UNIVERSIDAD CATÓLICA ANDRÉS BELLO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA INFORMÁTICA

# Desarrollo de una ontología y un buscador utilizando tecnologías de Web Semántica

TRABAJO ESPECIAL DE GRADO

presentado ante la

UNIVERSIDAD CATOLICA ANDRES BELLO

como parte de los requisitos para optar al título de

INGENIERO EN INFORMÁTICA

REALIZADO POR **Israel Fermín Montilla**

PROFESOR GUIA **Ph.D: Wilmer Pereira**

FECHA Caracas, septiembre de 2011

# AGRADECIMIENTOS

...

A mi tutor, Dios, Familia, escuela, jaladita de bola a la universidad y de más cursilerías que uno pone aquí

...

*Israel Fermín Montilla*

# SINOPSIS

Toda la paja en una página.

# INTRODUCCIÓN

Copiar y pegar lo mismo de la propuesta

CAPÍTULO 1

# PROBLEMA

## 1.1. Planteamiento del Problema

La World Wide Web (WWW), ha cambiado radicalmente la manera como las personas se comunican entre sí, la forma como la información se distribuye y como se diseminan los mensajes y los modelos de negocio de muchas empresas alrededor del mundo[1]. Basta con revisar las páginas web de las grandes empresas a nivel nacional e internacional, como Polar, Apple Computer o Microsoft para darse cuenta de que ya no se enfocan sólo en tener presencia en radio y televisión, sino también en las distintas redes sociales que han surgido a través de la Web 2.0.

Ciertamente Internet, desde su aparición en 1975, ha evolucionado, ya los sitios web no son páginas con texto estático e imágenes en las que sólo un subconjunto de los usuarios es capaz de publicar contenido, mientras que otros sólo leen y reciben información. Ahora se cuenta con páginas y aplicaciones web complejas, en las que todos los usuarios están en capacidad de generar y publicar contenido. Esto se conoce como Conocimiento Colectivo [2], de esta manera,

con cada ves más usuarios publicando contenido en la web, la cantidad de información ha crecido exponencialmente, ya para 2009, el estimado de usuarios de Internet en América Latina y el Caribe era de 175,8 millones de personas aproximadamente [3]. Lo que implica un enorme volumen y tráfico de datos en la WWW pues todos esos usuarios intercambian, publican y consultan información permanentemente.

Todos estos usuarios poseen necesidades de información esperando a ser satisfechas. La información para satisfacer a todos y cada uno de los usuarios, está disponible en línea, pero resulta difícil acceder a ella si no se sabe dónde se encuentra, es decir, si no se conoce su URL. Es por ello que fueron creados los buscadores como Yahoo, Google, Altavista y Bing, por mencionar algunos de los más populares todos estos funcionan buscando las palabras claves que proporciona el usuario en los documentos que ha indexado, es decir, son buscadores basados en palabras clave. De esta manera, si el usuario introduce, por ejemplo, “Internet”, el buscador retornará todos los documentos conocidos que contienen esa palabra, bien sea en el título o en el cuerpo del texto. Pero cuando la búsqueda involucra más de una palabra, las cosas pueden complicarse un poco, por ejemplo: si el usuario introduce “Internet de Venezuela” los buscadores basados en palabras clave omiten los conectores pues aparecen en todos los documentos y sitios web, por lo que la búsqueda sería “Internet Venezuela” y el resultado de la búsqueda contendría todos los recursos en los que aparezcan ambas palabras o alguna de las dos.

Pero qué ocurre cuando el usuario desea realizar búsquedas más especializadas, por ejemplo “Aerolíneas que viajan a Valera”, probablemente el buscador nos de lo que estamos buscando, pero requiere un trabajo adicional de búsqueda por parte del usuario pues los resultados obtenidos contendrían páginas de “Aerolíneas”, páginas que contienen la palabra “viajan”, y noticias sobre “Valera”, localidad del Estado Trujillo, o sobre personas con el apellido “Valera”. Esta resulta ser la principal desventaja de los buscadores actuales basados en palabras clave o “key words", cuando se desea hacer una búsqueda basada en un tópico específico o enmarcada en un contexto dado, los resultados de la búsqueda no son muy cercanos a lo que el usuario desea buscar pues los buscadores no son capaces de interpretar el significado detrás de las palabras clave que componen la consulta realizada.

Casos como el anteriormente expuesto se ven a diario entre los estudiantes de carreras afines a las Ciencias de la Computación, como lo es la Ingeniería Informática. Muchas veces se trata de buscar información que aparece en la web bajo otro nombre o, por ejemplo, al buscar conocimientos complejos como “cálculo de rutas óptimas”, se consiguen resultados muy avanzados para el nivel de experiencia que se tiene en esa área, y no se desglosa los resultados en tópicos que son necesarios para dominar el objeto de la búsqueda, por ejemplo, resultados sobre el manejo de matrices de adyacencia y resultados sobre algoritmos ya existentes que tienen dicha utilidad como Dijkstra y Bellman-Ford. De esta manera, el estudiante puede tener una guía para poder atacar el tópico de su interés.

Por todo lo dicho con anterioridad, se plantea el desarrollo de una aplicación con capacidad semántica que sirva como herramienta de consulta en materia de Ciencias de la Computación e Ingeniería Informática.

Resulta de interés resaltar que todo lo relativo a la Web Semántica y sus estándares y tecnologías, se encuentran actualmente en definición y en proceso de desarrollo, por ello, se debe tener presente que la Web Semántica se encuentra en un nivel experimental, de igual manera que el desarrollo de este trabajo.

## 1.2. Objetivo General

- Desarrollar una ontología y un prototipo de buscador sobre dicha ontología utilizando tecnologías y estándares de Web Semántica.

## 1.3. Objetivos Específicos

1. Construir una Base de Conocimientos.
2. Evaluar de manera cualitativa distintos Motores de Inferencia y seleccionar el que mejor se adapte a las necesidades del proyecto.
3. Definir las reglas de inferencia a ser aplicadas sobre el vocabulario creado.
4. Construir una infraestructura básica de búsqueda para realizar las pruebas necesarias sobre la ontología.
5. Desarrollar las interfaces necesarias para realizar consultas y mostrar los resultados.
6. Implementar un procedimiento semiautomático para la evolución de la ontología.

## 1.4. Alcance

Desarrollar una ontología y un buscador con capacidad semántica para realizar búsquedas utilizando ontología desarrollada.

## 1.5. Limitaciones

- La ontología se realizará únicamente sobre tópicos referentes al área de Ciencias de la Computación e Ingeniería en Informática.

- Debido a la enorme cantidad de temas existentes dentro del área de Ciencias de la Computación e Ingeniería Informática, se cubrirá el dominio de temas en extensión.
- En general, se profundizará a tres (3) niveles en todos los temas cubiertos.
- Dado que el vocabulario a ser utilizado no es estándar sino que es producto del presente trabajo, la realización de búsquedas masivas en Internet no será un tema pertinente a este trabajo.

## 1.6. Justificación

En la última década (2000 – 2010), la cantidad de usuarios de Internet a nivel mundial se ha incrementado en un 446 % [3], esto da una idea acerca de la importancia que tiene la World Wide Web como herramienta de comunicación, búsqueda e intercambio de información entre las personas en la actualidad. La WWW, ha cambiado la manera como las personas interactúan entre si, extendiendo la forma también como se presta servicio de educación, mediante plataformas de e-learning. La web ha evolucionado y se ha vuelto tan confiable que en la última década la tendencia de muchas grandes empresas ha sido "subir todo a la nube", es decir, aprovechar plataformas de Cloud Computing para tener sus servicios accesibles en cualquier momento y en cualquier parte del mundo.

Dada la importancia de la WWW a nivel mundial, día a día se ven iniciativas para hacer más placentera la experiencia de los usuarios en la red, así como muevas tecnologías y estándares para optimizar y mejorar su funcionamiento, al punto que gracias a tecnologías como el Really Simple Syndication (RSS), no es necesario visitar constantemente un sitio web para leer sus actualizaciones u obtener el último episodio de nuestro Podcast favorito. Todo esto no sólo le hace la vida más fácil al usuario, sino que optimiza el funcionamiento de la web.

La Web Semántica es una iniciativa del World Wide Web Consorcium (W3C) que pretende “extender la web, dotándola con un mayor significado para que cualquier usuario pueda conseguir respuesta a sus preguntas en Internet de manera rápida y sencilla"[4]. La Web Semántica viene a resolver el problema de acceso a la información en internet, ya que al añadirle semántica a la Internet, se puede buscar, transferir y compartir información de una manera más sencilla, permitiéndole al usuario, delegar estas tareas en software de manera confiable.

Por todo lo anteriormente mencionado, cualquier proyecto o investigación destinado al mejoramiento y optimización de la WWW y, más aún, dentro del campo de la Web Semántica, tiene pertinencia en el presente, ya que es un tema novedoso y que es objeto de investigación en muchas Universidades importantes alrededor del mundo. Desde el año 2001, cuando Tim Berners-Lee escribió un artículo para la revista Scientific American describiendo las posibilidades de la

Web Semántica, muchos grupos de investigadores han tomado este tema como tópico de investigación. Actualmente, en Venezuela, se llevan a cabo numerosas investigaciones en este campo, siendo la Universidad Simón Bolívar (USB) la que presenta mayor actividad, teniendo un grupo de investigadores dedicados a este tema, entre los que destacan la Prof. María Esther Vidal y la Prof. Soraya Abad-Mota, ambas especialistas y ponentes internacionales en Web Semántica.

En la Universidad Católica Andrés Bello (UCAB), no existe ninguna investigación en esta área, lo que da aún más pertinencia a un Trabajo de Grado en esta área, pues permitiría a la UCAB incursionar en este campo, contribuyendo al desarrollo de una tecnología de software nueva y a nivel internacional y abriría las puertas a una nueva línea de investigación dentro de la Escuela de Ingeniería Informática de la UCAB, permitiendo la realización de nuevos trabajos más avanzados acerca de este tema.

Para este proyecto, no sólo será necesario implementar estándares ya creados pues, como se explicará más adelante, casi la totalidad de los estándares y tecnologías que construyen la Web Semántica, aún cuando ya están siendo utilizados, se encuentran en desarrollo todavía. Será necesario también elaborar un procedimiento de traducción de la consulta del usuario a lenguaje de consultas sobre RDF (SPARQL), este vendría siendo, si se quiere, el elemento de mayor dificultad para lograr el buen funcionamiento y desempeño del prototipo de buscador que se desea desarrollar. Además, el desarrollo de un proceso de inclusión de nuevos conceptos a la ontología añade un factor más de dificultad que hace que sean necesarias las habilidades y conocimientos de un Ingeniero en Informática para resolver problemas que no sólo tienen que ver con programación o codificación de algoritmos (que es una de las capacidades del Ingeniero en Informática), sino con la definición y construcción de los mismos para resolver un problema computacional específico de manera eficaz y eficiente.

Por otra parte, el desarrollo de una ontología acerca en materia de las Ciencias de la Computación e Ingeniería Informática, permitiría organizar el conocimiento disponible en esos campos y, además, disponer de un repositorio dónde consultar dicho conocimiento, con el valor agregado de que las búsquedas se realizarían con sentido semántico, lo que da una mayor seguridad al usuario de que los resultados del buscador tienen coherencia con la consulta realizada

y la información deseada. El producto de este trabajo, podría implantarse como una herramienta de consulta electrónica para los estudiantes de carreras afines a la Ingeniería Informática en la UCAB. Posteriormente, la herramienta, podría ser extendida a otros temas mediante otros trabajos similares e incluso, podría portarse a otras carreras o a otras universidades.

CAPÍTULO 2

# MARCO TEÓRICO

Para poder adentrarnos en el tema de la Web Semántica, resulta necesario estudiar y comprender los antecedentes y los eventos que poco a poco han ido llevando a la evolución de la Internet a la herramienta que tenemos hoy día y que, poco a poco también, irán llevando la Web actual, la Web 2.0, a su evolución natural: la Web 3.0, es decir, la Web Semántica.

Si actualmente nos encontramos en la llamada Web 2.0, tuvo que, en algún momento, existir una Web 1.0. Esta primera versión, por así llamarla, sólo contaba con portales que únicamente exponían contenido. La Web 1.0 estaba destinada a la publicación de contenidos corporativos, no daba la posibilidad de participación abierta a los usuarios, no existían espacios para la publicación de contenido por parte de los usuarios y, estos usuarios, eran importantes en tanto fueran consumidores [5], estas, según Ricardo Casanova, eran algunas de las características principales de la Web 1.0 que, además, resultan ser grandes limitaciones. En la Web 1.0, únicamente personas especializadas eran capaces de crear contenido, por ello, sólo las grandes empresas podían disponer de un espacio en la red, el resto de los usuarios únicamente podían recibir (consumir) el mensaje o el contenido que era publicado, sin la posibilidad de participar

en la generación y actualización del mismo.

El término Web 2.0 aparece a mediados del año 2004, y fue creciendo paulatinamente hasta convertirse en portada de los principales seminarios y congresos en la navidad de 2006. Según O'Reilly (citado por Pardo y Cobo, 2007), principal promotor de la Web 2.0, algunos de sus principios básicos son: La web como plataforma, el aprovechamiento de la Inteligencia Colectiva, la gestión de Bases de Datos como competencia básica, el software no limitado a un solo dispositivo y brindar experiencias enriquecedoras al usuario [2]. Todo esto quiere decir, que ahora la interacción de los usuarios es bidireccional, sigue siendo un subgrupo técnico de estos usuarios el que crea los portales, pero ahora, la gran diferencia es que todos los usuarios son capaces de generar contenido en dichos portales. El usuario ya no es un simple consumidor, sino que pasa a ser consumidor-generador de contenido en la web. Pasamos de una red pasiva, de páginas estáticas, a una red activa de páginas dinámicas que interactúan con bases de datos para almacenar y actualizar su contenido que, a su vez, es generado por los usuarios que hacen vida dentro del portal.

La evolución es un factor constante en todas las áreas y tecnologías de las Ciencias de la Computación y la Ingeniería Informática, Internet no es la excepción a esta regla, por lo es fácil deducir que la Web 2.0 no es más que el estado actual y transitorio de esta tecnología que, eventualmente, evolucionará a una Web 3.0, cuyas características están siendo definidas por el W3C bajo el marco de la Web Semántica.

## 2.1. La Web Semántica

El W3C define a la Web Semántica como "una Web extendida, dotada de mayor significado en la que cualquier usuario de Internet, podrá encontrar respuestas a sus preguntas de forma más rápida y sencilla gracias a que la información está mejor definida"[4]. Al dotar a la Web de un mayor significado y, por consiguiente, de mayor semántica, es posible obtener solución a problemas comunes de búsqueda de información, gracias a la implementación de una infraes-

tructura y lenguajes comunes de búsqueda, destinados a resolver dichos problemas, realizando dichas búsquedas tomando en cuenta el contexto y el significado real de la consulta.

Una vez definida la Web Semántica a nivel conceptual, conviene examinar brevemente cómo funciona. Supongamos que la Web tiene la capacidad de construir una base de conocimiento sobre las preferencias de los usuarios y que, a través de una combinación entre su conocimiento y la información disponible en la Internet, sea capaz de atender de forma exacta las demandas de información por parte de los usuarios, por ejemplo, reserva de hoteles, vuelos, médicos o libros.

Si esto ocurriese en la vida real, el usuario, al intentar encontrar "todos los vuelos a Praga para mañana por la mañana", obtendría resultados exactos sobre su búsqueda. Desafortunadamente la realidad es otra, los buscadores actuales mostrarían resultados acerca de "Praga" como localidad turística, noticias de sucesos ocurridos en "Praga", quizás páginas de periódicos locales, foros sobre "Praga", en resumen, sitios que contienen las palabras que conforman la consulta hecha por el usuario. Estos resultados son inexactos y, por sí solos, no satisfacen las necesidades de información del usuario, es necesario un segundo filtro, examinar uno a uno los resultados y extraer manualmente la información que resulte interesante. Por otro lado, un buscador con capacidad semántica, mostraría información más exacta a lo que se desea obtener. La ubicación geográfica desde la cual se envía la consulta sería detectada de manera automática, sin necesidad de indicar el punto de partida, además, elementos de la oración como "mañana" adquirirían significado, siendo convertidos en un día concreto calculado en función de un "hoy". De igual manera ocurriría con el segundo "mañana", el cual sería interpretado como un momento determinado del día. Todo ello a través de una Web en la que los datos, dejan de ser sólo datos y pasan a ser información llena de significado.

La manera como se procesará la información no sólo será en términos de entrada y salida de parámetros de búsqueda, sino en términos de su semántica, apoyándose en una infraestructura basada en meta-datos. Vale acotar, que no se trata de Inteligencia Artificial, sino de dar a las máquinas la capacidad de resolver problemas bien definidos, a través de operaciones bien definidas, ejecutadas sobre datos existentes bien definidos a través de meta-datos.

Dentro de la Web Semántica convergen una serie de estándares y tecnologías, muchas de ellas aún en proceso de definición y desarrollo por parte de investigadores alrededor del mundo y el W3C, las cuales se explican brevemente a continuación:

## 2.2. Ontologías

Según el W3C, "una Ontología define los términos utilizados para describir y representar conocimiento en un área" [10]. Las Ontologías son utilizadas por personas y aplicaciones que necesitan compartir conocimiento acerca de un área específica, es por ello que las ontologías son relativas a un tópico o área en especial, además, incluyen información acerca de los conceptos básicos y las relaciones entre ellos, esta información puede ser utilizada por las computadoras. De esta manera, una Ontología codifica el conocimiento de un área y lo hace accesible y reutilizable.

La palabra Ontología, se ha utilizado a lo largo de la historia para definir elementos con distintos grados de estructura: desde taxonomías simples hasta teorías lógicas complejas. La Web Semántica necesita ontologías con cierto grado de estructura y significado para poder especificar descripciones para los siguientes conceptos:

- Clases y dominios de interés.
- Las relaciones que pueden existir entre las clases descritas.
- Las propiedades o atributos que pueden tener las clases descritas.

Las Ontologías son escritas en lenguajes basados en lógica, de esta manera se tiene la garantía de que son precisas, detalladas, consistentes y significativas[10]. Muchas herramientas para Ontologías pueden realizar procesos de razonamiento sobre el conocimiento que definen, de esta manera, se puede tener cierta "inteligencia" y se pueden desarrollar aplicaciones con capacidades complejas como consulta de información de manera semántica y conceptual, so-

porte a la toma de decisiones, gestión de conocimiento, bases de datos inteligentes, comercio electrónico y entendimiento del lenguaje natural.

En la Web Semántica, resulta necesario el uso de Ontologías ya que proporciona una manera sencilla y eficiente de representar la semántica de los documentos descritos y permitir que sean utilizadas y consultadas por aplicaciones y agentes de software. Para el W3C, el uso de ontologías, permitirá a las aplicaciones del futuro actuar de manera "inteligente" para cumplir con su trabajo de una manera más rápida y con mayor exactitud.

## 2.3. RDF

Sus siglas significan "Resource Description Framework", y es "un lenguaje para la descripción de recursos disponibles en la World Wide Web"[6]. Es un modelo estándar para el intercambio de datos en la Web, RDF extiende la estructura de enlaces de la Web utilizando URIs para dar nombre a las cosas, es así como se establecen enlaces entre dos extremos: el sujeto y el objeto, a través de una relación o propiedad definida a través de RDF. Normalmente, a esta manera de expresar relaciones, se le conoce como "triples". Este modelo de relaciones permite que los datos estructurados o semi-estructurados sean mezclados, expuestos y compartidos a través de diversas aplicaciones en distintas plataformas.

Toda la estructura descrita en un RDF forma una estructura de grafo, donde los arcos representan las relaciones entre dos recursos, los cuales son representados por los nodos de dicho grafo[7]. Este modelo mental, es la manera más utilizada para lograr explicaciones visuales y fáciles de comprender.

La sintaxis de RDF, está basada en XML, al igual que muchos otros metalenguajes y lenguajes de marcado, como HTML y XHTML. De esta manera, es posible además definir atributos adicionales a los recursos que son descritos en el documento.

## 2.4. RDFS

Siendo RDF un lenguaje basado en notación XML, pareciera lógico que siga más o menos el mismo esquema. Si recordamos, y si somos estrictos, cada XML debería tener un XML Schema (o, en su defecto, un DTD, pero estos están siendo poco a poco desplazados por los XMLS), que resulta ser otro archivo XML, que lo define. RFDS significa "RDF Schema" y es "un lenguaje para la descripción de vocabularios en la Web"[8]. Es una extensión semántica de RDF para describir vocabulario, RDFS no busca definirlos, sino describirlos, proporcionar la facilidades para describir tipos y clases de un mismo dominio y servir como sistema de tipos para RDF.

Un RDF Schema, se escribe utilizando las mismas reglas de RDF, es decir, el Esquema RDF es un documento RDF que, a su vez, es capaz de definir a otros RDF. El RDFS, extiende al RDF tradicional para poder implementar jerarquías de clases y relaciones un poco más complejas que las que RDF permite definir por sí solo.

RDFS, al igual que RDF, aún se encuentran en proceso de definición y desarrollo, aunque ya el trabajo está bien adelantado y se puede trabajar y desarrollar con ambos. Un ejemplo de esto es el proyecto FOAF o "Friend of a Friend", que es un proyecto de la Web Semántica para describir personas y las relaciones entre ellas[16] utilizando documentos RDF.

## 2.5. OWL

OWL, "es un Lenguaje de Ontologías Web"[9]. Existen muchos lenguajes y herramientas para desarrollar y trabajar con Ontologías, pero ninguna de las existentes hasta el momento resultaba compatible con una arquitectura Web y mucho menos con la Web Semántica.

El Lenguaje de Ontologías Web, rectifica esto aprovechando una conexión proporcionada por RDF para dar a las Ontologías las siguientes capacidades [9]:

- Capacidad de ser distribuidas y compartidas a través de varios sistemas.

- Escalable a las necesidades de la Web.
- Compatible con los estándares internacionales de accesibilidad Web e Internacionalización (I18N).
- Abierto y extensible.

Adicionalmente, OWL es una extensión de RDF Schema para permitir la expresión de relaciones más complejas entre elementos y clases. Algunos de los recursos que tiene OWL y de los que carece RDFS son las siguientes [9]:

- Recursos para inferir cuáles elementos que tienen varias propiedades son miembros de una clase en particular.
- Recursos para determinar si la totalidad de elementos de una clase tendrán una propiedad determinada o si puede ser que sólo algunos elementos la tengan.
- Recursos para diferenciar relaciones 1:1, 1:N y N:1, permitiendo que las "claves foráneas" de las bases de datos puedan ser representadas en la Ontología.
- Recursos para expresar relaciones entre clases definidas en documentos diferentes a través de la Web.
- Recursos para definir nuevas clases a partir de uniones, intersecciones y complementos de otras ya existentes.
- Recursos para restringir rangos y dominios para especificar combinaciones de clases y propiedades.

Siendo OWL un lenguaje tan extenso, resulta costoso explotarlo a su máxima expresión en aplicaciones web, ya que resulta muy difícil para el motor de inferencia tomar una decisión rápida respecto a la consulta que se le está realizando, es por ello que OWL se divide en tres (3) subconjuntos, según la extensión y el nivel de detalle y profundidad que se desee dar a la

ontología, sin llegar a un modelo de datos en el que la capacidad de decición del razonador no se complique demasiado ni, mucho menos, llegue a ser indecidible.

### OWL Lite

Es el subconjunto más restrictivo de OWL[**?**, **?**, swp] pero, aún así, es el más utilizado para la Web Semántica pues incluye los constructores básicos y necesarios para escribir modelos cuya decibilidad no sea muy compleja.

### OWL DL

Es un subconjunto de OWL un poco más amplio y muy popular en el ámbito de la Lógica Descriptiva (Descriptive Logic)[**?**, **?**, swp] sigue garantizando eficiencia computacional en la inferencia sobre el modelo restringiendo el uso de algunos constructores de OWL.

### OWL Full

Es el conjunto completo del lenguaje OWL, sin ningún tipo de restricción de constructores o utilización de los mismos[**?**, **?**, swp] la ventaja de OWL Full es que es lo más flexible que existe para el modelado de ontologías para Web Semántica, la desventaja es que el nivel de expresividad del lenguaje es tan rico y poderoso que afecta la decibilidad del razonador.

OWL, se encuentra aún en proceso de desarrollo y de definición como estándar, pero el trabajo se encuentra bien adelantado y es posible desarrollar aplicaciones con lo que existe actualmente. Existen numerosas herramientas que permiten modelar y trabajar con RDF, RDFS y OWL, siendo el más utilizado el editor de ontologías Protegé (desarrollado por la universidad de Stanford), permite editar ontologías en OWL de manera gráfica y, además, tiene varios razonadores (motores de inferencia) integrados para validar la integridad de la ontoligía que se está escribiendo.

## 2.6. SPARQL

En la Web Semántica, toda la información acerca de los recursos se encuentra modelada en Ontologías definidas mediante RDF, RDFS y OWL (en cualquiera de sus tres dialectos). SPARQL, "es un lenguaje de consultas para RDF" [11], es decir, así como en el modelo relacional, pueden consultarse las bases de datos a través de SQL y, de esta manera, obtener subconjuntos de la información almacenada según un conjunto de restricciones, mediante consultas SPARQL puede obtenerse subconjuntos de la información contenida en uno o varios documentos RDF en forma de grafos, dependiendo de las relaciones y de lo que el usuario desee buscar.

SPARQL, viene a ser la evolución de RQL y aún se encuentra en proceso de desarrollo, aunque existen numerosos motores de consulta en SPARQL y se puede trabajar sobre lo que ya existe.

## 2.7. Motores de Inferencia

Una vez que se tiene la Ontología ya definida, es necesario definir ciertas reglas de inferencia a ser aplicadas sobre ese conocimiento ya descrito. Estas reglas de inferencia son aplicadas por un Motor de Inferencia, que es un programa que explora la base de conocimiento, aplicando ciertas reglas, hasta dar con la solución que mejor se adapte a las necesidades del usuario [12].

En la Web Semántica, es el motor de inferencia quien traduce la consulta del usuario y evalúa, mediante reglas bien definidas, qué es lo que el usuario realmente está buscando, es por ello que el motor de inferencia juega un papel de suma importancia en este marco, vale acotar, que los motores de inferencia para Web Semántica, deben ser capaces de "razonar" sobre RDF, RDFS y OWL.

CAPÍTULO 3

# MARCO METODOLÓGICO

Las metodologías de desarrollo tradicionales, parecieran haber sido diseñadas para proyectos extensos, con equipos numerosos, en los que se goza de roles bien definidos y que, cada rol, cumple con una labor específica en cada etapa del ciclo de vida.

La realidad de este proyecto es otra, sólo se cuenta con una persona para realizar el trabajo de todos los roles, además, el tiempo de entrega es limitado y resulta, además, conveniente entregarlo lo antes posible. Es por ello que se propone trabajar con una metodología Iterativa Incremental bajo el esquema ágil. Los preceptos del esquema de trabajo enmarcado en el Desarrollo Ágil, fueron definidos por Kent Beck año 2001[13] en un documento denominado El Manifiesto Ágil (The Agile Manifesto), este manifiesto se cita a continuación:

"Estamos descubriendo nuevas maneras de desarrollar software tanto por nuestra propia experiencia como ayudando a terceros. A través de esta experiencia hemos aprendido a valorar:

- **Individuos e interacciones** sobre procesos y herramientas.
- **Software que funciona** sobre documentación exhaustiva.

- **Colaboración con el cliente** sobre negociación de contratos.
- **Responder ante el cambio** sobre el seguimiento de un plan.

Esto es, aunque los elementos a la derecha tienen valor, nosotros valoramos por encima de ellos a los que están a la izquierda".

El Trabajo Especial de Grado, tiene un carácter investigativo y aplicativo, es decir, debe investigarse acerca del tema y realizar un pequeño aporte al área de investigación mediante el desarrollo de algún producto final. Por ello, dado el carácter de un TEG, todos y cada uno de los principios definidos en el Manifiesto Ágil de Kent Beck tienen sentido en un proyecto de este tipo pues, debe valorarse “Individuos e interacciones sobre procesos y herramientas", esto es, debe valorarse más la interacción entre el tesista y el tutor que los procesos y herramientas necesarios para ello, por ejemplo. Debe darse más valor al “Software de funciona sobre la documentación exhaustiva", en este caso, para la presentación final, debe mostrarse algo funcional, si bien la documentación es importante, no se hará de manera exhaustiva, únicamente lo necesario para poner orden en el proyecto y para que, posteriormente, si el presente trabajo resulta de interés para alguien, pueda entender qué fue lo que se hizo. “Colaboración con el cliente sobre negociación de contratos", si bien esto no tiene mucha pertinencia al hablar de un TEG, puede interpretarse como que la Escuela de Ingeniería Informática y el Tesista deberían colaborar pues ambos persiguen un objetivo común: innovar. Finalmente, Responder ante el cambio sobre el seguimiento de un plan"pues al tratarse de un proyecto de investigación, no todos los requerimientos están claramente definidos ni pueden predecirse en su totalidad, es por ello que conforme se va investigando y despejando la incertidumbre en algunos temas, pueden ir surgiendo nuevas cosas que no se esperaban y se debe estar preparado para responder y adaptar el plan a las nuevas condiciones.

Existen numerosas metodologías y ciclos de vida basados en el modelo ágil, para este proyecto en particular se propone utilizar una metodología basada en Scrum para organizar el proyecto y XP (Programación Extrema o eXtreme Programming en inglés) para la organización de cada una de las iteraciones.

## 3.1. Descripción de la Metodología Planteada

Para describir la metodología que se utilizará para el desarrollo del proyecto, es necesario primero estudiar las dos metodologías antes mencionadas:

### SCRUM

SCRUM es una metodología creada por Jeff Sutherland y su equipo de desarrollo a principios de la década de 1990 [14]. Los principios de SCRUM están enmarcados dentro del Manifiesto Ágil y es un proceso que lleva el Desarrollo de Software a través de las siguientes actividades: requerimientos, análisis, diseño, evolución y entrega. Cada una de esas actividades son realizadas dentro de un patrón de trabajo llamado Sprint, todo el trabajo realizado dentro de un Sprint es adaptado al problema y frecuentemente modificado por el equipo de desarrollo a medida que las condiciones van cambiando.

SCRUM hace énfasis en la utilización de procesos de software que son efectivos en proyectos con tiempos cortos de entrega y requerimientos cambiantes. Esos procesos, en general, definen dos grandes actividades de desarrollo:

- **Backlog:** el backlog, constituye una lista priorizada de requerimientos que agregan valor de negocio al producto, estos requerimientos pueden ser agregados en cualquier momento y, de esta manera, se introducen los cambios en el proyecto [14]. El backlog puede ser, además, modificado en cualquier momento para adaptar las prioridades a los cambios del negocio.

- **Sprints:** los Sprints constituyen paquetes de trabajo que son necesarios para desarrollar un requerimiento o un conjunto de ellos [14]. Los cambios no pueden ser introducidos dentro de un Sprint, de esta manera se asegura que el trabajo que se realiza es estable, pues los requerimientos que se seleccionan para ser desarrollados en un Sprint ya deben estar definidos y se debe tener la certeza de que, en caso de cambiar, será manejable.

- **Scrum Meetings:** son reuniones cortas que, usualmente, se realizan todos los días durante un sprint [14]. Durante los Scrum Meetings se responden a tres preguntas básicas:
  - ¿Qué has hecho desde la última reunión?.
  - ¿Cuáles obstáculos has encontrado?.
  - ¿Cuál es tu planificación hasta la próxima reunión?.
- **Demos:** se van entregando los resultados de las funcionalidades desarrolladas al cliente para que puedan ser evaluados [14]. De esta manera, se va entregando Software Listo y Funcional que agrega valor al negocio del cliente en cada iteración.

De esta manera, SCRUM permite que los equipos trabajen de manera eficiente y estable en proyectos en los que la incertidumbre siempre está presente.

## eXtreme Programming

La metodología eXtreme Programming (XP) es la más utilizada y la más conocida de las metodologías ágiles [15]. Fue inicialmente concebida por Kent Beck a finales de la década de 1980 y se enfoca en hacer énfasis en la etapa de implementación del ciclo de vida, tal como su nombre lo indica, en la parte de programación o codificación, tomando como buena práctica la programación en parejas, de esta manera, mientras uno de los desarrolladores codifica, el otro está observando y advirtiendo de errores a quien escribe el código.

Kent Beck definió cinco valores para establecer las bases de XP como metodología. Cada uno de esos valores se utilizan para poner en marcha las actividades, acciones y tareas de XP [14]:

- **Comunicación**: entre los clientes y los desarrolladores. XP valora más la comunicación informal por ser más rápida en comparación con volúmenes enormes de documentación como medio de comunicación principal.

- **Simplicidad**: XP restringe a los desarrolladores a realizar el diseño y la implementación de código que satisfaga los requerimientos actuales en vez de pensar en factores futuros, si el diseño puede ser mejorado, puede ser modificado posteriormente.
- **Retroalimentación**: durante un proyecto en necesario muchas veces regresar a una etapa previa y modificar algo para que los requerimientos actuales funcionen sin alterar los que ya fueron implementados. El cambio es algo constante en todo proyecto, pero para poder hacerlo correctamente, se necesita retroalimentación.
- **Coraje**: implementar código sin miedo a las consecuencias, pero siempre de manera coherente.

En eXtreme Programming, no se libera una pieza de software hasta que está totalmente funcional y probada, para asegurar que el software funciona, se desarrollan Pruebas Unitarias.

La metodología propuesta para el desarrollo del presente proyecto, toma la organización de SCRUM y los valores y la filosofía basada en la codificación de XP para llevar a cabo las tareas y actividades concernientes al desarrollo del producto final de este trabajo, obviando la exigencia de XP acerca de la programación en parejas ya que el presente trabajo es realizado por sólo una persona.

## Plan General de Trabajo

Se plantea, organizar el desarrollo en las siguientes iteraciones, con una duración de tres a cuatro semanas cada una:

1. Investigación y análisis.
2. Diseño de la arquitectura del sistema.
3. Establecimiento del ambiente de desarrollo.
4. Anlálisis y desarrollo de la ontología.

5. Integración del modelo de conocimiento con el API seleccionado.

6. Desarrollo del traductor de lenguaje natural a SPARQL.

7. Desarrollo del motor de búsqueda.

8. Desarrollo e integración con las vistas.

CAPÍTULO 4

# DESARROLLO

El desarrollo de este trabajo, se describe siguiendo la metodología planteada anteriormente. Las iteraciones citadas en el *Marco Metodológico* serán descritas a continuación de manera consecutiva:

## 4.1. [Iteración 1]: Investigación y análisis

Durante esta etapa no se desarrolló ningún tipo de software, es por ello que no se verán historias de usuario, diseño, desarrollo ni pruebas unitarias.

### Desarrollo de las tareas

Las tareas de esta iteración se basaron principalmente en la investigación y profundización de los temas relacionados a este trabajo, así como la búsqueda de herramientas para el ensamblaje de la plataforma de desarrollo.

**Investigación de los temas relacionados**

La mayor parte de la investigación realizada fue descrita en el *Marco Teórico* del presente trabajo. El tópico central del presente trabajo resulta ser *La Web Semántica*, por ello, es necesario tomar en cuenta todos los conceptos que se desprenden de dicho tópico. Los conceptos más importantes que se desprenden de *La Web Semántica* son citados a continuación:

- Ontologías y modelos de conocimiento.
- Motores de inferencia.
- Servicios Web.

Las *Ontologías* y *Modelos de conocimiento*, son el marco central de la *Web Semántica*. Es lo que establece toda la estructura de metadatos en la que se basa el etiquetado e indexado de los recursos que forman parte de la base de conocimientos. Además, establece las relaciones entre las metaentidades que conforman la *Ontología*. Define el vocabulario que modela el dominio del problema a ser resuelto.

Los *Motores de inferencia* son herramientas que, dadas ciertas reglas sobre un *modelo de conocimiento*, son capaces de deducir nueva información y nuevas relaciones entre las metaentidades y, por lo tanto, nuevas relaciones entre los recursos dentro de la base de conocimientos.

Los *Servicios Web*, si bien son un concepto que ya tiene tiempo, cobran especial importancia en la *Web Semántica* pues, cuando se haga efectiva la evolución de la Web 2.0 a la Web 3.0, debe garantizarse total interoperabilidad e independencia de plataformas para que los agentes puedan ser capaces de consultar e intercambiar información y *modelos de conocimiento* de distintas fuentes. Los *servicios* web exponen toda la funcionalidad de un sistema a través de métodos invocables utilizando HTTP sobre TCP. Es por ello que, a través de *Servicios Web*, es posible desarrollar servicios sobre cualquier plataforma y, sin importar en cual otra esté desarrollado, cualquier cliente será capaz de consumir los servicios pues la invocación se encapsula dentro de un protocolo común entre ambas plataformas.

### El problema y los requerimientos

Del problema general del proyecto, puede plantearse de la siguiente manera: *¿Cómo facilitar el acceso a la información acerca de Ciencias de la Computación a personas interesadas en el área.* De este problema, puede identificarse los siguientes requerimientos:

- Una interfaz web donde los usuarios puedan interactuar con el sistema.
- Una interfaz de edición que permita a usuarios autorizados agregar nuevos recursos y extender la base de conocimiento.
- Una interfaz de edición que permita a usuarios autorizados agregar metainformación nueva y extender el modelo de conocimiento.
- Un componente de traducción que convierta consultas en lenguaje natural a SPARQL, el lenguaje de consultas sobre RDF/RDFS/OWL.
- Desarrollo de un modelo de conocimiento del área de Ciencias de la Computación e Ingeniería Informática.
- Un componente capaz de realizar consultas e inferencias sobre el modelo de conocimiento realizado.

### Selección de la plataforma

Al ser una aplicación bajo la arquitectura Cliente-Servidor, hay que tener especial cuidado en seleccionar las mejores herramientas para desarrollar cada uno de los componente de dicha arquitectura.

Al principio, se planteó llevar a cabo todo el desarrollo utilizando *Python* como lenguaje de programación y las librerías nativas disponibles a través de *easy_install*, pues *Python* es uno de los pocos lenguajes que ofrece librerías nativas para la manipulación de documentos RDF,

además de las ventajas propias que ofrece el lenguaje desde el punto de vista sintáctico y de facilidad de aprendizaje y de programación. Pero a la hora de buscar Frameworks que permitieran el desarrollo de aplicaciones basadas en Web Semántica y Motores de Inferencia, la información y la cantidad de herramientas resultó limitada.

Debido a la situación expuesta en el párrafo anterior, se planteó evaluar el desarrollo por separado: seleccionar la mejor plataforma para desarrollar el servidor y, por otra parte, la mejor plataforma para desarrollar el cliente:

**Servidor**

Las dos opciones más fuertes para el desarrollo del lado del Servidor fueron *Python* por un lado, por ser una plataforma 100 % libre y de código abierto y por la gran cantidad de librerías disponibles para extender el lenguaje, y por otro lado *Java* representaba una opción viable, por ser una plataforma con más de 15 años de desarrollo y estándar *de-facto* de la industria, a continuación se presentan las características más destacables de cada una de las opciones:

- **Python[17]:**
  - Totalmente abierto y libre.
  - Sintaxis clara y legible.
  - Capacidad poderosa de introspección.
  - Multiparadigma: funcional, estructurado y orientado a objetos.
  - Tipos de dato dinámicos.
  - Gestión de errores basada en excepciones.
  - Puede extenderse a través de la escritura de módulos en lenguaje C o C++.
  - Gran cantidad de librerías, entre ellas varias para procesar documentos RDF.

- Facilidad para interoperar con otras plataformas.
- Precompilado y semi-interpretado.

- **Java[18]:**
  - Sencillo, fue diseñado para facilitar las tareas del programador profesional.
  - Orientado a objetos.
  - Distribuido, facilita el desarrollo de aplicaciones que hacen uso de la red mediante la incorporación de clases que manejan protocolos TCP/IP.
  - Precompilado y semi-interpretado.
  - Arquitectura neutra, debido a que se ejecuta en una máquina virtual.
  - Portable, dado que al “compilar"no se produce un archivo ejecutable como ocurre en los lenguajes compilados (como C, C++ y Pascal, por ejemplo), sino un *bytecode* que es ejecutado por la máquina virtual, este *bytecode* puede ser ejecutado en cualquier otro sistema operativo, siempre y cuando exista en él una instancia de la *Máquina Virtual de Java*.
  - Multihilo, un programa en *Java* puede ejecutar múltiples tareas de manera simultánea

Ambas plataformas ofrecen prestaciones muy similares, si bien el núcleo de *Python* no es muy amplio, posée gran cantidad de librerías para extender su funcionalidad. De igual manera, *Java* incorpora cientos de clases que lo convierten en un lenguaje amplio y poderoso.

Para el servidor, se decidió trabajar con *Java*, ya que, como se verá más adelante, ofrece el mejor framework para el desarrollo de aplicaciones basadas en Web Semántica y, al seleccionar dicho framework, la transitividad nos lleva a trabajar con este lenguaje.

### Cliente

El cliente, es el encargado de realizar peticiones al servidor e interactuar con el usuario final, es por ello que la generación de vistas es un aspecto importante en esta parte del sistema.

Para el desarrollo del cliente, fue seleccionada la plataforma *Python*, debido a todas las razones expuestas anteriormente.

### Selección de los frameworks de desarrollo

Luego de realizar una investigación en la web, los frameworks de desarrollo que parecen ser más utilizados para las aplicaciones semánticas son:

- **Sesame:** escrito en *Java*.
- **RedLand:** escrito en C y accesible desde *Python* mediante *Wrapping*.
- **Jena:** escrito en *Java*.
- **CubicWeb:** escrito en *Python*.

En este caso, se tomó la decisión de trabajar con el framework *Jena* debido a que es el que ofrece la mayor cantidad de documentación disponible en línea, además, es compatible con todos los niveles de representación semántica de metainformación (RDF, RDFS y OWL), posée varios motores de inferencia ya integrados y permite la integración con motores de inferencia externos, tiene un motor de consultas SPARQL y ofrece la posibilidad de integrarse con medios de persistencia externos, además de ser libre y de código abierto[19].

El framework *Sesame*, ofrece prestaciones técnicas similares a las de *Jena*, sin embargo, la documentación disponible en línea no es comparable con la que ofrece este último y, a pesar de ser de código abierto, es propiedad de una empresa alemana llamada *ADUNA*[20] y para poder acceder a información más profunda o solicitar ayuda en algo relacionado al framework,

es necesario pagar por horas de consultoría, lo que hace poco viable la utilización de *Sesame* para este proyecto.

*RedLand*, escrito en C y accesible desde *Python*, luego de leer la documentación, es compatible sólo con RDF y, para este trabajo, el modelo de conocimientos planteado utilizaría anotaciones definidas en la especificación RDFS y, posiblemente, algunas definidas en el dialecto OWL-Lite, popr lo que fue descartado. Además, el proceso de instalación y configuración resulta complicado comparado al de las otras opciones[21].

Finalmente *CubicWeb*, a pesar de ser un framework realmente completo pues ofrece toda la plataforma para el desarrollo: desde la persistencia, hasta la generación de vistas, pero fue descartado por no ser compatible con SPARQL, sino con su antecesor: RQL[22].

**Selección del medio de persistencia**

Para la persistencia de datos, si bien los manejadores de base de datos relacionales tradicionales como MySQL y Postgres ofrecen paquetes para la gestión de documentos RDF y OWL, existe una alternativa que ofrece dicha funcionalidad de manera nativa. Se trata de *Virtuoso*, un manejador de base de datos con capacidad de gestionar información en formato RDF y XML, compatible con los estándares ODBC y JDBC, posée un motor de inferencia interno, puede correr en ambientes federados[23] y, además, exite una edición OpenSource bastante completa y muy bien documentada.

CAPÍTULO 5

# RESULTADOS

Exponer los resultados del desarrollo

CAPÍTULO 6

# CONCLUSIONES Y RECOMENDACIONES

## 6.1. Conclusiones

...

## 6.2. Recomendaciones

...

# BIBLIOGRAFÍA


[1] Antoniou, Grigori y Van Harmelen, Frank. (2003). A Semantic Web Primer, Massachusetts: MIT Press.

[2] Cobo, Cristóbal y Pardo, Hugo. (2007) Planeta 2.0: Inteligencia Colectiva o Medios Fast Food. México DF: Grup de Recerca d'Interaccions Digitals.

[3] Éxito Explorador (Agosto 31, 2010). Estadísticas Mundiales de Internet [Datos en línea] en http://www.exitoexportador.com/stats.htm [Consulta: 2010, Noviembre 28].

[4] W3C. Guía Breve de la Web Semántica [Documento en línea]. Disponible: http://www.w3c.es/divulgacion/guiasbreves/websemantica [Consulta: 2010, Noviembre 28].

[5] Casanova, Ricardo. (2010). El Modelo Web 2.0. Presentación sobre el modelo de negocios orientado a la Web 2.0.

[6] W3C. RDF Primmer [Documento en línea]. Disponible: http://www.w3.org/TR/2004/REC-rdf-primer-20040210/ [Consulta: 2010, Noviembre 28].

[7] W3C. Resource Description Framework (RDF) [Documento en línea]. Disponible: http://www.w3.org/RDF/ [Consulta: 2010, Noviembre 29].

[8] GSI. RDF Schema (RDFS) [Presentación en línea]. Disponible: www.gsi.dit.upm.es/ gfer/ssii/RDFS.pdf [Consulta: 2010, Noviembre 30].

[9] W3C. Preguntas Frecuentes del OWL [Documento en línea]. Disponible: http://www.w3c.es/Traducciones/es/SW/2005/owlfaq [Consulta: 2010, Noviembre 30].

[10] W3C. OWL: Use Cases and Requirements [Documento en línea]. Disponible: http://www.w3.org/TR/2004/REC-webont-req-20040210/#onto-def [Consulta: 2010, Diciembre 01].

[11] W3C. SPARQL Query Language for RDF [Documento en línea]. Disponible: http://www.w3.org/TR/rdf-sparql-query/ [Consulta: 2010, Diciembre 02].

[12] Diaz, Selim. Sistemas Expertos. Un paso en la simulación del razonamiento humano [Documento en línea]. Disponible: http://www.monografias.com/trabajos23/sistemas-expertos/sistemas-expertos.shtml [Consulta: 2010, Noviembre 30]

[13] Beck, Kent et al. Manifesto for Agile Software Development [Documento en línea]. Disponible: http://agilemanifesto.org/ [Consulta: 2010, Diciembre 02].

[14] Pressman, Roger. (2010). Software Engineering: a Practitioner's Approach. New York: McGraw Hill.

[15] Sommervile, Ian. (2007). Software Engineering. New York: Pearson Education.

[16] The Friend of a Friend Project. FOAF Project [Documento en línea]. Disponible: http://www.foaf-project.org [Consulta: 2011, Enero 15].

[17] Python Community. About Python [Documento en línea]. Disponible en: http://www.python.org/about [Consulta: 2011, Agosto 08].

[18] Joyanes, Luis y Zahonero Ignacio. Programación en Java 2 (2002). Madrid: McGraw Hill.

[19] Jena Community. About Jena [Documento en línea]. Disponible: http://jena.sourceforge.net/index.html [Consulta: 2011, Agosto 08].

[20] Autor desconocido. About Sesame [Documento en línea]. Disponible: http://www.openrdf.org/about.jsp [Consulta: 2011, Agosto 08].

[21] RedLand Community. RedLand FAQ [Documento en línea]. Disponible: http://librdf.org/FAQS.html [Consulta: 2011, Agosto 08].

[22] CubicWeb Community. The Semantic Web is a Construction Game [Documento en línea]. Disponible http://www.cubicweb.org/ [Consulta: 2011, Agosto 08].

[23] Virtuoso Community. Virtuoso FAQ [Documento en línea]. Disponoble: http://virtuoso.openlinksw.com/dataspace/dav/wiki/Main/VOSVirtuoso6FAQ [Consulta: 2011, Agosto 08].