

H.1. Los cms como pieza fundamental en el despliegue de la web semántica

Juan-Antonio Pastor-Sánchez

18 julio 2011

Pastor-Sánchez, Juan-Antonio. "Los cms como pieza fundamental en el despliegue de la web semántica". *Anuario ThinkEPI*, 2012, v. 6, pp. 184-189.



Resumen: La web semántica tiene suficiente masa crítica para ampliar el alcance de sus tecnologías a los sistemas de gestión de contenidos (cms). Es posible detectar un aumento de la aplicación de estándares interoperables para publicar grandes conjuntos de datos. Rdf, owl y sparql están sustituyendo los desarrollos clásicos basados en sistemas de gestión de bases de datos relacionales, para la gestión y el almacenamiento de información. Los cms comienzan a utilizar dichas tecnologías para publicar contenidos en formatos de la web semántica y definir mapeados, con elementos de vocabularios rdf. En este sentido, también incorporan funciones para añadir información semántica en el código xhtml, utilizando rdfa, y usar sparql para permitir la consulta de datos o para incorporar datos externos.

Palabras clave: Cms, Linked open data, Rdf, Rdfa, Sistemas de gestión de contenidos, Sparql, Web semántica.

Title: Content management systems as a cornerstone in the deployment of semantic web

Abstract: The semantic web has enough critical mass to expand its technological scope to content management systems (cms). Currently, an increase in the application of interoperable standards to publish large datasets is detected. Rdf, owl and sparql are replacing traditional developments based on relational database management systems for management and storage of information. Cms are beginning to use these technologies to publish content with semantic web formats, and to define mappings with elements of rdf vocabularies. In this regard, Cms also incorporate features to add semantic information into the xhtml code using rdfa and applying sparql to query data or incorporate external data.

Keywords: Cms, Content management systems, Linked open data, Rdf, Rdfa, Semantic web, Sparql.

1. Introducción

Hace ya algún tiempo que las tecnologías de la web semántica (**Pastor-Sánchez**, 2011) han creado la suficiente masa crítica que permite pasar de una fase de laboratorio a otra de explotación. Algunas iniciativas como *linked open data* destacan por su atractivo y empuje, al tiempo que la incorporación de muchas instituciones a este nuevo mundo precisa de inversiones para el desarrollo de nuevos proyectos. Por tanto se necesitan herramientas para representar datos, información y conocimiento, que permitan la interoperabilidad de sistemas de forma precisa, eficiente y escalable.

Este cambio conlleva una evolución en la política de gestión de información en las organizaciones. Para adquirir o implantar estas nuevas tecnologías no se precisan grandes inversiones, sino modificaciones en la planificación estratégica de los servicios y productos de información, así como en las aplicaciones a utilizar. Estamos ante

un nuevo escenario que precisa de un cambio de mentalidad corporativa y que se apoya en tres factores determinantes:

1) El auge del software libre y de código abierto ya permite afianzar un modelo de creación de herramientas y aplicaciones de calidad, con el compromiso de comunidades de usuarios dedicadas a su elaboración y actualización (**Fitzgerald**, 2006). Ya no es sostenible el viejo argumento de que este tipo de software no ofrece garantías en su uso. El código abierto permite una revisión de las funciones de los programas y la detección de problemas de seguridad. Por otro lado, las posibilidades de adaptabilidad e interoperabilidad, uso de estándares y especificaciones abiertas, la independencia de proveedores y el ahorro en los costes de adquisición de licencias, suponen una serie de ventajas obvias.

2) El desarrollo en el ámbito de la Web de estándares y especificaciones abiertas para la representación de datos de un modo transparente y evolutivo a través de niveles de abstracción

cada vez mayores. Su creación y mantenimiento es realizada por comunidades participativas de usuarios (formadas por investigadores, docentes, programadores y empresas), cuyo trabajo es coordinado y supervisado por el W3C. Se trata de un modelo colaborativo de desarrollo tecnológico de largo recorrido y que ha dado frutos tales como xml, rdf, owl, skos o sparql¹.

3) La difusión abierta de datos y contenidos, como parte de una política de productos y servicios de información de las organizaciones. Este aspecto se está llevando a cabo en administraciones públicas, instituciones culturales, empresas, proyectos de investigación o iniciativas de comunidades de usuarios. Para la administración pública resulta fundamental compartir datos según unos criterios de interoperabilidad, que faciliten la comunicación transparente con ciudadanos y empresas (Alanis et al., 2007 y Ding et al., 2010). Grupos como el *Government Linked Data (GLD) Working Group* o iniciativas como *Europeana* parten de un apoyo claro a este tipo de iniciativas.

Muchas instituciones han encontrado en este nuevo contexto la solución a problemas que bloqueaban el avance de proyectos de colaboración en comunidades heterogéneas de usuarios². Estas tecnologías permiten la coordinación a nivel institucional, ya sea adoptando esquemas de descripción compartidos, o utilizando soluciones propias que puedan mapearse entre sí³.

Dentro del diverso grado de evolución de los elementos de la web semántica, rdf se vislumbra por su solidez y grado de implantación como el instrumento para representar vastas cantidades de datos. Éstos pueden ser utilizados directamente por aplicaciones sin que se tengan que realizar costosos procesos de conversión o traducción, puesto que se utilizan mecanismos compartidos de representación.

“Rdf se vislumbra por su solidez y grado de implantación como el instrumento para representar vastas cantidades de datos”

2. Definiendo el campo de batalla: infraestructura de software para la integración de los cms en la web semántica

La mayor parte de contenidos de la Web no se encuentra en formatos adecuados para la web semántica. En este sentido, la Web con la que trabajan las personas precisa de una interfaz para

la consulta de información en donde las técnicas de accesibilidad, usabilidad y arquitectura de la información son fundamentales. Así pues, ¿debe hablarse a partir de ahora de dos Webs separadas, una para personas y otra para máquinas?, ¿se formarán dos universos paralelos de desarrollo de aplicaciones y tecnologías pero separados para contenidos en html y datos rdf? Aquí es donde los sistemas de gestión de contenidos (cms) tienen mucho que decir.

Una de las principales características de estos sistemas radica en la superación del modelo de edición centrado en la gestión de ficheros a otro más cercano a procesos de organización y estructuración de contenidos definidos a partir de criterios de arquitectura de la información. También deben tenerse en cuenta los cms especializados, cuyo campo de trabajo es la creación de repositorios digitales, la gestión de foros de debate, wikis o plataformas educativas (*learning cms*, *lcms*).

“¿Debe hablarse a partir de ahora de dos Webs separadas, una para personas y otra para máquinas?”

Este paradigma sólidamente establecido de gestión de sitios web se fundamenta en la definición de taxonomías, tipos de contenidos, sistemas de navegación facetados, uso de plantillas de presentación, reutilización de objetos e información, y conexión con fuentes de datos externas. Además, la mayoría de estos sistemas utiliza bases de datos relacionales (sgbdr) para almacenar los componentes informativos de los contenidos y otros aspectos operativos, separando estructura, información y presentación.

Puesto que tenemos información –con un alto grado de estructuración– almacenada en bases de datos, es razonable plantear la cuestión de si es posible utilizar un cms para:

- reutilizar los datos almacenados, utilizados en principio para la elaboración de las páginas de un sitio web, e incorporarlos a la web semántica;
- añadir información semántica procesable por máquinas dentro de un contenido pensado inicialmente para su consulta por personas;
- explotar fuentes externas de datos rdf para enriquecer los contenidos de un sitio web (Heath; Bizer, 2010).

Como se ha mencionado anteriormente, la mayoría de cms utilizan sgbdr para almacenar los datos. La web semántica se basa en el uso de un modelo de descripción totalmente distinto (rdf) que difiere considerablemente del modelo

relacional (Allemand; Hendler, 2011, p. 23; Velegrakis, 2010).

El inconveniente es que el uso de soluciones relacionales para el soporte de base de datos del cms (la inmensa mayoría: *MySQL*, *MariaDB*, *PostgreSQL*, *Oracle*, *SQLite*, *SQL Server*, etc.) no permite operar directamente con rdf. Es necesario utilizar una capa intermedia que permita almacenar datos rdf en tablas relacionales. De este modo se facilita la elaboración de aplicaciones que utilicen la misma infraestructura de sgbd con la que funciona el cms al tiempo que se trabaja con datos rdf. Las aplicaciones informáticas solucionan este problema utilizando una capa de abstracción que suele tomar la forma de librerías o de *middleware* (Bishop; Karne, 2003), que en este caso interactúan con el cms mediante algún tipo de api en combinación con *sparql* para recuperar datos rdf.

Esta integración de software resulta compleja en las fases iniciales, pero una vez que se pone en marcha y se comprenden los principios básicos de funcionamiento y gestión resulta más sencilla. Las funciones de los cms suelen ampliarse agregando módulos o *plugins* que hacen uso de las librerías o del *middleware* para realizar procesos encargados del almacenamiento de datos en un *triplestore*⁴, uso y definición de *Sparql endpoints* para obtener y exponer datos, parseado (análisis), extracción y conversión de rdf.

En entornos php es muy popular el uso de *arc*, que proporciona una potente librería para la manipulación de datos rdf y su almacenamiento en un sgbd. En el caso de java debemos referirnos a *Jena*, que además ofrece un motor de inferencia basado en owl. Si buscamos cierta independencia con respecto al soporte de lenguaje de programación web utilizado puede utilizarse *Sesame*, aunque la solución más potente y versátil tal vez sea *Virtuoso universal server* que combina un servidor web, con un *framework* de desarrollo/ejecución de aplicaciones y un completo soporte de base de datos y gestor de datos estructurados. *Virtuoso* puede almacenar los datos directamente en rdf, pero permite “virtualizar” cualquier sgbd, por lo que podríamos instalar y manejar un cms de forma totalmente transparente, y ampliar sus funciones para la web semántica mediante módulos de terceros o desarrollos propios.

3. Publicar, semantizar y reutilizar: un camino de ida y vuelta

Una de las primeras tareas que se desea realizar con un cms en el ámbito de la web semántica es ofrecer contenidos en rdf. La explotación de información generada por una aplicación web se realiza utilizando protocolos para el intercambio

de datos. Es decir, una aplicación se conecta a una url y accede a un conjunto de datos que se encuentran generalmente en formato xml. Rdf puede codificarse con xml (rdf/xml) y es una primera aproximación a la publicación en la web semántica de los contenidos gestionados por un cms.

Otro método es la creación y uso de *Sparql endpoints*. Se trata de servicios web que utilizan *sparql* como lenguaje de consulta y como protocolo para obtener los datos rdf recuperados. Podríamos decir que se trata del sql de rdf. Lo interesante es que pueden ser utilizados por personas o por aplicaciones. De este modo no siempre es necesaria la publicación de vastos conjuntos de datos rdf/xml accesibles a través de un url para que otras aplicaciones puedan descargar ficheros y reutilizar los datos de nuestro sitio. Es posible acceder a información más específica, enviando una consulta al *Sparql endpoint* que a su vez devuelve los datos en un formato determinado. Este mecanismo proporciona mayor flexibilidad en el desarrollo de aplicaciones recolectoras de datos que el uso de protocolos como oai-pmh con un conjunto limitado de instrucciones.

Ya se utilice una u otra opción, es necesario mapear los elementos de los contenidos gestionados por el cms a sus correspondientes elementos de vocabularios rdf. Aquí reside la dificultad para que muchos cms aporten contenidos a la web semántica y participen en el universo *linked data*. No se trata de “volcar” tablas de una base de datos relacional a rdf/xml y por supuesto estamos buscando algo más que ofrecer canales o fuentes rss. El proceso es más complicado, puesto que hay que tomar decisiones: qué vocabulario es el más adecuado para representar un determinado campo de un tipo de contenido, qué elemento debe utilizarse, qué nivel de detalle y agregación, etc.

Algunos cms como *Liferay* y *Drupal* disponen de soluciones que permiten publicar datos utilizando rdf e incluso ofreciendo *Sparql endpoint*. En el caso de *Liferay* existen algunas soluciones, como el módulo *Liferay linked data*, que precisa de *D2R-server*. *Drupal* ofrece soporte rdf en su núcleo que se amplía mediante la combinación de la librería *arc* y módulos, como *RDF extensions* y *Views datasource* o *Restful web services*, que permiten publicar nodos individuales o vistas en cualquier formato (entre los cuales se encuentra rdf/xml), así como la definición de *Sparql endpoint* limitado al contenido de nodos y taxonomías.

Participar en la web semántica no conlleva participar en la corriente *linked open data*. De hecho, *linked data* utiliza un subconjunto de las tecnologías implementadas al amparo de la web semántica. Cualquier sitio web puede incorporar información semántica dentro de una página html⁵. Para ello debemos tener en cuenta que

rdf es para las máquinas lo que html para las personas. Pero, ¿existe alguna forma de que ambas versiones coexistan en una única url de la que personas y máquinas puedan consultar y extraer información respectivamente?

Es posible “insertar” declaraciones rdf dentro del código html de una página. Para ello podemos utilizar rdfa, de forma que es posible reutilizar el código html para insertar información semántica (Pemberton, 2009).

Un ejemplo: para indicar el título de un documento en su contenido informativo se usaría el siguiente código:

```
<h1>Título del documento</h1>
```

con rdfa el código podría quedar:

```
<h1 about="http://ejemplo.org/documento.html" property="dc:title">Título del documento</h1>
```

“La incorporación a nuestro sitio web de datos externos es el aspecto mejor resuelto en la mayoría de los cms”

Su funcionamiento es muy similar al de los microformatos, pero en este caso se utiliza rdf como modelo para describir un elemento de información del contenido. A través de aplicaciones conocidas como *RDFa extractors* es posible obtener los datos rdf del propio código html para su posterior procesamiento.

Otra de las posibilidades de la web semántica es la incorporación a nuestro sitio web de datos externos. Este es el aspecto mejor resuelto en la mayoría de los cms. No importa el sistema utilizado: *Joomla, Wordpress, Drupal, OpenCMS, Liferay, eZ Publish, Mediawiki, phpBB, Plone...* Todos disponen de algún módulo o incorporan alguna función para recuperar, procesar, formatear e interactuar con datos rdf. El principal motivo es que la mayoría de los cms incorporan en su instalación básica herramientas para la sindicación de fuentes rss, entre cuyos formatos encontramos rss 1.0 (*rdf site summary*).

El cms siempre puede descargar un fichero de datos rdf. Pero un funcionamiento eficiente precisa de los refinamientos adicionales que ofrece sparql. *Typo3, Drupal y Liferay* destacan en este apartado. En los tres casos el funcionamiento de los módulos es muy similar: se registra en el sistema un *Sparql endpoint* externo y a continuación se utiliza el lenguaje de consulta sparql para establecer los criterios de selección de los datos externos. Cuando se lanza la consulta se recuperan una serie de datos que a partir de ese momento se manipulan como si fueran con-

tenidos generados por el propio cms. Por regla general la recuperación de datos y por tanto su visualización se realiza de forma dinámica. Es decir, los datos no suelen incorporarse al cms, sino que éste actúa de interfaz para su recuperación y visualización. Sin embargo, en algunas circunstancias estos datos se utilizan para crear nuevos contenidos gestionados por el cms, generalmente por motivos de rendimiento. En este caso el sistema actúa de un modo similar a como lo hace un agregador oai-pmh.

4. Alcanzar metas mediante la integración de componentes: simplificar lo aparentemente complejo

El panorama descrito muestra un conglomerado de sistemas, arquitecturas, elementos de software y especificaciones que funciona. Los cms son capaces de publicar datos, reutilizar fuentes externas rdf e incluir información semántica en sus contenidos. Incluso podrían ser el soporte para crear contenidos y modelos de interacción que se adapten a diferentes tipos de plataformas y dispositivos. La combinación de datos rdf interrelacionados, su integración en contenidos web y la adaptación de su consulta a las posibilidades del dispositivo final del usuario abren un mundo nuevo de ideas y aplicaciones. Es decir, se va más allá de la adaptación de contenidos a través de hojas de estilo css personalizadas para cada dispositivo: la información marcada semánticamente permite su reutilización hasta el límite que marque el nivel de detalle de los datos que la conforman.

En este sentido no sería descabellado pensar en aplicaciones avanzadas para un *smartphone* que utilicen información marcada semánticamente y que puedan indicar, por ejemplo, qué restaurante es el mejor valorado en un radio de 50 metros. O bien otras aplicaciones para planificar un trayecto entre dos ciudades que precise el uso de varios medios de transporte y realizar la compra de los billetes correspondientes. Incluso acceder a imágenes de obras y documentos sobre Leonardo da Vinci de cualquier biblioteca o museo de Europa según unos criterios de búsqueda.

Un mismo contenido podría interactuar con el usuario de un modo diferente en función de si se visualiza con una pantalla táctil, si se cuenta con una superficie de visualización amplia o si el dispositivo utilizado dispone de gprs. Son algunos ejemplos de aplicaciones que mostrarían datos de un modo atractivo y totalmente transparente al usuario, que no tiene por qué saber que la información consultada se basa en datos rdf o en la ejecución de inferencias.

A la pregunta “¿puede utilizarse un único cms para realizar todas estas funciones y conseguirlo de un modo sencillo?”, no sería atrevido contestar con un “no”, pero con ciertos matices.

Una vez adquirida una visión de conjunto de las posibilidades de los cms en la web semántica, es difícil no tener una sensación de dispersión de tecnologías y sistemas. La existencia de diferentes entornos de desarrollo, por ahora, como son php y java, complica aún más si cabe la situación. Existen diversas soluciones, entre las cuales las basadas en java están obteniendo una mayor aceptación en el desarrollo de proyectos de envergadura, asociados a la creación de repositorios, iniciativas *linked data* y la publicación y manejo de grandes conjuntos de datos rdf.

Por contra los cms más utilizados actualmente y que ofrecen una mayor flexibilidad y capacidad de personalización se basan en php. Además, la realidad en algunas organizaciones (universidades, administración regional, proyectos europeos) tiende también a la dispersión tecnológica. Incluso cms o gestores de portales basados en java carecen de un soporte nativo global para una integración completa en la web semántica.

Como puede verse, es una situación a veces incomprensible, que produce una brecha entre diferentes tipos de necesidades (publicar contenidos y publicar datos), que conduce a su vez a contemplar de un modo separado la gestión de sitios web, el potencial de datos que ofrece la web semántica, y la integración de éstos en aquellos, y viceversa.

Y por si esto fuera poco, las soluciones basadas en *plugins* o módulos de ampliación de las funciones de los cms aportan mayor confusión. No es inusual encontrarse con un complejo ecosistema de módulos, en donde no se termina de resolver una necesidad de forma completa e integral con una única opción. A veces hay que instalar varios módulos que interactúen con un *middleware* específico y actualmente muchas de estas soluciones se encuentran en fase beta.

Pese a todo lo anterior, es muy posible que se alcance una simplificación del escenario actual. Se observa una tendencia a la integración en el núcleo de algunos cms de ciertas funciones de la web semántica. Las nuevas versiones de *OpenCMS*, *Drupal* o *Liferay* “absorben” o integran módulos y *plugins* relacionados con la web semántica. Así pues, conforme se produzca esta integración, la optimización y eficiencia en la manipulación de datos rdf se incrementará considerablemente, lo que conducirá inevitablemente a que los cms lleguen a utilizarse como *frameworks* para el desarrollo de interfaces entre datos de la web semántica y contenidos de los sitios web que gestionen⁶. Mientras tanto identificamos que algunos cms, en especial *Drupal* y *phpBB*, hacen

uso de rdfa a un nivel muy básico siguiendo un modelo de representación de sus contenidos con rdf⁷.

La hoja de ruta de la incorporación de los cms en la web semántica pasa ineludiblemente por la instalación y configuración de todo el software (entornos de ejecución, *middlewares*, cms, módulos...) de un modo sencillo, para acelerar la productividad de los sistemas y comenzar rápidamente a publicar ciertos tipos de contenido mediante *Sparql endpoints*, generar urls de fuentes de datos rdf para su acceso por parte de otras aplicaciones o integrar fuentes de datos de, por ejemplo, *Europeana* o *Dbpedia*.

En definitiva, se precisan soluciones del tipo “instalar y listo” esenciales para simplificar la complejidad que supone el uso de software heterogéneo sin tener que caminar (y posiblemente perderse) en el laberinto “modular” de los cms. Todo lo anterior con un objetivo claro: evitar los problemas que en la era de la web semántica acarrearía el hecho de que datos y contenidos sigan dos caminos divergentes en su gestión y uso.

El futuro demostrará en qué grado y en qué forma se cumplirán todas estas expectativas y necesidades.

5. Notas

1. Xml, rdf, owl y sparql con especificaciones del W3C. xml (*extensible markup language*) define una sintaxis para el intercambio de datos entre aplicaciones. Rdf es el modelo de descripción de recursos fundamental de la web semántica, basado en tripletas del tipo sujeto-objeto-predicado que conforman grafos complejos. Owl se usa junto con rdf y permite definir ontologías que describen aspectos lógicos de las relaciones entre recursos. Skos es una ontología owl para la representación de sistemas de organización del conocimiento muy utilizado para la elaboración de tesauros y clasificaciones. Sparql es un lenguaje de consulta para interrogar y recuperar datos de los grafos rdf.

2. Existen infinidad de catálogos sobre *open linked data*. A este respecto existe un mapa con todas las iniciativas de este tipo de la *Fundación Ctic*. También puede consultarse el registro de la *Open Knowledge Foundation* con *datasets* de *linked open data*.
<http://datos.fundacionctic.org/sandbox/catalog/faceted>
<http://ckan.net>

3. La expresión “mapeado” proviene del término inglés “mapping” que se asocia a la definición de correspondencias entre elementos de diferentes conjuntos. En este caso se trataría de definir equivalencias o similitudes entre elementos de diferentes especificaciones de descripción.

4. Un *triplestore* es una base de datos que almacena *triplets* (declaraciones rdf). Suele utilizarse mediante una interfaz de usuario, un *Sparql endpoint* o mediante una api que permite el diseño de aplicaciones que utilicen el *triplestore* para almacenar datos.

5. Al referirme a html también lo estoy haciendo a xhtml.

6. Es muy sugerente el ejemplo de *Islandora* que integra *Drupal* y *Fedora*.
<https://wiki.duraspace.org/display/ISLANDORA/Islandora>

7. A modo de ejemplo puede consultarse:
<http://blog.semantic-web.at/semantic-web-and-drupal>

6. Referencias bibliográficas

Alani, Harith; Dupplaw, David; Sheridan, John; O'Hara, Kieron; Darlington, John; Shadbolt, Nigel; Tullo, Carol. "Unlocking the potential of public sector information with semantic web technology". *ISWC/ASWC*, 2007, pp. 708-721.
<http://eprints.ecs.soton.ac.uk/14429>

Allemang, Dean; Hendler, James. *Semantic web for the working ontologist: effective modelling in rdfs and owl* (2ª ed.). Morgan Kaufmann, 2011. ISBN: 978 01 237 3556 0

Berners-Lee, Tim. *Linked data*, 2006.
<http://www.w3.org/DesignIssues/LinkedData.html>

Bishop, Tony A.; Karne, Ramesh K. "A survey of middleware". En: *18th intl conf on computers and their applications*, March 26-28, 2003.
<http://triton.towson.edu/~karne/research/middlew/survey.pdf>

Bizer, Christian. "The emerging web of linked data". *Journal IEEE intelligent systems archive*, 2009, v. 24, n. 5, pp. 87-92.
<http://pis.csd.auth.gr/mtpx/sw/material/IEEE-IS/IS-24-5.pdf>

Ding, Li; Michaelis, James; McGuinness, Deborah L.; Hendler, Jim. "Making sense of open government data". En: *Procs of the WebSci10: Extending the frontiers of society on-line*, April 26-27, 2010.

http://journal.webscience.org/394/2/websci10_submission_112.pdf

Fitzgerald, Brian. "The transformation of open source software". *MIS quarterly*, 2006, v. 30, n. 3, pp. 587-598.
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.159.5834&rep=rep1&type=pdf>

Heath, Tom; Bizer, Christian. "Linked data: evolving the Web into a global data space". *Synthesis lectures on the semantic web: theory and technology*, 2001, v. 1, n. 1, pp. 1-136.
<http://linkeddatabook.com/book>

Hedstrom, Margaret; King, John L. *On the LAM: library, archive, and museum collections in the creation and maintenance of knowledge communities*. Paris, Francia: Organización para la Cooperación y el Desarrollo Económico (OCDE), 2003.
<http://www.oecd.org/dataoecd/59/63/32126054.pdf>

Hiramatsu, Kaoru; Reitsma, Femke. "GeoReferencing the semantic web: ontology based markup of geographically referenced information". En: *Joint EuroSDR/EuroGeographics workshop on ontologies and schema translation services*, Paris, Abril 2004.
http://www.mindswap.org/2004/geo/geoStuff_files/HiramatsuReitsma04GeoRef.PDF

Pastor-Sánchez, Juan-Antonio. *Tecnologías de la web semántica*. Barcelona: Editorial UOC, 2011, Colección El profesional de la información, n. 1, 120 pp. ISBN: 978 84 9788 474 7.

Pemberton, Steven. *Rdfa for html authors*. W3C/CWI, 2009.
<http://www.w3.org/MarkUp/2009/rdfa-for-html-authors>

Velegrakis, Yannis. "Relational technologies, meta-data and rdf". En: Virgilio, Roberto de; Giunchiglia, Fausto; Tanca, Letizia (eds.). *Semantic web information management: a model-based perspective*. Heidelberg; New York: Springer, 2010, pp. 45-70. ISBN: 978 3 642 04328 4
<http://disi.unitn.it/~velgias/docs/Velegrakis10.pdf>