Introduction to the Semantic Web

Rafael Corchuelo

Universidad de Sevilla, Dep. de Lenguajes y Sistemas Informáticos.
ETSI Informática, Avda. Reina Mercedes, s/n, Sevilla 41012.
Email: corchu@us.es, Teléfono: 954552770, Fax: 954557139

Abstract La Web Semántica se presenta con frecuencia como una gran revolución que permitirá a los ordenadores *entender* los contenidos de la Web clásica. El objetivo de este documento es servir de introducción a los conceptos y tecnologías fundamentales sobre los que se sustenta. Está pensado para personas con conocimientos informáticos que aún no están familiarizadas con este tema, por lo que no se trata de un documento formal y detallado, sino de un texto eminentemente introductorio y didáctico. Se proporcionan también varias referencias a la bibliografía y herramientas que ayudarán al lector a profundizar y entender mejor todos los detalles de la Web Semántica.

1 Introducción

La Web Semántica fue presentada por Berners-Lee, Hendler y Lassila en el año 2001 en un artículo de la revista Scientific American que sin duda alguna está ya incluido en los anales de la historia de la informática. En este artículo hicieron un análisis de la Web conocida hasta entonces y no es difícil intuir que la conclusión principal es que se trataba de una fenomenal fuente de información para las personas, puesto que la mayor parte de las tecnologías existentes estaban orientadas a construir sitios web más fáciles de usar, más intuitivos, capaces de aprovechar mejor las posibilidades gráficas y multimedia de los ordenadores de la época, etcétera. En definitiva, era una Web por y para las personas.

Por desgracia, esta ventaja rápidamente se torna en desventaja cuando es necesario alimentar una aplicación o un proceso de negocio con información que se encuentra en la Web, puesto que los formatos adecuados para las personas en rara ocasión facilitan a los programadores el acceso a la información. Berners-Lee, Hendler y Lassila idearon la Web Semántica como la solución a este problema y proporcionaron la siguiente definición:

> La Web Semántica es una extensión de la Web actual en la que la información tiene un significado bien definido que permite a personas y ordenadores trabajar mejor de forma conjunta.

La idea clave de esta definición es que la Web Semántica no es una nueva Web, sino tan sólo un complemento a la actual que facilitará a los programadores

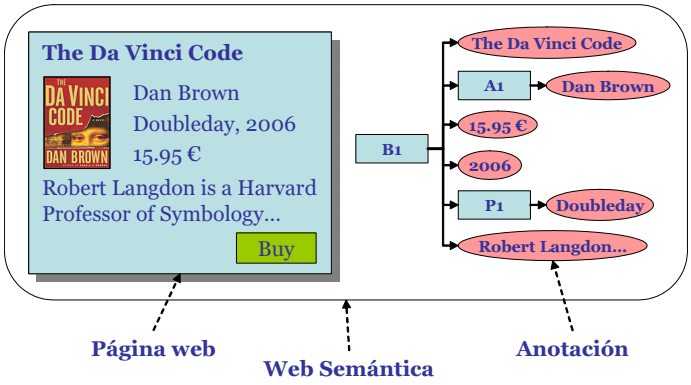

Figura 1. La Web Semántica aporta anotaciones que describen los recursos existentes en la Web.

la tarea de desarrollar nuevas aplicaciones capaces de hacer uso de la información que contiene la Web clásica con un esfuerzo prácticamente nulo. Tenga en cuenta que no estamos diciendo que actualmente no sea posible utilizar la información que contiene la Web clásica en procesos informatizados, tan sólo que el esfuerzo necesario para conseguirlo no siempre es razonable. Para entender esto mejor, tome una página cualquiera de su sitio de comercio electrónico favorito y reflexione un momento sobre cómo podría hacer un programa capaz de leer la información sobre las ofertas que proporciona ese sitio; seguramente su primera idea será identificar algunas marcas que delimiten de forma no ambigua la información que desea extraer de las páginas, pero seguro que no es una tarea fácil. Para intentar paliar este problema, muchos investigadores han trabajado en sistemas llamados wrappers que permiten de una forma semi-automática extraer información estructurada a partir de páginas web que fueron originalmente pensadas tan sólo para personas.

Los sistemas de wrapping son una solución de ingeniería efectiva que en muchos casos ayuda a reducir los costes de desarrollo. El objetivo de la Web semántica es reducirlos a cero y la forma en que esta promesa se está materializando consiste en el uso de anotaciones para describir los recursos de la Web clásica mediante datos estructurados, como se ilustra en la figura 1. Evidentemente, para que esta idea sea efectiva y realmente facilite el desarrollo de aplicaciones que usan la información disponible en la Web es necesario que las anotaciones estén escritas en un lenguaje estructurado y fácil de tratar mediante programación; este lenguaje es RDF.

Aunque esta materialización de la Web Semántica es tremendamente simple, permitirá, entre otras cosas, mejorar los motores de búsqueda y crear agentes personales mucho más avanzados que los que existen en la actualidad. Los motores de búsqueda actuales están basados en el uso de palabras clave, no conceptos. Por ejemplo, si le damos a Google la palabra `Triana` nos devolverá una lista de páginas que contienen esta palabra o alguna variación de la misma; el problema

es que esto incluye páginas sobre el barrio de Triana en Sevilla, sobre el grupo de música Triana Pura, sobre la escritora Andreya Triana, o incluso sobre el sistema distribuido de análisis de señales de igual nombre. Si la Web Semántica fuera una realidad a día de hoy, los motores de búsqueda podrían agrupar fácilmente los resultados en función a la clase de los recursos, proporcionando así una experiencia de búsqueda mucho más simple para los usuarios. Un agente personal es un pequeño programa que realiza alguna función útil para las personas, por ejemplo: determinar cuál es la tienda en que se ofrece un determinado producto más barato, monitorizar la bolsa y alertar de cuándo es un buen momento para comprar determinadas acciones, o simplemente ofrecer interfaces de búsqueda más avanzadas que las que proporcionan los sitios web actuales. La clave, de nuevo, son las anotaciones, que permiten a los programadores diseñar aplicaciones que acceden a la información en la Web con muy poco esfuerzo adicional.

Nuestro objetivo con este documento es ayudarle a introducirse en el mundo de la Web Semántica, proporcionando una idea tan clara como sea posible del concepto y las tecnologías que están ayudando a materializarlo en la práctica. Antes de seguir adelante, no obstante, es importante que tenga en cuenta que la visión que el autor presenta es desde la perspectiva de la Ingeniería del Software, es decir, desde la perspectiva de mejorar los procesos de producción de software reduciendo costes, incrementando la automatización y mejorando la calidad de los productos resultantes. Si este documento hubiera sido escrito por un especialista en Inteligencia Artificial, la perspectiva seguramente sería distinta.

El resto del documento está estructurado de la siguiente forma: en la sección 2 profundizamos en qué significa entender, pues esto nos permitirá estructurar adecuadamente el resto de la presentación entorno a los conceptos de modelo, inferencia y preguntas; en la sección 3 presentaremos el concepto de ontología, que es la base de los modelos en la Web Semántica, mientras que en la sección 4 trataremos el tema de la inferencia y en la sección 5 el de las preguntas; en la sección 6 le proporcionaremos algunos detalles sobre microformatos y GRDDL, que son las técnicas que apoya la W3C para facilitar la creación de anotaciones; la sección 7 presenta algunas conclusiones finales; el apéndice A proporciona algunas referencias bibliográficas y detalles sobre algunas herramientas que le serán útiles en su estudio de la Web Semántica.

2 Qué Significa Entender

Generalmente se suele decir que gracias a la Web Semántica las máquinas podrán entender el contenido de la Web clásica. Por desgracia, esta frase se malinterpreta muy frecuentemente, lo que conduce a conclusiones generalmente equivocadas. Por este motivo, lo primero que haremos es profundizar en qué significa entender.

Comenzaremos por un sencillo test: échele un vistazo a las dos páginas que aparecen en la figura 2. Seguro que es capaz de entender su contenido y por lo tanto sabe que hablan de un libro llamado “The Da Vinci Code” y de un escritor llamado Dan Brown; además, seguramente pensará en otras personas que le han hablado sobre este libro, o incluso en la película que se hizo recientemente y en

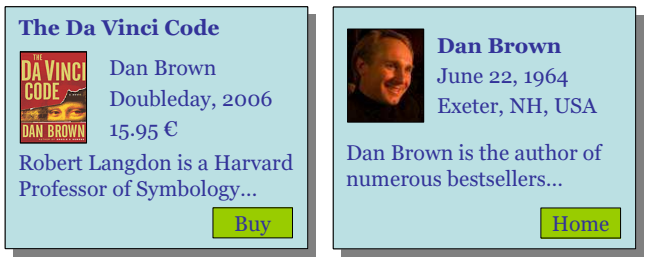

Figura 2. Dos páginas web muy sencillas. Las personas pueden entender fácilmente su contenido. Las máquinas, por el contrario, no.

sus actores. La idea intuitiva acerca del término entender parece estar clara, pero si tuviéramos que formalizarla seguro que tendríamos algo más de problema. Por suerte, en el mundo de la psicología, el concepto está bien definido y se dice que entender es equivalente a ser capaz de modelar la realidad, inferir conocimiento a partir de dichos modelos y responder a preguntas sobre los mismos.

La capacidad de modelar la realidad está relacionada con la habilidad que Ud. tiene para identificar en un problema determinado cuáles son las clases de objetos o recursos[1] que aparecen en el mismo y cuáles son sus propiedades. En el caso de las páginas web de la figura 2 estamos seguro de que ha sido capaz de obtener un modelo sencillo en el que ha identificado al menos un par de clases de recursos que representan los conceptos de libro y de escritor, por lo que podríamos denominarlas `Book` y `Writer`, respectivamente; además, seguro que también ha sido capaz de identificar propiedades como `title` o `price` para la primera clase y propiedades como `name` o `biography` para la segunda; finalmente, también debe haber sido capaz de identificar al menos una relación entre estos dos conceptos, dado que todo libro involucra al menos a un escritor.

En la Web, la capacidad de inferencia está íntimamente relacionada con la capacidad para clasificar recursos, es decir: dado un determinado recurso, somos capaces de determinar cuál de las múltiples clases que conocemos es la que describe mejor su significado; en nuestro ejemplo de la figura 2, esta capacidad es evidente en el momento en que Ud. ha identificado que estas páginas pueden ser bien descritas por las clases `Book` y `Writer`. La inferencia también está relacionada con la capacidad de correlacionar, es decir, de establecer relaciones entre lo que ya conocemos y lo que vamos descubriendo. La correlación nos permite, por ejemplo, determinar que el libro sobre el que habla la primera página de la figura 2 es el que leímos el pasado verano, pero también establecer correspondencias entre los diversos modelos que conocemos o que otras personas han podido idear para un mismo tipo de recurso. Por ejemplo, es posible que Ud. haya ignorado la imagen que aparece en la página, pero otra persona podría considerar que

[1] En el mundo de la Web Semántica, los objetos con los que se trabaja son siempre documentos existentes en la Web que se pueden identificar mediante una URI, por lo que es habitual hacer referencia a ellos como recursos.

es realmente importante e incluirla en su modelo; gracias a nuestra capacidad innata de correlacionar tanto recursos como modelos esto no supondrá ningún problema. Además de la capacidad de clasificación y de correlación, la inferencia también está relacionada con la capacidad de extrapolar los conocimientos adquiridos; por ejemplo, si a partir de otras fuentes de información hemos descubierto que se han vendido unos 25 millones de copias del libro que nos sirve de ejemplo y que el precio debe rondar los 15.95 €, fácilmente podemos extrapolar nuestro conocimiento y concluir que se trata de un libro que ha debido reportar unos grandes beneficios a su escritor y a las editoriales que lo han comercializado.

Fíjese en que las dos capacidades anteriores están relacionadas con procesos que se desarrollan en nuestra mente y de los que prácticamente no somos conscientes. La única forma de demostrar que estos procesos se han producido y que realmente hemos sido capaces de entender la información que se muestra en las páginas de la figura 2 es respondiendo a preguntas que otras personas nos puedan plantear. Las preguntas pueden ser de evidencia, de inferencia o cuestiones generales. Las primeras son las más sencillas y son las que generalmente responden los sistemas de gestión de bases de datos típicos pues tan sólo se trata de proporcionar información sobre aquello que conocemos, por ejemplo, el título o el precio del libro en la figura 2; por el contrario, las preguntas de inferencia requieren clasificar y correlacionar recursos o modelos, así como extrapolar información; las cuestiones generales son un nivel superior en el que responder a una pregunta puede llevar consigo la búsqueda de nuevas fuentes de información, es decir, lo que de forma general podríamos resumir diciendo que son preguntas que requieren investigación para poder responderlas.

Estamos seguro de que algunos de los comentarios en los párrafos anteriores han debido de parecerle de perogrullo, demasiado evidentes. Esto es debido a que Ud. es una persona y las personas nos caracterizamos porque somos capaces de entender, es decir, tenemos muy buenas habilidades para modelar, inferir y responder a preguntas. Por desgracia, los ordenadores no tienen buenas capacidades de modelado puesto que aún no hemos sido capaces de diseñar algoritmos apropiados. Piense en que tuviera que hacer un programa al que se le den un conjunto de páginas web y sea capaz de modelar la información que aparece en ellas de una forma automática; esto es aún un gran reto de investigación por lo que seguramente no podría comprometerse a terminar ese programa en un tiempo razonable. Por el contrario, sí que existen muchas propuestas que permiten a las máquinas inferir conocimiento y también muchas que les permiten responder a preguntas, sobre todo las de evidencia.

La conclusión más inmediata de lo que acabamos de contar es que si nos falta capacidad de modelado automático, difícilmente podremos asumir que los ordenadores podrán entender el significado de la Web actual, por lo que en la práctica, no podrán ni inferir ni responder a preguntas por sí solos. La clave son los modelos y si las máquinas no son capaces de obtenerlos de forma automática, y parece que esto no va a cambiar en muchos años, la única solución es que seamos las personas las que se los proporcionemos.

3 Modelos ontológicos

A los modelos que se usan en la Web Semántica se les suele hacer referencia como ontologías. Este término surgió en el mundo de la filosofía, en donde hace referencia al estudio del ser y de la existencia. En el contexto de la Web Semántica, se ha hecho tremendamente popular, pero al no ser una palabra de uso común, entender qué significa no siempre es sencillo. Uno de los primeros autores que utilizó este término en el contexto de la informática fue Tom Gruber, que en un artículo de 1993 lo definía de la siguiente forma:

> Una ontología es una especificación de una conceptualización.

En otras palabras más sencillas: para Gruber una ontología era simplemente un modelo conceptual y éste es el significado que en la mayor parte de las ocasiones se le da al término en el contexto de la Web Semántica. Tan sólo queda un pequeño detalle para que estos modelos realmente tengan éxito en la Web: deben de ser compartidos por una amplia comunidad, pues de poco sirve en la Web un modelo que tan sólo sea útil para una o dos personas. Este detalle fue introducido en un artículo un poco posterior.

Dado que una ontología no es más que un modelo conceptual, en principio, cualquier lenguaje de modelado podría ser apropiado para definirlas, por ejemplo, UML y otros lenguajes del mundo de las bases de datos. No obstante, estos lenguajes no resultan del todo apropiados para la Web Semántica. Uno de los principales problemas es que trabajan con objetos, es decir con entidades cuya estructura y semántica queda perfectamente definida sobre la base de atributos y métodos. Por el contrario, los recursos en la Web clásica en rara ocasión describen la estructura de los datos que presentan. Piense por ejemplo en un documento HTML o en una animación Flash; se trata de formatos orientados a la presentación, por lo que ponen todo el énfasis en la comunicación visual, no en la descripción de la estructura y la semántica de los datos presentados. Por lo tanto, una de las diferencias principales entre lenguajes como UML y los lenguajes específicos para ontologías es la propia naturaleza de las entidades que modelan; mientras que en el primer caso el modelo es la base y no es posible que existan objetos a menos que se haya creado previamente un modelo para los mismos, en el segundo el objetivo es modelar recursos que ya existen en la Web y que van a evolucionar con completa independencia a los posibles modelos que se puedan aplicar a los mismos.

Por lo tanto, la Web Semántica requiere lenguajes de modelado mucho más flexibles que UML o los provenientes del mundo de las bases de datos. Su principal característica es que permiten modelar los recursos de forma completamente desacoplada, es decir, no están orientados a restringir de manera alguna la estructura o el formato de los recursos en sí, sino tan sólo a escribir anotaciones que permiten clasificarlos y describir sus propiedades; por supuesto, estas anotaciones suelen realizarse usando lenguajes que son fácilmente procesables mediante programación, pues esto es sin duda alguna lo que facilitará el desarrollo de aplicaciones novedosas capaces de hacer uso de la información existente en la Web

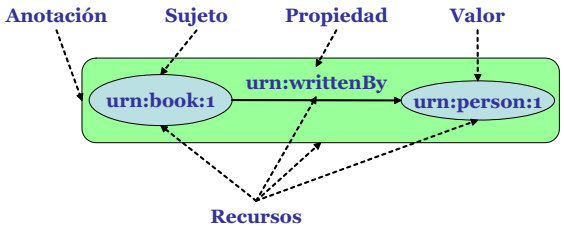

Figura 3. Estructura de las anotaciones en RDF.

Semántica. Desde 2001 han sido muchas las propuestas que han ido apareciendo, aunque la W3C tan sólo recomienda tres, RDF, RDFS y OWL, que son las que examinaremos con algo más de detalle en las siguientes secciones.

3.1 Resource Description Framework (RDF)

RDF es un lenguaje para escribir anotaciones acerca de recursos que se encuentran en la Web. Uno de estos recursos es cualquier artefacto que cuente con una URI, por lo que RDF nos permite escribir anotaciones sobre páginas web, sobre fragmentos de las mismas, sobre imágenes, documentos PDF, animaciones Flash o incluso servicios web. Además, dado que estas anotaciones se suelen almacenar como recursos en la Web, es posible escribir anotaciones sobre otras anotaciones, lo que proporciona una gran potencia al lenguaje.

Las anotaciones en RDF son tripletas que constan de un sujeto, que es el recurso al que hace referencia la anotación, una propiedad, que se define mediante otro recurso generalmente en RDFS u OWL, y un valor, que puede ser un dato simple, por ejemplo, una cadena, un entero o una fecha, o bien otro recurso, como se ilustra en la figura 3. En este ejemplo hemos usado las URIs `urn:book:1`, `urn:writtenBy` y `urn:person:1` por simplicidad; en un caso real, `urn:book:1` podría hacer referencia al recurso disponible en `http://en.wikipedia.org/wiki/The_Da_Vinci_Code`, `urn:person:1` al recurso disponible en la dirección `http://www.danbrown.com` y `urn:writtenBy` a una propiedad descrita en una ontología disponible también en algún lugar de la Web. Usando estas tripletas tan sencillas es posible construir grafos tan elementales como el de la figura 3 u otros muchísimo más complejos, como por ejemplo las bases de datos bibliográficas Bibsonomy o TDG Scholar.

La notación gráfica que hemos usado en la figura 3 es perfecta para visualizar grafos de tripletas RDF, pero no es la más apropiada para ser tratada por ordenador. Por este motivo, existen notaciones alternativas para representar RDF de forma textual, por ejemplo Turtle o N3, aunque, sin duda alguna, la más habitual es una notación basada en XML. Además, la W3C también recomienda el uso de RDFa, que es una extensión de XHTML que permite embeber las anotaciones RDF directamente en las páginas web.

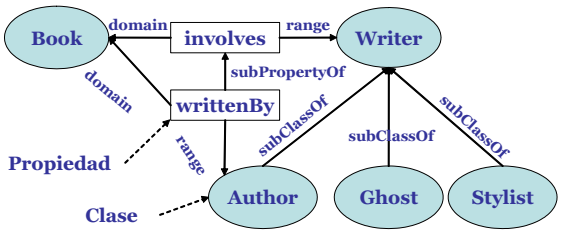

Figura 4. Un modelo sencillo en RDFS.

3.2 RDF Schema (RDFS)

Con RDF podemos escribir anotaciones sobre recursos existentes en la Web, pero es muy importante que tenga en cuenta que estas anotaciones no tienen ningún tipo, en el sentido de los lenguajes de programación o los lenguajes de modelado al estilo de UML. Para describir estos tipos se puede usar el lenguaje RDFS, cuyos ingredientes básicos son muy simples: definiciones de clases y definiciones de propiedades, que a su vez se pueden refinar para dar lugar a subclases o subpropiedades.

La figura 4 muestra un modelo RDFS sencillo que define los conceptos que hemos empleado en los ejemplos anteriores: `Writer`, que es la clase de los recursos que tienen información sobre escritores, y `Book`, que es la clase de los recursos con información sobre libros, así como dos propiedades denominadas `involves`, que relaciona un libro con los escritores involucrados en el mismo, y `writtenBy` que relaciona una publicación con sus autores. En el mundo editorial es frecuente que autores en una misma publicación participen varios tipos de escritores a los que se les suele denominar `Author`, el autor que aparece en la portada de la publicación, `Ghost`, que son aquéllos que trabajan en la sombra documentando a los autores o incluso escribiendo partes de un libro para ellos, y `Stylist` que son las personas que se encargan de pulir el estilo del texto escrito.

Fíjese en que al igual que `Author`, `Ghost` y `Stylist` son subclases de `Author`, `writtenBy` es una subpropiedad de `involves`. Esta suele ser una diferencia importante con respecto a otros lenguajes de modelado, que tan sólo permiten especializar las clases, pero no las propiedades. Además, al igual que RDF, lo más habitual es escribir RDFS en formato XML para que se pueda procesar fácilmente mediante programación, no en el formato gráfico de la figura 4.

3.3 Web Ontology Language (OWL)

Como puede comprobar, RDFS es un lenguaje tremendamente sencillo, por lo que las ontologías que se pueden describir con él son también muy elementales. Cuando necesitamos capacidades de modelado más avanzadas tenemos que utilizar un lenguaje denominado OWL.

OWL permite definir clases de recursos utilizando operadores booleanos, clases disjuntas y enumeradas. Los operadores booleanos nos permiten definir

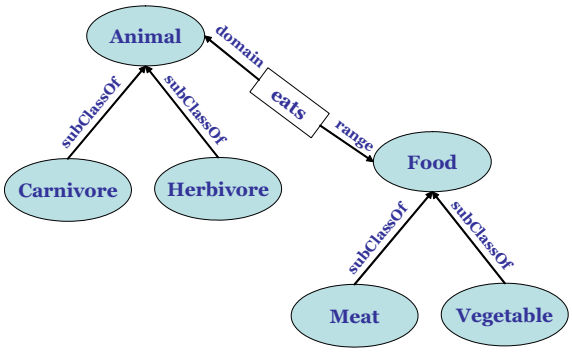

Figura 5. Un modelo que requiere ámbito no local de las propiedades.

por ejemplo, el conjunto de los recursos que describen publicaciones como el conjunto de recursos de tipo `Book` unido con el conjunto de recursos de tipo `Article` o el conjunto de recursos que describen vehículos anfibios como la intersección entre los recursos de tipo `TerrestrialVehicle` y `AquaticVehicle`. Las clases disjuntas son aquéllas tales que no comparten ningún recurso. Finalmente, las clases enumeradas son aquéllas cuyos recursos forman un conjunto finito conocido de antemano, por ejemplo la clase de los directores de una editorial.

Con respecto a las propiedades, OWL permite distinguir entre propiedades dato y propiedades objeto: las primeras son aquéllas cuyo rango es un dato simple como un entero, un booleano, una fecha y otros tipos de datos sencillos definidos en XML Schema; las segundas son aquellas propiedades que relacionan los recursos entre sí. Además, también es posible poner restricciones en la cardinalidad de algunas propiedades para indicar, por ejemplo, que un recurso de tipo `Book` sólo puede tener una propiedad de tipo `name`. OWL también permite indicar qué propiedades son reflexivas, simétricas, transitivas o inversas, por poner algunos ejemplos, y refinar sus dominios, a lo que a veces se hace referencia como ámbito no local de las propiedades. Esta última característica es muy importante en aquellos contextos en los que existen jerarquías de clases; por ejemplo, en la figura 5 se muestra un modelo muy sencillo en el que se recogen cuatro tipos de recursos denominados `Carnivore`, `Herbivore`, `Meat` y `Vegetable` que representan a los animales carnívoros y herbívoros, así como a la comida de tipo la carne y la comida de tipo vegetal; evidentemente, ambos tipos de animales son de la clase más general `Animal` y ambos tipos de comida de la clase más general `Food`. Fíjese en que hemos definido una propiedad llamada `eat` que permite relacionar un recurso de tipo `Animal` con un recurso de tipo `Food`; si dejamos el modelo así, podríamos llegar fácilmente a una situación en la que un león se alimenta de zanahorias, puesto que nada en el modelo impide que un recurso de tipo `Carnivore` esté relacionado con uno de tipo `Vegetable` mediante una propiedad de tipo `eats`. Para evitar este problema, OWL permite refinar las propiedades dentro de la jerarquía de clases de forma que podamos dejar claro que los carnívoros comen carne y los herbívoros comen vegetales. Además, OWL

también tiene algunas características que están orientadas a la propia gestión de las ontologías, como son, por ejemplo, la posibilidad de versionarlas y de importar unas en otras.

Como fácilmente puede hacerse una idea, OWL es un lenguaje complejo que permite definir modelos muy ricos y detallados de los recursos que proporciona la Web; esta gran expresividad va asociada a un coste computacional alto, por lo que suele ser habitual hacer una distinción explícita entre tres lenguajes dentro de OWL, a saber:

OWL Full: Este es el nombre que se da al lenguaje completo, con todas sus capacidades de modelado.

OWL DL: Esta versión proporciona un subconjunto de capacidades de modelado que se pueden expresar utilizado lógica de descripciones. Las restricciones más importantes son las siguientes: un recurso no puede ser al mismo tiempo el sujeto de una tripleta, por ejemplo, y una propiedad en otra tripleta; todos los recursos deben tener un tipo explícito; el conjunto de propiedades dato y propiedades objeto es disjunto; no se pueden poner restricciones de cardinalidad en las propiedades transitivas. Existen algunas otras restricciones adicionales, pero con éstas es más que suficiente para darse cuenta de que las limitaciones no son demasiado grandes, por lo que OWL DL es en la práctica una de las versiones más usadas del lenguaje.

OWL Lite: Esta versión es la menos potente. Además de las restricciones de OWL DL, también se ponen las siguientes: no se permiten clases enumeradas, ni disjuntas, ni tampoco operadores booleanos para definirlas; la única restricción de cardinalidad para las propiedades es cero o uno (propiedades opcionales). Existen algunas restricciones más que no hemos comentado, pero, de nuevo, se trata de un lenguaje más que suficiente para muchos tipos de aplicaciones, por lo que también es muy usado en la práctica.

4 Inferencia

Los lenguajes que hemos comentado en la sección anterior nos permiten anotar los recursos existentes en la Web actual, lo que suele ser suficiente para muchas aplicaciones que trabajan con la información que reside en ella. El siguiente paso es realizar inferencia sobre dicha información, pues, no olvidemos, este es el segundo ingrediente del entendimiento.

La inferencia es importante para determinar la consistencia de un modelo o de un conjunto de recursos en relación con un modelo, para clasificar los recursos, es decir, para asignarles tipos, también para correlacionar, es decir, encontrar qué recursos hacen referencia al mismo objeto del mundo real o qué modelos se pueden considerar equivalentes, y, finalmente, para extrapolar, es decir, para obtener nuevo conocimiento a partir del que ya conocemos.

Para realizar inferencias se hace uso de motores que están basados en algún tipo de lógica, por ejemplo: proposicional, de Horn, de descripciones, de primer orden, de orden superior, temporal, modal, causal, probabilística, etcétera. La

diferencia más importante entre estas lógicas es su capacidad expresiva: de forma general, cuanto más expresiva sea una lógica más ineficientes son las herramientas que trabajan con ellas. Por este motivo no es de extrañar que al final el tipo de lógica más usado sea el más sencillo; en concreto, las lógicas basadas en reglas como la siguiente:

$$\overbrace{A_1, A_2, \ldots, A_n}^{\text{Antecedente}} \Longrightarrow \overbrace{B}^{\text{Consecuente}}$$

En general, se distinguen dos tipos de reglas:

Monótonas: Cuando se realiza inferencia con ese tipo de reglas, tenemos la garantía de que ninguna de las conclusiones alcanzadas en un proceso de inferencia podrá ser contradicha por un proceso de inferencia posterior.

No monótonas: En este caso, un proceso de inferencia puede invalidar las conclusiones alcanzadas en un proceso de inferencia anterior.

En la práctica, las reglas que parecen poder modelar de una forma más real los procesos de inferencia habituales son las no monótonas, pero suele ser más frecuente encontrar sistemas de inferencia para reglas monótonas puesto que los algoritmos para tratarlas son mucho más sencillos y mucho mejor conocidos.

Por desgracia, incluso usando una lógica basada en reglas, las cosas no son tan sencillas como pudiera parecer puesto que existen muchos lenguajes basados en reglas, por ejemplo: SWRL, RuleML, SWRL-FOL, Prolog o Jess, por citar algunos ejemplos. El problema es que ninguno de ellos parece destacar sobre los demás, por lo que la recomendación de la W3C es tan sólo en relación con un lenguaje denominado Rule Interchange Format (RIF) que está pensado para el intercambio de reglas descritas usando diferentes lenguajes. Actualmente se está trabajando intensamente en RIF, pero aún no existe una versión definitiva.

5 Preguntas

Una vez hemos modelado los recursos de la Web actual y hemos descrito las bases de nuestros procesos de inferencia usando reglas, el paso definitivo para determinar si entendemos la información que aparece en la Web es formulando preguntas.

Hasta hace poco, han existido varios lenguajes de consulta para la Web Semántica que han estado compitiendo fuertemente entre sí, pero la W3C tan sólo recomienda SPARQL, que está basado en la selección de tripletas mediante patrones. Por ejemplo, supongamos que tenemos un grafo RDF como el que aparece en la figura 6; sobre él podemos lanzar una consulta como la siguiente cuyo objetivo es recuperar el nombre de todas las personas:

```
select ?n
where { ?p name ?n }
```

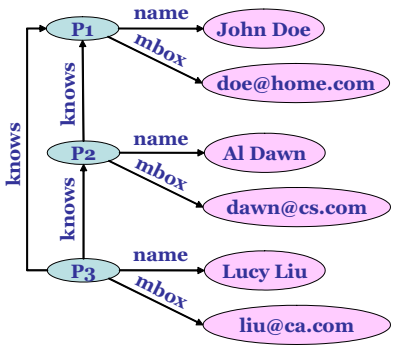

Figura 6. Un grafo RDF con información el correo electrónico de varias personas.

Los identificadores prefijados con una interrogación denotan variables. Para entender esta consulta debe fijarse en la cláusula `where`, que está formada por uno o varios patrones sobre las tripletas almacenadas en el grafo RDF que estamos consultando. En este caso concreto, el patrón es de la forma `?p name ?n`, lo que indica que se deben tener en cuenta todas las tripletas cuya propiedad sea `name`, sin imponer ningún tipo de restricción sobre el sujeto o el valor. La cláusula `select` indica qué partes de las tripletas seleccionadas se deben devolver; en este caso es `?n`, que corresponde al valor de la propiedad `name`. Por lo tanto, una consulta como la anterior debe devolver un resultado como el siguiente:

```
?n
------------------------
"John Doe"
"Al Dawn"
"Lucy Liu"
```

Demos un paso más, por ejemplo para conocer cuál es la dirección de correo de la persona llamada `Luci Liu`. Para ello debemos lanzar una consulta como la siguiente:

```
select ?mb
where {
    ?p name "Lucy Liu" .
    ?p mbox ?mb
}
```

En este caso la cláusula `where` contiene dos patrones: el primero se queda con aquellas tripletas que tienen como propiedad `name` y como valor para la misma la cadena `"Luci Liu"`; en el ejemplo de la figura 6 tan sólo hay una tripleta que cumple esta restricción, por lo que la variable `?p` queda inmediatamente ligada al recurso `p3`. El siguiente patrón selecciona aquellas tripletas que tienen como

propiedad `mbox` y como sujeto el valor de la variable `?p`, es decir, la tripleta que informa sobre la dirección de correo de la persona llamada `Luci Liu`.

Aunque SPARQL incluye características más avanzadas como son la posibilidad de realizar filtros avanzados sobre los valores de las tripletas, agrupar u ordenar los resultados, creemos que estos dos ejemplos pueden ser suficientes para tener una idea básica de cuáles son sus posibilidades. Por desgracia, de los tres niveles de preguntas que planteamos en la introducción, SPARQL tan sólo es capaz de tratar con consultas de evidencia, aunque sobre almacenes de datos mucho más grandes que el del ejemplo, claro está.

6 Microformatos y GRDDL

La idea que hasta ahora hemos transmitido en relación con la Web Semántica es la de una Web en la que los recursos tienen asociadas anotaciones que son fáciles de procesar mediante programación. Estas anotaciones pueden almacenarse como recursos persistentes en la propia Web, es decir, como recursos con su propia URI; no obstante, una alternativa muy interesante es usar microformatos y GRDDL para generar las anotaciones bajo demanda.

Los microformatos son convenios sencillos a la hora de utilizar XHTML que ayudan a entender mejor cuál es el contenido de una página web. Un ejemplo muy conocido de microformato es hCard, que permite introducir en una página información sobre una persona. Por ejemplo, lo siguiente es un fragmento de XHMTL en el que se usa este microformato:

```
...
<div class="vcard">
    <span class="fn">John Doe</span>
    <div class="org">University of Wonderland</div>
    <div class="adr">
        <div class="street-address">London Street</div>
        <span class="locality">Mariland</span>
    </div>
<div>
...
```

El convenio indica que la información de una persona debe estar dentro de una etiqueta HTML, `<div>` en este caso, de la clase `vcard`; a continuación, el convenio indica que el nombre de la persona debe encontrarse en una etiqueta de la clase `fn`, que la organización a la que pertenece debe estar en una etiqueta de clase `org`, etcétera. Estos sencillos convenios ayudan a estructurar la información que se encuentra dentro de la página web de forma que los buscadores puedan entender que la información en esta página es sobre una persona. Por supuesto, existen otros muchos microformatos para proporcionar información sobre calendarios, revisiones, currículos, etcétera.

El problema de los microformatos es que tan sólo son convenios a la hora de usar XHTML, por lo que por sí sólo tan sólo alivian un poco el problema de

Figura 7. Google ya hace uso de los microformatos.

acceder a la información que contiene una página web mediante programación. Lo normal es usarlos en combinación con GRDDL para obtener a partir de ellos anotaciones en RDF. GRDDL es una técnica recomendada por la W3C cuya idea es tan sencilla como indicar en los documentos XHTML la URL de una transformación, generalmente utilizando XSLT, que permita obtener de ellos anotaciones en formato RDF. Por ejemplo, para añadir GRDDL a la página anterior, tan sólo tendríamos que incluir en su cabecera lo siguiente:

```
...
<html xmlns="http://www.w3.org/1999/xhtml">
    <head profile="http://www.w3.org/2003/g/data-view
                   http://www.w3.org/2006/03/hcard">
        <title>John Doe's Home Page</title>
        <link rel="transformation"
              href="http://www.w3.org/2006/vcard/hcard2rdf.xsl"/>
    </head>
    <body>
 ...
```

El atributo `profile` en la etiqueta `<head>` indica que esta página contiene datos que se adaptan al microformato hCard y pueden ser tratados mediante GRDDL; en concreto, es la etiqueta `<link>` que viene a continuación la que indica el fichero con la transformación XSLT que se debe aplicar.

A la vista de lo que hemos comentado, queda claro que los microformatos o GRDDL no ofrecen una solución general al problema de proporcionar anotaciones en RDF para los recursos existentes en la Web, pues tan sólo se pueden usar con documentos XHTML, o, de forma más general, con documentos basados en

XML. No obstante, su uso está tan generalizado que muchos buscadores ya los tienen en cuenta. Por ejemplo, en la figura 7 se muestra una captura de pantalla con parte de los resultados de buscar las palabras `GRDDL hCard` en Google; el buscador ha reconocido el formato hCard en la primera de las páginas que ha encontrado y muestra automáticamente un mapa que indica dónde se encuentra la dirección a la que hace referencia esta página.

7 Conclusiones

Ha llegado el momento de recapitular sobre la definición de Web Semántica que Berners-Lee, Hendler y Lassila proporcionaron en 2001:

> La Web Semántica es una extensión de la Web actual en la que la información tiene un significado bien definido que permite a personas y ordenadores trabajar mejor de forma conjunta.

Algo que debe quedar claro es que el objetivo de la Web Semántica no es construir otra Web, sino facilitar a los programadores que puedan hacer uso de la información que reside en la Web actual. Para ello, la Web Semántica proporciona un lenguaje base denominado RDF para escribir anotaciones y dos lenguajes denominados RDFS y OWL para diseñar modelos ontológicos. Además, los microformatos y GRDDL sirven de apoyo para que las anotaciones se puedan generar de una forma muy sencilla, tan sólo siguiendo algunos convenios a la hora de escribir XHTML y haciendo uso de transformaciones XSLT.

La visión de la Web Semántica es radicalmente distinta si Ud. la ve desde la perspectiva de la Ingeniería del Software o desde el punto de vista de la Inteligencia Artificial. Desde la primera perspectiva, el término Web Semántica suele llevar fácilmente a malinterpretaciones que dan lugar a confusiones que en muchas ocasiones contribuyen a difuminar la idea de fondo. Es más que posible que desde este punto de vista un nombre mucho más adecuado hubiera sido la Web de los Datos, frente a la Web clásica que podríamos haber denominado la Web de los Documentos. El motivo es que cuando pensamos en Ingeniería del Software la Web Semántica se ve fundamentalmente como un conjunto de propuestas que permiten reducir el coste de desarrollar aplicaciones que hacen uso de los datos existentes en la Web, o, para ser más precisos, en los documentos accesibles a través de la Web. Por el contrario, si enfocamos la Web Semántica desde una perspectiva de Inteligencia Artificial, seguramente el énfasis estará en lograr llevar a cabo procesos de inferencia muy complejos haciendo uso de la información que proporciona la Web Semántica. En ambos casos la clave es facilitar el acceso a la información en la Web.

Ambas perspectivas son complementarias, sin duda alguna, aunque en este documento hemos puesto el énfasis en la primera. Por este motivo, le recomendamos encarecidamente que continúe profundizando en este campo tomando como punto de partida la bibliografía y las herramientas que recomendamos en la siguiente sección. Aprenda y obtenga sus propias conclusiones, pero, sobre todo, no olvide que, como todo en el terreno de las tecnologías, tan sólo estamos

hablando de herramientas. Lo realmente importante es que encuentre aplicaciones novedosas de la misma que tengan una gran utilidad para la sociedad.

A Bibliografía y Herramientas

Podrá encontrar el artículo de Berners-Lee, Hendler y Lassila que sirvió como impulso inicial para el desarrollo de la Web Semántica en la referencia [7]. En 2006, Shadbolt, Berners-Lee y Hall revisaron las ideas de este primer artículo y escribieron otro que puede servir como retrospectiva sobre los logros conseguidos y los pasos que aún queda por dar en este campo [21].

El artículo en el que Gruber presentó su definición del término Ontología en el contexto de la informática podrá encontrarlo en la referencia [14]; la versión ampliada que pone énfasis en la necesidad de que los modelos ontológicos sean compartidos podrá encontrarla en la referencia [15].

Uno de los libros más interesantes para introducirse en el mundo de la Web Semántica es el de Antoniou y van Harmelen [4]. Es muy didáctico, proporciona una bastante clara de los conceptos e incluye un capítulo dedicado a describir aplicaciones reales de la Web Semántica en empresas como Elsevier o Audi; además, proporciona una introducción amplia a XML, RDF, RDFS, OWL y SPARQL. Si quiere profundizar en XML, le recomendamos el libro de Young [26]; para profundizar RDF, RDFS y OWL le recomendamos el libro de Allemang y Hendler [2]; por desgracia, en el momento de escribir este documento no podemos recomendarle ningún libro sobre SPARQL, por lo que quizá la mejor forma de profundizar sea echar un vistazo al artículo de Dodds en la referencia [11]. Si está interesado en los lenguajes de consulta tanto para la Web clásica como para la Web Semántica, entonces seguro que estará interesado en el informe de Bailey, Bry, Furche y Schaffert [6]. Para aprender algo más sobre RDFa, la extensión de XHTML que permite embeber RDF en una página web puede tomar como punto de partida la referencia [1].

Para poder conocer mejor los lenguajes de la Web Semántica, necesitará también algunas herramientas. Le recomendamos que empiece por Protégé, que es una de las más populares [22]. Con esta herramienta podrá experimentar creando anotaciones en RDF y ontologías tanto en RDFS como en OWL. Recuerde que una característica de toda buena ontología es que debe ser compartida, por lo que quizá también esté interesado en buscar en la Web ontologías ya existentes; para ello puede usar motores de búsqueda especializados como Swoogle [12] o Watson [18]. Si también está interesado en conocer algo más sobre los sistemas de inferencia para la Web Semántica, entonces seguro que le vendrá bien echar un vistazo a la referencia [20], que proporciona información sobre un gran número de razonadores basados en lógica de descripciones [5,13]. Si lo que desea es tener acceso a una fuente de anotaciones bastante amplia sobre publicaciones científicas, entonces le recomendamos Bibsonomy [17] o TDG Scholar [23].

La fuente de conocimiento más actualizada sobre microformatos es la referencia [19], aunque también hay un libro disponible que ha sido escrito por Allsopp [3]. Con respecto a GRDDL no podemos recomendarle ningún libro,

pero sí la referencia [16], que contiene una pequeña introducción a la técnica, y el libro de Tidwell sobre XSLT [24], que es una de las bases fundamentales para dominar GRDDL [24]. Actualmente existen varios plugins que permiten a los navegadores leer microformatos; si está interesado en este tema, puede echarle un vistazo a las referencia [8].

Si está interesado en conocer algo más sobre los sistemas de wrapping puede consultar las siguientes referencias fundamentales [9,25] y un reciente número especial de la revista J.UCS [10]. Recuerde de la introducción a este documento, que los sistemas de wrapping suelen proporcionar soluciones efectivas para extraer información estructurada de las páginas web que no cuentan con anotaciones ni tampoco microformatos. Mientras la Web Semántica se establece definitivamente, este tipo de sistemas es, sin duda alguna, la única alternativa viable en muchos problemas relacionados con integrar fuentes de información web en procesos informáticos.

Además de este documento y de las referencias anteriores, quizá esté interesado en ver una presentación relacionada que podrá encontrar en `http://www.tdg-seville.info/projects/IntegraWeb/semi-19-01-07.ppt` y un vídeo sobre la misma que se encuentra disponible en `http://www.tdg-seville.info/projects/IntegraWeb/semi-19-01-07.asf`.

Referencias

1. B. Adida and M. Birbeck. RDFa primer, 2008. Disponible en `http://rdfa.info/wiki/RDFa_Wiki`.
2. D. Allemang and J. Hendler. *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL*. Morgan Kaufmann, 2008.
3. J. Allsopp. *Microformats: Empowering Your Markup for Web 2.0*. Friends of Ed (Apress), 2007.
4. G. Antoniou and F. van Harmelen. *A Semantic Web Primer (2^a edición)*. The MIT Press, 2008.
5. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2003.
6. J. Bailey, F. Bry, T. Furche, and S. Schaffert. Web and semantic web query languages: A survey, 2005. Disponible en `http://www.cs.mu.oz.au/~jbailey/papers/PMS-FB-2005-14.pdf`.
7. T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, 2001. Disponible en `http://www.sciam.com/article.cfm?id=the-semantic-web`.
8. Bookmarklets home page, 2008. Disponible en `http://microformats.org/wiki/bookmarklets`.
9. C. Chang, M. Kayed, M.R. Girgis, and K.F. Shaalan. Survey of web information extraction systems. *IEEE Transactions on Knowledge and Data Engineering*, 18(10):1411–1428, 2006.
10. R. Corchuelo, J.L. Arjona, and D. Ruiz. Special issue on wrapping web data islands. *Journal of Universal Computer Science*, 14(11), 2008.

11. L. Dodds. Introducing SPARQL: Querying the Semantic Web, 2008. Disponible en http://www.xml.com/pub/a/2005/11/16/introducing-sparql-querying-semantic-web-tutorial.html.
12. eBiquity Research Group. Swoogle home page, 2008. Disponible en http://swoogle.umbc.edu.
13. E. Franconi. Description logics, 2008. Disponible en http://www.inf.unibz.it/~franconi/dl/course.
14. T.R. Gruber. A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2):199–220, 1993. Disponible en http://tomgruber.org/writing/ontolingua-kaj-1993.htm.
15. T.R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies*, 43(4–5):907–928, 1995. Disponible en http://tomgruber.org/writing/onto-design.htm.
16. H. Halpin and I. Davis. GRDDL primer, 2008. Disponible en http://www.w3.org/TR/xhtml-rdfa-primer.
17. Knowledge and Data Engineering Group. Bibsonomy, 2008. Disponible en http://www.bibsonomy.org.
18. Knowledge Media Institute. Watson home page, 2008. Disponible en http://watson.kmi.open.ac.uk/.
19. Microformats home page, 2008. Disponible en http://www.microformats.org.
20. U. Sattler. Description logic reasoners, 2008. Disponible en http://www.cs.man.ac.uk/~sattler/reasoners.html.
21. N. Shadbolt, T. Berners-Lee, and W. Hall. The semantic web revisited. *IEEE Intelligent Systems*, 21(3):96–101, 2006.
22. Stanford Center for Biomedical Informatics Research. Protégé home page, 2008. Disponible en http://protege.stanford.edu.
23. The Distributed Group. TDG Scholar home page, 2009. Disponible en http://scholar.tdg-seville.info.
24. D. Tidwell. *XSLT (2^a edición)*. O'Reilly Media, 2008.
25. J. Turmo, A. Ageno, and N. Català. Adaptive information extraction. *ACM Computing Surveys*, 38(2):#4, 2006.
26. M.J. Young. *XML Step by Step (2^a edición)*. Microsoft Press, 2002.