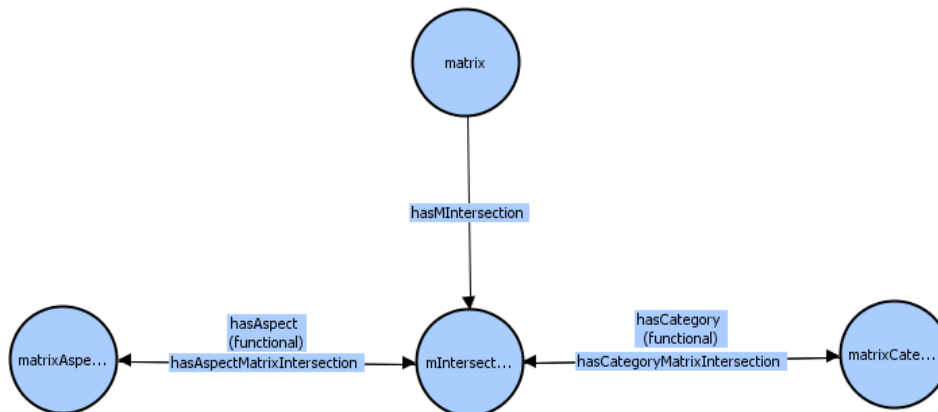


Ilustración 12 Ontología del modelo de la matriz

Como se puede ver, el elemento central de esta ontología es la clase mInterseccion que sirve para modelar la intersección entre las categorías (columnas) y aspectos (filas) de la matriz. Las categorías se representan utilizando la clase matrixCategory y los aspectos con la clase (matrixAspect). Una intersección se relaciona con los aspectos y categorías mediante las relaciones de objeto hasCategory y hasAspect respectivamente. Se definen estas dos relaciones como funcionales en la medida en que una instancia de la clase mInterseccion solo puede apuntar a una categoría y una instancia de la clase mInterseccion solo puede apuntar a un aspecto.

Adicionalmente se definen las relaciones `hasAspectMatrixIntersection` y `hasCategoryMatrixIntersection` como las relaciones inversas de `hasAspect` y `hasCategory` respectivamente. Esta definición permitirá facilitar la creación de las instancias por parte del administrador de la ontología y así mismo facilitará el recorrido del grafo en las consultas SPARQL que más tarde tendrá que realizar la aplicación.

Teniendo definidas todas la ontologías anteriores es necesario ahora realizar la integración de las mismas. Esta integración se puede hacer simplemente definiendo como equivalentes, por un lado, las clases `matrixAspect` y `learningResource`, y por el otro lado, las clases `matrixCategory` y `sCategory`. De esta manera, la ontología para representar la matriz se convierte en el articulador entre el dominio del estudiante y el dominio del objeto de aprendizaje.

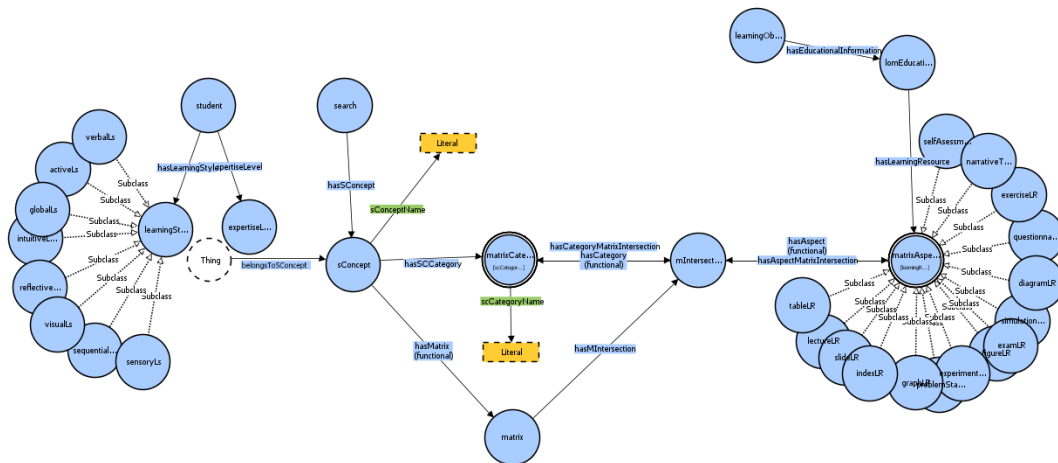


Ilustración 13 Ontología con todos los modelos integrados

Como se puede observar en la ilustración, las clases equiparadas aparecen en un círculo doble. Adicionalmente se define una relación funcional entre la clase `sConcept` y la clase `matrix`, lo cual establece que por cada concepto de búsqueda definido solo existe una matriz para definir las relaciones entre las categorías de búsqueda y los recursos de aprendizaje.

El último paso en la definición de la ontología de dominio consiste en establecer algún mecanismo que permita expresar el hecho de que “Un objeto de aprendizaje se ajusta en cierta medida a un estudiante con una característica específica”. En términos menos generales, poder establecer una representación que permita responder preguntas como: ¿En qué medida se ajusta un objeto de aprendizaje a un estudiante con estilo de aprendizaje activo? o ¿En qué medida se ajusta un objeto de aprendizaje a un estudiante con estilo de aprendizaje global? o ¿En qué medida se ajusta un objeto de aprendizaje a un estudiante con nivel de experticia bajo?

6.4. Construcción de la aplicación.

6.4.1. Descripción general de la aplicación

Una vez definida la ontología integrada es posible desarrollar una aplicación que haga uso de esta para entregar al usuario las opciones de búsqueda.

La aplicación OASearch fue construida en lenguaje Java como una aplicación web utilizando Java Server Pages (JSP). Se eligió este lenguaje debido a que JENA, una de las librerías más avanzadas para el manejo de Ontologías y Documentos RDF está disponible bajo este lenguaje.

Como se ve en la ilustración 18, la aplicación tiene como entrada todos los datos RDF de los objetos de aprendizaje, la ontología integrada y la consulta de usuario. La salida de esta es un conjunto ordenado de metadatos relacionados con los objetos de aprendizaje que mejor se ajustan a la consulta.

Las etapas de ejecución de la aplicación son las siguientes:

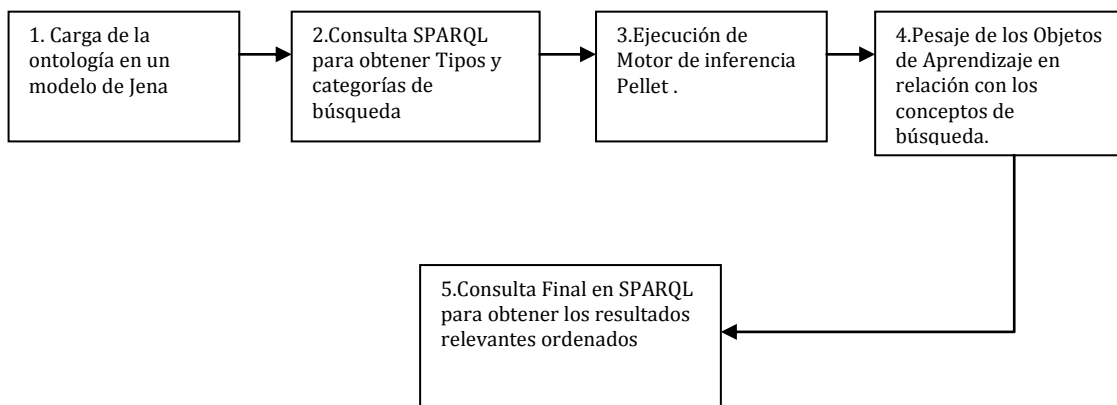


Ilustración 15 Etapas de ejecución de la aplicación OASearch.

En la primera etapa la ontología integrada es cargada en la estructura Model definida en la librería Jena. Para este caso, la ontología fue creada con la herramienta Prótegé. En el modelo de Jena se carga tanto la ontología como la información de los objetos de aprendizaje expresada en entidades RDF.

Con la información en el Modelo se extraen los tipos de búsqueda que están disponibles así como los aspectos relacionados con cada uno de estos tipos. Para extraer de la ontología esta información en esta etapa así como la información de etapas posteriores se utilizan consultas expresadas en SPARQL.

Una vez extraídos los tipos y aspectos de búsqueda es posible continuar con la ejecución del razonador. Para este caso se utilizó Pellet, un razonador ampliamente conocido y que se provee a través de una librería para Java. Al

ejecutar el razonador, se genera una nueva clasificación de las entidades que está disponible en un nuevo modelo de Jena llamado el Modelo Inferido. El modelo inferido se calcula aquí, para efectos de la prueba de concepto, como parte del proceso de consulta de usuario pero en un esquema de alto rendimiento, este proceso debe ejecutarse aparte pues con grandes cantidades de información, es decir, con un repositorio de OA muy grande, tomaría mucho tiempo.

Teniendo el modelo inferido es posible realizar, para cada objeto, la sumatoria de todos los valores asociados a cada intersección de la matriz relacionada con las diferentes categorías de búsqueda. Cada valor es asociado luego como una sentencia RDF a cada objeto de aprendizaje en un nuevo modelo llamado el Modelo Ponderado.

Al final se hace una consulta al Modelo Ponderado de acuerdo a la categoría seleccionada por el usuario. Los resultados de esta consulta son presentados al usuario a través de la interfaz. Los objetos se presentan ordenados de acuerdo al peso. El de mayor peso, es decir, el objeto que mejor responde a la categoría de búsqueda se presenta de primero.

En este punto, el usuario ha recibido la información de una manera más precisa y acorde al dominio de búsqueda que ha planteado en la consulta.

6.4.2. Aspectos fundamentales de codificación de la aplicación

Retomando las 5 fases expuestas en la sección anterior, se procede ahora a describir los aspectos fundamentales relacionados con el código desarrollado en cada una de ellas.

En la fase 1 se hace la carga de la ontología en un modelo de Jena. La función que realiza esta tarea es `addDataFromFile()`

```
private void addDataFromFile(String realPath) throws IOException{
    ModelMaker modelMaker = ModelFactory.createFileModelMaker("temp/filesave.owl");
    Model modeltmp = modelMaker.createDefaultModel();
    _modelMem = ModelFactory.createOntologyModel(OntModelSpec.OWL_DL_MEM, modeltmp);
    InputStream inLomInstance = FileManager.get().open(realPath+"\\WEB-INF\\Ontologies\\LOM2OWLentitiesLearningStyle5.owl");
    _modelMem.read(inLomInstance, defaultNameSpace);
    inLomInstance.close();
}
```

Un modelo es básicamente la representación en memoria de una ontología OWL junto con todas las tripletas RDF. Según lo definido en la librería Jena, una de las formas de generarlo es obteniendo la información desde un flujo serializado, en este caso, el archivo generado por protege en formato OWL/XML.

Se instancia primero un objeto de tipo `modelMaker` utilizando el método `createFileModelMaker` de la clase estática `ModelFactory`. Este método recibe como parámetro una ruta para escribir un archivo temporal con la información del modelo. Luego, el objeto `modelMaker` sirve para instanciar un modelo temporal genérico definido por defecto en Jena utilizando el método `createDefaultModel`.

Este modelo genérico es utilizado luego para crear el modelo específico al pasarlo como parámetro en el método createOntologyModel de la clase estática ModelFactory. Así mismo, se pasa también al método una constante que especifica el tipo de modelo que se quiere crear, en este caso, un modelo bajo la especificación OWL-DL. El resultado de ejecutar este método es devuelto a una variable de clase de tipo OntoModel.

Para poder cargar la ontología en el modelo se abre un flujo de datos que apunta al archivo de la ontología y luego se ejecuta el método read() del modelo que recibe el objeto del flujo de datos. Es en este punto que el modelo tiene toda la información de la ontología y puede comenzarse a manipular dentro del programa. Por último se cierra el flujo de datos.

En la fase 2, para obtener los tipos de búsqueda y las categorías respectivas se utilizan dos funciones getTipos() y getAspectos()

Para obtener los tipos de búsqueda, dentro de la función getTipos(), se ejecuta la siguiente consulta SPARQL

```
SELECT ?oa ?cName WHERE { ?oa lom2owl:sConceptName ?cName. FILTER (langMatches( lang(?stname), 'ES'))}
```

Esta consulta devolverá todos los objetos existentes de la clase sConcept junto con el valor de la propiedad de datos sConceptName.

Para obtener las categorías o aspectos de búsqueda relacionados con el tipo escogido se utiliza la siguiente consulta SPARQL dentro de la función getAspectos(tipoSeleccionado)

```
SELECT ?caName WHERE { ?concept lom2owl:sConceptName '"+tipoSeleccionado+"'@es. ?concept lom2owl:hasSCCategory ?category. ?category lom2owl:scCategoryName ?caName. FILTER(langMatches(lang(?caName), 'ES'))}
```

Como se ve, la consulta obtiene todos los nombres de las categorías relacionadas con los conceptos de búsqueda que tienen el nombre que se pasa a la función en la variable tipoSeleccionado.

Es de anotar que las dos consultas anteriores utilizan un filtro para obtener todos los resultados en el idioma español. En la ontología es posible definir diferentes idiomas para los valores literales, pero para efectos de esta prueba de concepto solo se utiliza el idioma español.

En la fase 3, se ejecuta el razonador pellet mediante el siguiente código:

```
Reasoner reasoner = PelletReasonerFactory.theInstance().create();
InfModel inferredModel = ModelFactory.createInfModel(reasoner, (Model)_modelMem);
```

Se crea un objeto de tipo razonador (Reasoner) que recibe el resultado de ejecutar el método de creación en la clase estática PelletReasonerFactory. Luego se utiliza la clase ModelFactory de la librería Jena para crear un nuevo modelo el cual es el resultado de aplicar el proceso de inferencia sobre el modelo OWL cargado desde archivo. (Ver descripción de la fase 1)

Una vez se crea el modelo, Jena ejecuta el proceso de clasificación. Este proceso toma un tiempo considerable para ejecutarse con la ontología desarrollada. En proyectos futuros es pertinente considerar mejoras en el desempeño del proceso de razonamiento, pero para efectos de esta prueba de concepto el tiempo es aceptable.

De otro lado, es importante tener también en cuenta que dentro de esta prueba de concepto, cada vez que se ejecuta el programa, se ejecuta el razonador lo cual no es eficiente en uso de recursos. En un ambiente de producción es necesario desligar este proceso del proceso de consulta de la ontología mediante un disparador que lo ejecutaría cada cierto tiempo.

Continuando con la fase 4 se hace el pesaje de los objetos de aprendizaje en relación con las categorías de búsqueda. Esta tarea se ejecuta mediante la función `addAdequationValues()`.

Para realizar el pesaje de los objetos se utilizan tres ciclos anidados. A continuación se describen las consultas SPARQL que se ejecutan en cada ciclo:

- a. Inicialmente se buscan toda las categorías existentes que corresponden al concepto/tipo de búsqueda seleccionado por el usuario mediante la siguiente consulta de SPARQL:

```
SELECT ?matrixCategory ?matrixCategoryName WHERE {?matrixCategory
lom2owl:belongsToConcept ?concept. ?concept lom2owl:sConceptName ?conceptName.
?matrixCategory lom2owl:scCategoryName ?matrixCategoryName.
FILTER(langMatches(lang(?matrixCategoryName), 'ES') &&
?concept='"+this.selectedType+"@es) }
```

El resultado de esta consulta se almacena en un objeto de solución llamado `soln`

- b. Ciclo de primer nivel. Por cada categoría obtenida en el paso anterior se obtienen los objetos de aprendizaje que tienen relacionado algún recurso de aprendizaje. Se utiliza la siguiente consulta:

```
SELECT DISTINCT ?learningObject WHERE {?learningObject
lom2owl:hasEducationalInformation ?lomEducational. ?lomEducational
lom2owl:hasLearningResource ?learningResource}
```

El resultado de esta consulta se almacena en un objeto de solución llamado `soln2`.

- c. Ciclo de segundo nivel. Por cada objeto obtenido en el paso anterior se obtiene la suma de los valores asociados a las intersecciones de la matriz que tienen como categoría la correspondiente actual al ciclo de primer nivel.

```
SELECT (SUM(?intersectionValue) AS ?categoryValue) WHERE {?learningObject
lom2owl:hasEducationalInformation ?lomEducational. ?lomEducational
lom2owl:hasLearningResource ?learningResource. ?learningResource
lom2owl:hasAspectMatrixIntersection ?matrixIntersection. ?matrixIntersection
lom2owl:hasCategory ?category. ?matrixIntersection lom2owl:hasIntersectionValue
?intersectionValue. ?category lom2owl:belongsToCharacteristic ?characteristic
```



```
FILTER(?category = <"?soln.get("?matrixCategory)+>" && ?learningObject = <"?soln2.get("?learningObject)+>")}
```

El resultado de esta consulta se almacena en un objeto de solución llamado soln3.

- d. Ciclo de tercer nivel. Por cada resultado del paso anterior se agrega la información a un array que luego se adiciona a un arrayList

```
String[] values = new String[3];
values[0] = soln3.get("?categoryValue").toString();
values[1] = soln.get("?matrixCategory").toString();
values[2] = soln2.get("?learningObject").toString();
result.add(values);
```

Por último, dentro de la misma fase 4, se genera un ciclo que recorre todo el arrayList para agregar las siguientes tripletas en un nuevo modelo llamado weightedModel. Es de anotar que estas modificaciones no se hicieron al interior del ciclo de tercer nivel debido a que un modelo de Jena no puede ser modificado al mismo tiempo que es leído. Si hace esto, se genera un error de concurrencia en el hilo lanzado por Pellet para la ejecución de las inferencias sobre el modelo.

El código que modifica el modelo es el siguiente:

```
int i=0;
for (String[] value : result){
    //Crea y adiciona las tripletas para el objeto actual y la categoría actual
    Resource subject=weightedModel.getResource(value[2]);
    Property predicate = ResourceFactory.createProperty(
        "http://www.cc.uah.es/ie/ont/lom2owl#hasAdequation" );
    Resource object=ResourceFactory.createResource(value[2]+"adq"+i++);
    Statement addStatement = ResourceFactory.createStatement (subject, predicate,
        object);
    weightedModel.add(addStatement);
    System.out.println("adicionando tripleta: "+value[2]+" "+"
        lom2owl:hasAdequation"+" lom2owl:"+value[2]+"adq"+i);

    //Crea y adiciona la tripleta que define la relación entre la clase --
    //adequateFor y scCategory
    Resource subject2=weightedModel.getResource(object.getURI());
    Property predicate2 = ResourceFactory.createProperty (
        "http://www.cc.uah.es/ie/ont/lom2owl#hasAdequationTo");
    Resource object2=weightedModel.getResource(value[1]);
    Statement addStatement2=ResourceFactory.createStatement(subject2, predicate2,
        object2);
    weightedModel.add(addStatement2);
    System.out.println("adicionando tripleta: "+object.getURI()+"
        "+"lom2owl:hasAdequation"+" "+value[1]);

    //Crea y adiciona la tripleta para asignar el valor al objeto a través de la
    //relación hasAdequationValue
    Resource subject3=weightedModel.getResource(object.getURI());
    Property predicate3 = ResourceFactory.createProperty (
        "http://www.cc.uah.es/ie/ont/lom2owl#hasAdequationValue");
    Resource object3=ResourceFactory.createResource(value[0]);
    Statement addStatement3=ResourceFactory.createStatement(subject3, predicate3,
        object3);
    weightedModel.add(addStatement3);
    System.out.println("adicionando tripleta: "+object.getURI()+"
        "+"lom2owl:hasAdequationValue"+" "+value[0]);
}
```

Como se puede ver en el código anterior para agregar una tripleta a un modelo, se utilizan los objetos de tipo Resource para definir las entidades y los objetos de tipo Property para definir las propiedades. Una vez instanciados los objetos que conforman la tripleta, se genera la misma mediante el método createStatement de la clase estática ResourceFactory provista en la librería de Jena. El objeto devuelto por este método es cargado luego al modelo a través de la función add.

En la fase 5 se finaliza el proceso mediante una consulta al modelo ponderado teniendo en cuenta la categoría de búsqueda elegida por el usuario. Esto se lleva a cabo mediante la función getResult(aspectoSeleccionado) que ejecuta la siguiente consulta de SPARQL:

```
SELECT ?weight ?subject ?literal ?literal2 WHERE { ?subject lom2owl:isLocated ?literal.  
?subject lom2owl:hasTitle ?object2. ?object2 lom2owl:includesSingleLangString ?object3.  
?object3 lom2owl:hasValue ?literal2. ?subject lom2owl:hasAdequation ?adequationObject.  
?adequationObject lom2owl:hasAdequationValue ?weight. ?adequationObject  
lom2owl:hasAdequationTo ?category. ?category lom2owl:scCategoryName ?categoryName  
FILTER(?categoryName = '"+aspectoSeleccionado+"'@es)} ORDER BY DESC(?weight)"
```

7. CONCLUSIONES Y TRABAJO FUTURO

Con este trabajo se demuestra cómo el uso de ontologías permite inferir propiedades de los objetos de aprendizaje como resultado de las relaciones al interior de sus metadatos. Teniendo en cuenta esto, es posible definir búsquedas predeterminadas cuando se conoce a priori cómo los metadatos construyen una propiedad determinada.

Las ontologías no solo permiten construir las propiedades derivadas de los metadatos sino que se convierten en un mecanismo para desacoplar la definición para la construcción de nuevas propiedades de la lógica de la aplicación. Con esto es posible que a futuro se pueda implementar una interfaz con la que usuarios con un perfil especial construyan tipos de búsqueda que pueden ser usados en la aplicación.

Con el uso de un mecanismo como el propuesto en este documento es posible implementar bancos de objetos de aprendizaje cuya información puede ser consumida por otras aplicaciones.

A futuro es necesario indagar por la manera en que las búsquedas de objetos de aprendizaje mediante un perfilado predefinido de consultas, como se propone en este documento, mejoran la precisión de los resultados obtenidos por el usuario. También es pertinente estudiar la manera en que debe ser desarrollada una interfaz que permita ampliar la ontología de dominio para poder expresar múltiples tipos de búsqueda abstrayendo el manejo del lenguaje OWL.

De otro lado es importante profundizar en la manera más eficiente de realizar las inferencias en el modelo, probablemente a través de una tarea programada. Esto es equivalente a la generación del índice en un sistema clásico de recuperación de la información.

Si bien, en este trabajo se construyó lo que se define como La Ontología Integrada como una unidad que incluye la ontología de aplicación y la ontología de Dominio, es conveniente separar estas dos últimas en entidades diferentes. Esto requerirá agregar un proceso de alineación ontológica a la aplicación pero facilitará la definición de nuevos tipos de búsqueda mediante la ampliación de la ontología de Dominio.

Este trabajo se constituye en una prueba de la viabilidad de desarrollar un sistema hipermedial adaptativo que pueda entregar a los estudiantes contenidos adecuados a su situación de aprendizaje. Se propone a futuro la construcción de un modelo del estudiante a través del cual se puedan inferir diferentes propiedades, como por ejemplo, su estilo de aprendizaje. Con esta información y

el uso de las ontologías, la integración con la aplicación sería fácil de implementar para lograr una recomendación automática de contenidos.

BIBLIOGRAFÍA

- [1] D. Wiley, "Connecting learning objects to instructional design theory: A definition, a metaphor and a taxonomy," in *Instructional Use of Learning Objects*, Agency for Instructional Technology and Association for Educational Communications and Technology, 2002.
- [2] "LOM Standard. Draft Standard for Learning Object Metadata." IEEE Computer Society.
- [3] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," 2001. [Online]. Available: <http://www.scientificamerican.com/article.cfm?id=the-semantic-web>. [Accessed: 20-Aug-2010].
- [4] G. Antoniou and F. van Harmelen, *A Semantic Web Primer, 2nd Edition*, 2nd ed. The MIT Press, 2008.
- [5] T. Berners-Lee, "Semantic Web roadmap," *Semantic Web Road map*, Sep-1998. [Online]. Available: <http://www.w3.org/DesignIssues/Semantic.html>. [Accessed: 24-Aug-2010].
- [6] J. Hebel, M. Fisher, R. Blace, and A. Perez-Lopez, *Semantic Web Programming*. Wiley, 2009.
- [7] F. Manola and E. Miller, "RDF Primer," 10-Feb-2004. [Online]. Available: <http://www.w3.org/TR/rdf-primer/>. [Accessed: 25-Aug-2010].
- [8] T. Segaran, C. Evans, and J. Taylor, *Programming the Semantic Web*, 1st ed. O'Reilly Media, 2009.
- [9] D. Beckett, "RDF/XML Syntax Specification (Revised)," 10-Feb-2004. [Online]. Available: <http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/>. [Accessed: 29-Aug-2010].
- [10] D. Beckett and T. Berners-Lee, "Turtle - Terse RDF Triple Language," 14-Jan-2008. [Online]. Available: <http://www.w3.org/TeamSubmission/turtle/>. [Accessed: 29-Aug-2010].
- [11] J. Grant and D. Beckett, "RDF Test Cases," 10-Feb-2004. [Online]. Available: <http://www.w3.org/TR/rdf-testcases/#ntriples>. [Accessed: 29-Aug-2010].
- [12] G. Klyne and J. J. Carrol, "Resource Description Framework (RDF): Concepts and Abstract Syntax," 10-Feb-2004. [Online]. Available: <http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/>. [Accessed: 29-Aug-2010].
- [13] P. Hayes, "RDF Semantics," 10-Feb-2004. [Online]. Available: <http://www.w3.org/TR/2004/REC-rdf-mt-20040210/>. [Accessed: 29-Aug-2010].
- [14] D. Brickley and R. V. Guha, "RDF Vocabulary Description Language 1.0: RDF Schema," 10-Feb-2004. [Online]. Available: <http://www.w3.org/TR/2004/REC-rdf-schema-20040210/>. [Accessed: 29-Aug-2010].
- [15] S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider, and L. A. Stein, "OWL Web Ontology Language Reference," 10-Feb-2004. [Online]. Available: <http://www.w3.org/TR/owl-ref/>. [Accessed: 29-Aug-2010].

- [16] W. O. Working Group, "OWL 2 Web Ontology Language Document Overview," 27-Nov-2009. [Online]. Available: <http://www.w3.org/TR/owl2-overview/>. [Accessed: 29-Aug-2010].
- [17] A. Chiappe, Y. Segovia, and H. Rincón, "Toward an instructional design model based on learning objects," *Education Tech Research Dev*, vol. 55, no. 6, pp. 671–681, Dec. 2007.
- [18] "IMS Meta-data Best Practice Guide for IEEE 1484.12.1-2002 Standard for Learning Object Metadata." IMS Global Learning Consortium, Inc.
- [19] "Dublin Core Metadata Element Set, Version 1.1." Dublin Core Metadata Initiative, 14-Jun-2012.
- [20] N. Hernandez, J. Mothe, B. Ralalason, B. Ramamonjisoa, and P. Stolf, "A Model to Represent the Facets of Learning Object," *Interdisciplinary Journal of E-Learning and Learning Objects*, vol. 4, no. 1, pp. 65–82, Jan. 2008.
- [21] A. Kasimati and E. Zamani, "Education and Learning in the Semantic Web," 2011, pp. 338–344.
- [22] M. Čarapina, O. Staničić, and B. Nožica, "THE SEMANTIC WEB TECHNOLOGIES AND THEIR INTEGRATION WITHIN E-LEARNING SYSTEMS: AN OVERVIEW," *ICERI2013 Proceedings*, pp. 3796–3801, 2013.
- [23] V. Devedzic, "Education and the Semantic Web," *Int. J. Artif. Intell. Ed.*, vol. 14, no. 2, pp. 165–191, Abril 2004.
- [24] V. Devedzic, *Semantic Web and Education*, Edición: 2006. New York: Springer, 2006.
- [25] D. Keegan, "Distance education technology for the new millennium: Compressed video teaching," 1995.
- [26] Paulsen, *Online education and learning management systems - Global elearning in an scandinavian perspective*. Oslo: NKI Forlaget.
- [27] P. De Bra, L. Aroyo, and V. Chepegin, "The next big thing: Adaptive web-based systems," *Journal of Digital Information*, vol. 5, no. 1, 2004.
- [28] J. Jovanovic, V. Devedz'ic, D. Gasevic, M. Hatala, T. Eap, G. Richards, and C. Brooks, "Using semantic web technologies to analyze learning content," *Internet Computing, IEEE*, vol. 11, no. 5, pp. 45–53, 2007.
- [29] T. Anderson and D. Whitelock, "The Educational Semantic Web: Visioning and Practicing the Future of Education," *Journal of Interactive Media in Education*, vol. 2004, no. 1, 2004.
- [30] M. Vargas-Vera and M. Lytras, "Personalized Learning Using Ontologies and Semantic Web Technologies," in *Proceedings of the 1st World Summit on The Knowledge Society: Emerging Technologies and Information Systems for the Knowledge Society*, Berlin, Heidelberg, 2008, pp. 177–186.
- [31] H. Barros, A. Silva, E. Costa, I. I. Bittencourt, O. Holanda, and L. Sales, "Steps, techniques, and technologies for the development of intelligent applications based on Semantic Web Services: A case study in e-learning systems," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 8, pp. 1355–1367, Diciembre 2011.
- [32] M. Barla, M. Bieliková, and et al, "Rule-Based User Characteristics Acquisition from LOGS WITH SEMANTICS FOR PERSONALIZED WEB-BASED SYSTEMS," in *COMPUTING AND INFORMATICS*, 2009.
- [33] A. Gladun, J. Rogushina, F. Garcí'a-Sanchez, R. Martínez-Béjar, and J. T. Fernández-Breis, "An application of intelligent techniques and semantic

- web technologies in e-learning environments,” *Expert Systems with Applications*, vol. 36, no. 2, Part 1, pp. 1922–1931, Mar. 2009.
- [34] S.-L. Huang and C.-W. Yang, “Designing a semantic bliki system to support different types of knowledge and adaptive learning,” *Computers & Education*, vol. 53, no. 3, pp. 701–712, Nov. 2009.
- [35] G. Vega-Gorgojo, M. L. Bote-Lorenzo, J. I. Asensio-Pérez, E. Gómez-Sánchez, Y. A. Dimitriadis, and I. M. Jorrín-Abellán, “Semantic search of tools for collaborative learning with the Ontoolsearch system,” *Computers & Education*, vol. 54, no. 4, pp. 835–848, May 2010.
- [36] A. Ruiz-Calleja, G. Vega-Gorgojo, J. I. Asensio-Pérez, E. Gómez-Sánchez, M. L. Bote-Lorenzo, and C. Alario-Hoyos, “SEEK-AT-WD: A social-semantic infrastructure to sustain educational ICT tool descriptions in the web of data,” *Educational Technology and Society*, vol. 17, no. 2, pp. 321–332, 2014.
- [37] S. Isotani, R. Mizoguchi, S. Isotani, O. M. Capeli, N. Isotani, A. R. P. L. de Albuquerque, I. I. Bittencourt, and P. Jaques, “A Semantic Web-based authoring tool to facilitate the planning of collaborative learning scenarios compliant with learning theories,” *Computers & Education*, vol. 63, pp. 267–284, Abril 2013.
- [38] N. Rubens, D. Kaplan, and T. Okamoto, “E-Learning 3.0: Anyone, Anywhere, Anytime, and AI,” in *New Horizons in Web Based Learning*, D. K. W. Chiu, M. Wang, E. Popescu, Q. Li, and R. Lau, Eds. Springer Berlin Heidelberg, 2014, pp. 171–180.
- [39] P. Miranda, P. Isaias, and C. J. Costa, *From information systems to e-Learning 3.0 systems’s critical success factors: A framework proposal*, vol. 8523 LNCS. 2014.
- [40] P. Mohan and C. Brooks, “Learning objects on the semantic web,” in *Advanced Learning Technologies, 2003. Proceedings. The 3rd IEEE International Conference on*, 2003, pp. 195–199.
- [41] R. Farhat, B. Defude, and M. Jemni, “Towards a Better Understanding of Learning Objects Content,” 2011, pp. 536–540.
- [42] S. Ye and T.-S. Chua, “Learning object models from semistructured web documents,” *Knowledge and Data Engineering, IEEE Transactions on*, vol. 18, no. 3, pp. 334–349, 2006.
- [43] E. Kurilovas, S. Kubilinskiene, and V. Dagiene, “Web 3.0 – Based personalisation of learning objects in virtual learning environments,” *Computers in Human Behavior*, vol. 30, pp. 654–662, Enero 2014.
- [44] C. Pahl, M. Javed, Y. M. Abgaz, C. Pahl, M. Javed, and Y. M. Abgaz, “Ontology Evolution for Learning Content Management Systems,” presented at the World Conference on Educational Multimedia, Hypermedia and Telecommunications, 2013, vol. 2013, pp. 1431–1436.
- [45] A. Casali, C. Deco, A. Romano, G. Tome, A. Casali, C. Deco, A. Romano, and G. Tome, “An Assistant for Loading Learning Object Metadata: An Ontology Based Approach,” presented at the Proceedings of the Informing Science and Information Technology Education Conference, 2013, vol. 2013, pp. 77–87.
- [46] K. Verbert, X. Ochoa, M. Derntl, M. Wolpers, A. Pardo, and E. Duval, “Semi-automatic assembly of learning resources,” *Computers & Education*, vol. 59, no. 4, pp. 1257–1272, Diciembre 2012.

- [47] P. Raju and V. Ahmed, "Enabling technologies for developing next-generation learning object repository for construction," *Automation in Construction*, vol. 22, pp. 247–257, Mar. 2012.
- [48] C. Hartonas and E. Gana, "Learning Objects and Learning Services in the Semantic Web," 2008, pp. 584–586.
- [49] J.-M. L. Cobo, S. Arroyo, M.-A. Sicilia, and A. O. SAE, "A Semantic Web Service Architecture for Learning Object Repositories."
- [50] M. Hanafy and M. Fakhry, "A Proposed Architecture for Learning Object Selection and Discovery Based on WSMO," 2008, pp. 10–14.
- [51] A. Kardan and S. Zahmatkesh, "A Proposed Ontology for Effective Searching of Sharable Content Objects Emphasizing on Learning Objectives," 2009, pp. 1140–1145.
- [52] T. N. Teixeira, F. Campos, R. Braga, N. Santos, and E. Mattos, "BROAD Project: Semantic Search and Application of Learning Objects," *IEEE Technology and Engineering Education (ITEE)*, vol. 7, 2012.
- [53] M.-C. Lee, K. H. Tsai, and T. I. Wang, "An ontological approach for semantic-aware learning object retrieval," in *Advanced Learning Technologies, 2006. Sixth International Conference on*, 2006, pp. 208–210.
- [54] M. A. Neri, "Ontology-based learning objects retrieval," in *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2005*, E-Learn 2005–World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education, 2005, pp. 2793–2798.
- [55] K. Votis, C. Alexakos, K. Giotopoulos, and S. Likothanassis, "An Ontological Service Oriented and Web Searching Mechanism in Distributed e-learning Repositories," in *Sixth International Conference on Advanced Learning Technologies, 2006*, 2006, pp. 211–215.
- [56] D. Çelik, A. Elci, and E. Elverici, "Finding Suitable Course Material through a Semantic Search Agent for Learning Management Systems of Distance Education," 2011, pp. 386–391.
- [57] I.-C. Hsu and others, "Intelligent Discovery for Learning Objects Using Semantic Web Technologies," *Educational Technology & Society*, vol. 15, no. 1, pp. 298–312, 2012.
- [58] A. M. Fermoso García, "Una ontología en OWL para la representación semántica de objetos de aprendizaje."
- [59] C. Mejía, S. Baldiris, S. Gómez, and R. Fabregat, "Adaptation process to deliver content based on user learning styles," in *International Conference of Education, Research and Innovation (ISBN:978-84-6125091-2*, 2008.
- [60] A. M. Fermoso García, "lom2owl ontology." .
- [61] "Protégé." [Online]. Available: <http://protege.stanford.edu/>.
- [62] S. Negru, S. Lhoman, and F. Haag, "VOWL: Visual Notation for OWL Ontologies." .
- [63] P. Hayes and C. Welty, "Defining N-ary Relations on the Semantic Web." W3C, 12-Apr-2006.

ÍNDICE

- Agentes pedagógicos, 40
- algoritmos de tableau, 31
- anotación de objetos de aprendizaje, 45
- Anotaciones, 29
- aprendizaje adaptado, 37
- árboles de decisión, 30
- base de conocimiento, 30
- bases de conocimiento, 21
- bases de datos relacionales, 6, 23
- buscadores de ontologías, 33
- Búsqueda, 34
- búsqueda basada en consultas de usuario, 46
- búsqueda de objetos de aprendizaje, 45
- búsqueda para entrega personalizada de objetos de aprendizaje, 46
- búsqueda semántica, 20
- búsquedas de información en la web, 21
- búsquedas personalizadas, 39
- captura del contexto, 39
- Clases, 29
- compartición de modelos de usuarios, 39
- Consulta, 34
- Contenido Educativo*, 40, 42
- Desacople, 32
- descripción de recursos, 24
- documentos semiestructurados, 44
- Dublin Core, 35
- educación a distancia, 37
- educación basada en la web, 36
- e-Learning, 37
- Encabezado de la ontología, 29
- encadenamiento hacia adelante, 31
- encadenamiento hacia atrás, 31
- formato Turtle, 26
- framework, 30
- frameworks, 29
- función de clasificación, 9
- grafo, 24
- grafo RDF, 30
- IMS Object Metadata, 35
- índice, 8
- individuos, 29
- inferencia basada en reglas, 31
- integración de dominios, 33
- intercambio de datos, 23
- intercambio de la semántica, 23
- Intercambio de la sintaxis, 23
- interoperabilidad, 23
- IRI, 24
- Jena, 29
- lenguaje extensible de marcado, 23
- lógica de primer orden, 22
- lógica descriptiva, 22
- LOM, 35, 45
- marco de descripción de recursos, 24
- materiales de aprendizaje, 37
- mecanismos de marcado semántico, 42
- metadatos, 23, 42
- metadatos de objetos de aprendizaje, 44
- metadatos educativos, 36
- modelado del usuario, 39, 41
- modelo de capas, 23
- modelo de conocimiento, 29, 32
- modelo de información, 23, 24
- modelos clásicos para la recuperación de la información, 9
- modelos de objetos, 23
- modelos de representación, 21
- modelos relacionales, 23
- motor de inferencias, 30
- motor de razonamiento, 22
- motor de reglas, 30
- motores de búsqueda basados en palabras clave, 21
- motores de búsquedas, 22
- Navegación, 34
- N-Triples, 26
- Objetivos de Aprendizaje, 43
- objeto de aprendizaje, 35, 42
- objetos de aprendizaje, 39, 45
- objetos serializados, 23

- ontología, 28
- Ontología, 27
- ontología de base, 33
- ontología de dominio, 41
- ontologías, 20, 27, 40
- ontologías de personalización, 41
- ontologías fundacionales, 33
- operaciones textuales*, 6
- OWL, 27, 28, 30, 32
- OWL-DL, 31
- OWL-EL, 31
- OWL-Lite, 31
- OWL-QL, 31
- OWL-RL, 31
- perfiles OWL, 31
- personalización, 44
- Personalización, 40
- personalización del aprendizaje, 43
- Propiedades*, 29
- Ranking*, 9
- RDF, 23, 24, 26
- RDFS, 27
- recuperación de datos, 6
- recuperación de la información, 6
- registros, 33
- Reglas de inferencia, 21
- relevancia, 8
- repositorios, 33
- repositorios RDF, 30
- Resultados de Aprendizaje., 43
- reúso, 32
- SCORM, 35
- SEBWS, 39, 43
- semántica, 26
- Servicios de Aprendizaje, 45
- servicios web semánticos, 41
- sintaxis, 26
- sistema de recuperación de la información, 7
- sistemas educativos adaptativos hipermedia, 37
- sistemas educativos basados en web semántica, 39
- Sistemas Inteligentes de Tutoría, 37
- sistemas multiagente, 41
- SPARQL, 35
- stopwords, 6
- supuesto del mundo abierto, 28
- SWRL, 32
- taxonomía, 27
- teoría de grafos, 22
- Tipos de datos, 29
- tripleas, 25
- URIs, 24
- URL, 24
- vista lógica*, 6, 8
- vocabulario, 27
- web semántica, 20, 21, 25
- Web semántica, 21
- Web Semántica Educativa*, 40
- World Wide Web Consortium, 26
- XML, 23