

**Universidad
Autónoma de
Madrid**



**Escuela Politécnica Superior
EPS**

**Máster Universitario en
Investigación e Innovación en TIC**

Trabajo de Fin de Máster

**Aplicación de Tecnologías Open Data a los
datos académicos de los docentes e
investigadores de la EPS - UAM**

Autor: Freddy Mauricio Tapia León

Septiembre 2013

1.	Introducción	5
1.1.	Motivación.	6
1.1.1.	Gobierno Electrónico (eGovernment).....	6
1.1.2.	Gobierno Abierto (Open Government).....	7
1.1.3.	Open Data	8
1.2.	Propuesta	10
2.	Estado del arte	11
2.1.	Web Semántica.	11
2.2.	Arquitectura de la Web Semántica.	12
2.3.	Tecnologías de la Web Semántica.....	14
2.3.1.	XML.....	14
2.3.2.	RDF/RDFS.....	14
2.3.3.	OWL.....	17
2.3.3.1.	Sub Lenguajes de OWL.....	17
2.3.4.	SPARQL (SPARQL Protocol and RDF Query Language).....	18
2.3.5.	Linked Data.....	18
2.4.	URL, URN y URI.....	19
2.5.	Ontologías	20
2.6.	DBpedia	21
2.6.1.	Ontología DBpedia	23
2.6.2.	Mapeos en DBpedia	25
3.	Aplicación de las Tecnologías Semánticas para la generación de datos abiertos relacionados con los docentes de la EPS – UAM.....	26
3.1	Introducción.	26
3.2	Generación de tripletas a partir de una Base de Datos Relacional.....	27
3.3	Descripción del Método utilizado.	29
4.	Conclusiones y Trabajos Futuros.....	35
4.1.	Conclusiones.....	35
4.2.	Trabajos Futuros.....	36
5.	Anexos.....	38
5.1	Ejemplos Significativos.....	38
5.2	Recomendaciones y Buenas Prácticas sobre D2RQ.....	46

5.3	Fichero TTL	47
6.	Referencias.....	59

Ilustración 1: Desarrollo de eGov - Sud América.....	7
Ilustración 2: Naciones que han incorporado la Iniciativa Open Data.....	8
Ilustración 3: Web Tradicional Vs Web Semántica	11
Ilustración 4: Estructura de la Web Semántica	13
Ilustración 5: RDF Schema.....	16
Ilustración 6: Arquitectura de BB.DD.	18
Ilustración 7: Interconexiones en Linked Data.....	19
Ilustración 8: Estructura de una URI.....	20
Ilustración 9: Linked Data.....	22
Ilustración 10: Clases de Ontología.....	23
Ilustración 11: Propiedades contenidas dentro de la clase Activity	24
Ilustración 12: Propiedades de la Clase Game.....	24
Ilustración 13: Infobox de Ecuador	25
Ilustración 14: Propiedades Clase Country	26
Ilustración 15: Estructura D2RQ.....	27
Ilustración 16: Vinculación entre tablas.....	28
Ilustración 17: Obtención de tripletas - EPS.....	29
Ilustración 18: Conexión a la B.D. Relacional	30
Ilustración 19: Mapeo de la Tabla Empleado.....	31
Ilustración 20: Uso del JOIN - Vinculación entre tablas.....	32
Ilustración 21: Tripletas RDF – empleado.....	32
Ilustración 22: Tripletas RDF - Congreso.....	33
Ilustración 23: Pantalla Inicio D2R.....	33
Ilustración 24: Pantalla de Ingreso y Obtención de datos	34

1. Introducción

De acuerdo con IBM, cada día se generan alrededor de 2.5 quintillones (10^{30}) de bytes de datos, donde el 90% de todos los datos han sido creados en los últimos tres años (1). Estos datos provienen de sensores asociados al clima, de la interacción dentro de las redes sociales, de fotos y videos digitales, de transacciones electrónicas, así como, de múltiples servicios los cuales son obtenidos mediante los actuales dispositivos móviles. A la gestión eficiente de estos datos se le conoce como Big Data (1).

El término Big Data es más que una simple cuestión de tamaño, es una nueva oportunidad para proponer ideas que permitan analizar los datos y sus contenidos. Una gestión eficiente permitiría una toma de decisiones más ágil. Por este motivo, algunos gobiernos están abriendo sus plataformas de información con el objetivo de publicar informes, estadísticas, estudios o análisis, creados y gestionados por la administración pública en todos los ámbitos (salud, economía, educación, etc.); es decir, datos de interés colectivo (2).

En este contexto se han desarrollado algunas propuestas que permiten involucrarse en el proyecto Open Data, las cuales promueven el acceso y reutilización de los datos. Por lo tanto, el presente trabajo tiene por finalidad poner en abierto los datos académicos de los docentes e investigadores de la EPS – UAM, para lo cual se han considerado los siguientes capítulos a desarrollar:

En el Capítulo I (Introducción): Se presenta la motivación, qué es el Gobierno Electrónico, el Gobierno Abierto, Open Data y la propuesta a desarrollar.

En el Capítulo II (Estado del Arte): Se hace un recorrido sobre las distintas teorías, herramientas y tecnologías relacionadas con la propuesta planteada en el capítulo anterior.

En el Capítulo III (Aplicación de las Tecnologías Semánticas para la generación de datos abiertos relacionados con los docentes de la EPS – UAM): Se muestra todo el proceso de experimentación desarrollado, basado en el método propuesto. Así como también, se muestra un esquema sobre la vinculación que mantienen algunas tablas, y por último se realiza un ejemplo de consulta sobre SPARQL.

El Capítulo IV (Conclusiones y Trabajos Futuros): Recoge ideas principales extraídas del presente trabajo, así como también, se presentan algunas propuestas sobre posibles campos de acción relacionados al presente proyecto, además, de posibles futuras líneas de investigación.

El Capítulo V (Anexos): Como complemento a los anteriores capítulos, el presente trabajo incluye tres anexos que aportan información complementaria relacionada con la propuesta:

El Anexo 5.1 (Ejemplos significativos): Este anexo reúne los resultados obtenidos con los datos provistos por la EPS – UAM, es decir, se cumple el objetivo planteado en la propuesta.

El Anexo 5.2 (Recomendaciones y buenas prácticas sobre D2RQ): Esta guía tiene por finalidad el generar un manual básico y didáctico relacionado al proceso de instalación de la herramienta D2RQ.

El Anexo 5.3 (Fichero TTL): Este anexo incluye todos los mapeos realizados con los datos de la B.D. relacional, los cuales servirán para realizar el volcado de los datos a formato RDF.

El Capítulo VI (Referencias): Incluye todas las referencias bibliográficas citadas dentro del presente proyecto.

1.1.Motivación.

El World Wide Web Consortium (W3C) como vinculación al proyecto Open Data (3), propone mejorar la Web tradicional (publicación de documentos), por medio de datos compartidos, a la cual se la conoce como Web Semántica.

Por tanto, se abre un nuevo enfoque con el desarrollo de nuevas herramientas tecnológicas que permiten utilizar, de mejor forma, estos datos. Es por esto que muchos gobiernos, cada vez brindan a más ciudadanos la oportunidad de tener un mayor protagonismo en la incursión de aspectos de carácter gubernamental.

Con estos antecedentes, surge la motivación para involucrar a la Escuela Politécnica Superior – UAM dentro de la iniciativa Open Data, para lo cual se cuenta con los datos académicos de los docentes e investigadores de esta Escuela, y la herramienta D2RQ como medio para la generación de datos abiertos (RDF).

1.1.1. Gobierno Electrónico (eGovernment).

El gobierno electrónico no solo representa una mejora técnica en cuanto a dotación de equipos informáticos se refiere sino que, más bien, representa una transformación de la cultura de gobierno, promoviendo nuevos servicios electrónicos, así como, mejorar los ya existentes. Por tanto, el uso adecuado de las tecnologías de la información y las comunicaciones (TIC) debe fortalecer y modernizar el funcionamiento de los sistemas políticos con el fin de lograr administraciones más eficientes.

Muchos gobiernos han decidido reformar sus leyes para permitir la creación de Organizaciones Técnicas Especializadas adscritas al Gobierno Central, donde las principales tareas a cumplir son: proponer y liderar proyectos tecnológicos, reducir la brecha digital y convertir la sociedad de la información y el conocimiento en una oportunidad para todos los ciudadanos. Con mayor énfasis en los países de América Latina, en donde los aspectos económicos y socio culturales impiden una compenetración más rápida y profunda de este tipo de tareas. Por ejemplo: En el Ecuador el ordenamiento jurídico vigente, el cual regula al sector de las telecomunicaciones dentro del sistema económico garantizado en la Constitución Política de la República del Ecuador, dispone en su parte central: “Será

responsabilidad del Estado la provisión de servicios públicos de agua potable y de riego, saneamiento, fuerza eléctrica, telecomunicaciones, vialidad, facilidades portuarias y otros de naturaleza similar” (4).

En este contexto, la ilustración 1 muestra el desarrollo que los gobiernos de América Latina han tenido en aspectos relacionados a Gobierno Electrónico según estudio realizado por las Naciones Unidas en el año 2012.

Este estudio refleja el avance que han presentado algunos países, así como, el nivel de retroceso que tienen otros. El estudio se enfocó específicamente a dos años (2010 y 2012), en los cuales podemos observar que Chile lidera la tabla con un 67.69% y ocupa el puesto número 39 a nivel mundial, en cambio Ecuador registra un 48.69%, ocupando el puesto 102 a nivel mundial. Estos porcentajes hacen referencia al año 2012. (5). Corea del Sur obtuvo el primer lugar, seguido por Holanda y Reino Unido, a nivel mundial.

Country	E-gov. development index		World e-gov. development ranking	
	2012	2010	2012	2010
Chile	0.6769	0.6014	39	34
Colombia	0.6572	0.6125	43	31
Uruguay	0.6315	0.5848	50	36
Argentina	0.6228	0.5467	56	48
Brazil	0.6167	0.5006	59	61
Venezuela	0.5585	0.4774	71	70
Peru	0.5230	0.4923	82	63
Ecuador	0.4869	0.4322	102	95
Paraguay	0.4802	0.4243	104	101
Bolivia (Plurinational State of)	0.4658	0.4280	106	98
Guyana	0.4549	0.4140	109	106
Suriname	0.4344	0.3283	116	127

Ilustración 1: Desarrollo de eGov - Sud América

Fuente: <http://unpan1.un.org/intradoc/groups/public/documents/un/unpan048065.pdf>

Por tanto, queda mucho por avanzar en lo relacionado a eGovernment. En la siguiente sección nos referiremos al Gobierno Abierto, el cual va un poco más allá que el Gobierno Electrónico.

1.1.2. Gobierno Abierto (Open Government).

El Gobierno Abierto promueve la innovación de las Administraciones Públicas, donde se fomenta la transparencia y apertura de ciertos procesos gubernamentales, así como también promueve la participación y control por parte de los ciudadanos. En definitiva, es ir más allá del Gobierno Electrónico.

La aportación del Gobierno Abierto se sustenta en el valor que tienen los datos generados por el Estado, los cuales podrían ser consultados, utilizados y evaluados por cualquier ciudadano.

Como se puede evidenciar, este nuevo modelo de gobierno promueve una participación ciudadana más activa, a través de esta participación se tiene un acceso libre y gratuito a los datos. Estos aspectos son esenciales para la iniciativa Open Data (6) descrita a continuación.

1.1.3. Open Data

La exposición de los datos que generan las Administraciones Públicas es parte de la iniciativa Open Data (Datos Abiertos). Esta iniciativa promueve el acceso y reutilización de los datos por parte de los ciudadanos y las empresas, fomentando así su posterior tratamiento y redistribución (7).



Ilustración 2: Naciones que han incorporado la Iniciativa Open Data

Fuente: <http://datos.fundacionctic.org/sandbox/catalog/faceted/>

En la ilustración 2 se observa cómo cada vez son más los gobiernos que deciden abrazar esta filosofía, que tiene como mayor exponente a la “Asociación de Gobiernos Abiertos”, fundada en septiembre de 2011, por ocho países: Brasil, Indonesia, México, Noruega, Filipinas, África del Sur, Reino Unido y Estados Unidos (8).

En la actualidad se encuentran más de 200 iniciativas en todo el mundo, con el objetivo de optimizar los recursos públicos y rendir cuentas como prácticas de una gestión transparente (9).

La publicación y reutilización de la información se puede hacer de una forma básica, es decir, mediante la publicación de documentos en formatos para humanos (como por ejemplo, doc y pdf) pero a la vez menos eficiente, ya que estos formatos no son entendibles por las máquinas (lenguaje natural). Por

consiguiente, se deben utilizar estándares que permitan publicar datos estructurados, así como también, tecnologías que permitan consultar, procesar y combinar estos datos. Todo esto con el objetivo de lograr un máximo rendimiento en procesos de interoperabilidad (10).

Tim Berners-Lee, inventor de la Web y Director de la W3C (World Wide Web Consortium), clasifica la calidad de los datos publicados en cinco niveles. Esta clasificación tiene el nombre de las “cinco estrellas” (11):

- **Una estrella:** Ofrecer los datos en cualquier formato. Como por ejemplo un fichero pdf, o una imagen.
- **Dos estrellas:** Entregar los datos de manera mínimamente estructurada, como por ejemplo un archivo de Excel con extensión xls.
- **Tres estrellas:** Entregar los datos en un formato que no sea propietario, como por ejemplo un archivo XML, RDF y CSV en vez de Excel.
- **Cuatro estrellas:** usar URL para identificar cosas y propiedades, de manera que se pueda focalizar los datos a través de internet. Para esto se requiere el usar estándares: RDF.
- **Cinco estrellas:** Interrelacionar los datos con otros datos.

Varias son las instituciones que promueven y aportan a esta iniciativa. A continuación se describe a las más relevantes:

- **Open Knowledge Foundation:** Dedicada a la promoción de datos y contenidos abiertos en todas sus formas, como por ejemplo: datos gubernamentales, investigación financiada con fondos públicos y contenidos de dominio público en el área cultural (12).
- **World Wide Web Consortium (W3C):** Desarrolla protocolos y estándares Web, que aseguren una verdadera reutilización de los datos en diferentes formatos (13).
- **OMG Standard:** Desarrolla normas de integración empresarial, también se dedica a reunir usuarios finales dentro de agencias gubernamentales, universidades e instituciones de investigación para compartir experiencias en la transición a nuevos enfoques de gestión y tecnología como Cloud Computing (14).
- **Sunlight Foundation:** Es una organización sin fines de lucro, no partidista, que utiliza el poder de Internet para catalizar una mayor apertura y transparencia del gobierno, y proporcionar nuevas herramientas y recursos para los medios de comunicación y público en general (15).

1.2.Propuesta

Este trabajo de fin de máster pretende poner en práctica la iniciativa Open Data dentro de la EPS - UAM. Se usa la herramienta D2RQ como medio para la generación de datos abiertos (RDF). De igual forma es necesario comprender la relación que mantiene esta iniciativa con algunos temas asociados (eGovernment y Open Government).

Se ha seleccionado la herramienta D2RQ debido a la robustez que ha demostrado en los últimos años, así como, también por su capacidad para trabajar con B.D. relacionales (ver apartado 3.2). También, se usarán vocabularios estandarizados, como por ejemplo: Dublin Core (DC), Friend of a Friend (FOAF) y en especial el propuesto por DBpedia (ver apartado 2.6 y 3.2).

La herramienta D2RQ ha sido desarrollada por Freie Universität Berlin (ver apartado 3.1). Esta herramienta permite la creación de tripletas RDF (ver apartado 2.3.2) a partir de Bases de Datos tradicionales (relacionales). Para ello se emplearán datos reales provistos por La Escuela Politécnica Superior (EPS) – UAM para ponerlos en formato abierto (tripletas RDF). Estos datos son referentes a la investigación que han realizado los docentes de la EPS, entre el año 2004 y 2012. A partir de esta B. D., se pondrán en abierto los datos semánticos obtenidos.

Estos datos son de calidad (5 estrellas), por las siguientes razones:

- Los datos obtenidos son RDF digitales.
- Uso de vocabularios estandarizados.
- Acceso universal a los datos mediante una URI, las mismas que deberían mantener una semántica adecuada, es decir, expresar algo lógico.
- Vinculación de los datos generados con la Ontología DBpedia.

2. Estado del arte

2.1. Web Semántica.

A finales de los años 90, Tim Berners-Lee propuso reformular a la web tradicional (web de documentos), la cual sería conocida como Web Semántica o web de datos (ver ilustración 3 - b). Esta Web se comportaría como una gran base de datos, que permitiría la interoperabilidad entre cualquier sistema informático (10) debido al significado semántico que ésta podría contener, facilitando de esta forma la ubicación e identificación de los datos. Por ejemplo: Cervantes <escribe> El Quijote. Si analizamos los fragmentos de forma separada (Cervantes y El Quijote) se concluye que dichos datos no expresan nada, al contrario, si adicionamos un tercer fragmento <escribe> la expresión empieza a tener un mayor significado semántico.

Este nuevo concepto es conocido como la Web Semántica. El W3C (World Wide Web Consortium), ha sido el organismo encargado de regular y desarrollar los estándares requeridos (17).

En la ilustración 3 podemos observar algunas particularidades (18):

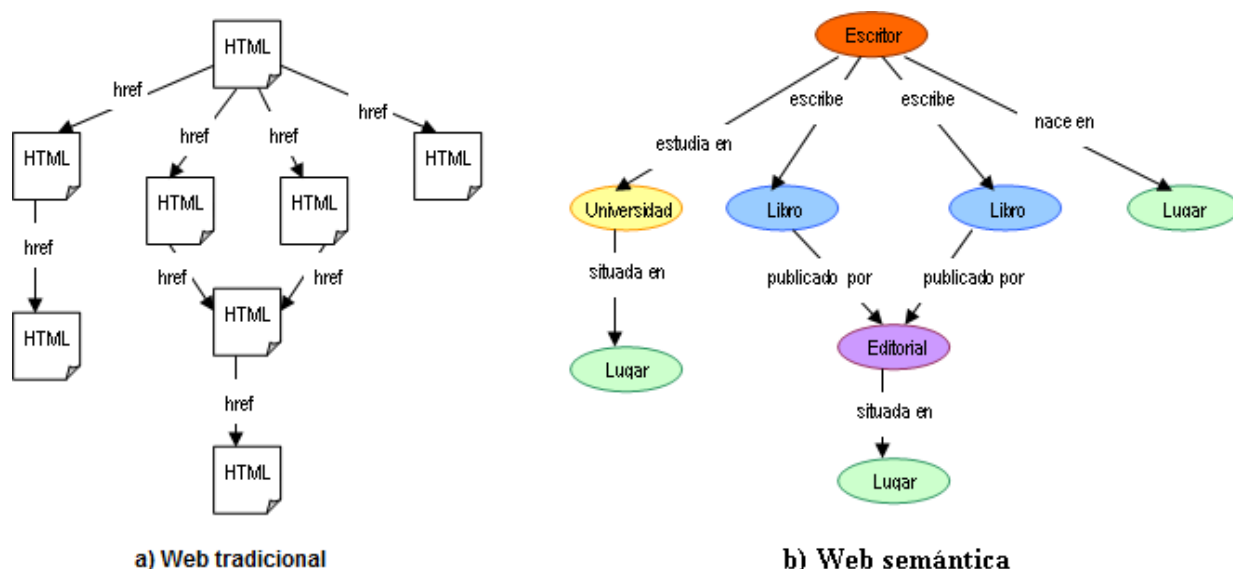


Ilustración 3: Web Tradicional Vs Web Semántica

Fuente: http://www.ekonsulta.net/ekonsulta/wiki/index.php/Web_Sem%C3%A1ntica

Web Tradicional (ilustración 3 - a):

- No distingue la relación que cada nodo puede tener dentro del grafo, lo mismo ocurre con la información contenida en cada nodo.
- Al no existir una adecuada gestión de la información, muchos de los datos están desestructurados, es decir, los recursos crecen caóticamente sin organización alguna.

Web Semántica (ilustración 3 - b):

- Provee de un mayor significado a toda la información, es decir, el grafo es perfectamente entendible por las máquinas.
- Identifica de mejor forma la información que posee cada recurso, así como también, la relación que tienen sus nodos.
- Utiliza una infraestructura común (RDF) la cual permite compartir información.

La Web Semántica utiliza ficheros en formato RDF, los cuales permiten el uso de las descripciones mencionadas anteriormente. También utiliza un lenguaje de consulta denominado SPARQL (ver apartado 2.3.4), y OWL como lenguaje para especificar ontologías (ver apartado 2.3.3). Todos estos mecanismos son parte de una infraestructura global que permite compartir, reutilizar datos entre diferentes personas (19).

2.2.Arquitectura de la Web Semántica.

En el año 2000, Tim Berners-Lee presentó una estructura de siete niveles (ver ilustración 4), basadas en la capacidad de representación semántica que podrían tener los datos (20), es decir, entendimiento e interpretación de la información por parte de las computadoras. A continuación se hace una breve descripción de la estructura propuesta.

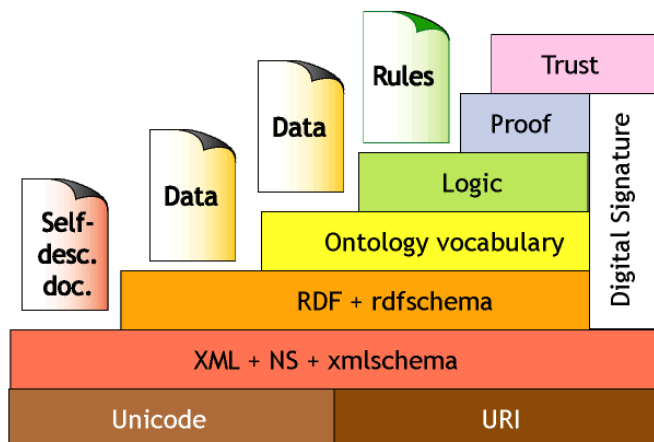


Ilustración 4: Estructura de la Web Semántica

Fuente: <http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide11-0.html>

- **Unicode:** Es un estándar que permite la codificación de texto, los cuales podrían ser utilizados para expresar información dentro de la Web Semántica.
- **URI (Identificador Uniforme de Recursos):** Permite el acceso y localización de un recurso dentro de una tripleta RDF (ver apartado 2.4).
- **XML (Extensible Markup Language) y XML Schema:** XML y XML Schema, son estándares que permiten definir tipos de documentos por medio de etiquetas (<title>, <description>, <subject>, etc.) lo que facilitaría su posterior codificación, procesamiento y explotación (ver apartado 2.3.1).
- **NS (Namespaces):** De acuerdo con el W3C, fueron introducidos en XML con el objetivo de estandarizar vocabularios, es decir, evitar conflictos entre ellos, permitiendo de esta forma el uso de múltiples vocabularios en un mismo documento (ver apartado 2.3.1).
- **RDF (Resource Description Framework) y RDF Schema:** El modelo RDF, permite expresar relaciones en formatos de una tripleta RDF (sujeto, predicado y objeto). Facilitando de esta forma su posterior uso a través de una URI (ver apartado 2.3.2).
Por otro lado, RDF Schema facilita la interpretación semántica de la tripleta RDF.
- **Ontology vocabulary:** Esta capa tiene por objetivo dotar de una mayor expresividad semántica, es decir, describir objetos y relaciones que faciliten su entendimiento y relación con otras ontologías, para esto se cuenta con algunos lenguajes que proporcionan la expresividad requerida tal es el caso de OWL (ver apartado 2.3.3).
- **Logic:** Esta capa permite el entendimiento de los recursos definidos por medio de reglas lógicas que permitan inferir en nuevos conocimientos mediante procesos de razonamiento.
- **Proof:** Comprueba que los resultados sean correctos, es decir, valida la información, desde el

punto de vista lógico. Por lo cual es necesario comprobar y validar la procedencia de la información, así como su integridad.

- **Trust y Digital Signature:** La Firma Electrónica garantiza el origen de la información, de esta forma se lograría que la información intercambiada sea confiable.

En la siguiente sección se describirá con mayor detalle las tecnologías semánticas más importantes.

2.3. Tecnologías de la Web Semántica.

Esta sección tiene por finalidad describir las herramientas y tecnologías relacionadas con la Web Semántica, para lo cual esta sección ha sido dividida en cinco subsecciones (XML, RDF/RDFS, OWL, SPARQL y LINKED DATA), todos estos términos son piezas claves y vinculantes con la Web Semántica.

2.3.1. XML

En el año 1998 aparece XML (eXtensible Markup Language), este lenguaje agrupa los datos por elementos y los delimita por etiquetas, facilitando de esta forma su posterior codificación y reutilización, es decir, dotarles de una mejor estructura a los datos (22).

Con el objetivo de evitar duplicidad y problemas en cuanto a vocabularios, el W3C propone el uso y estandarización de namespaces, como medio para combinar elementos y atributos en XML, consiguiendo de esta forma su reutilización (20).

XML es un lenguaje independiente de la plataforma que se use (software y hardware), garantizando de esta forma una reutilización e interoperabilidad básica. XML carece de especificaciones semánticas, por lo tanto, este proceso es delegado a las capas superiores como lo muestra la ilustración 4, de este modo serían otros los lenguajes encargados de definir aspectos semánticos (21).

2.3.2. RDF/RDFS

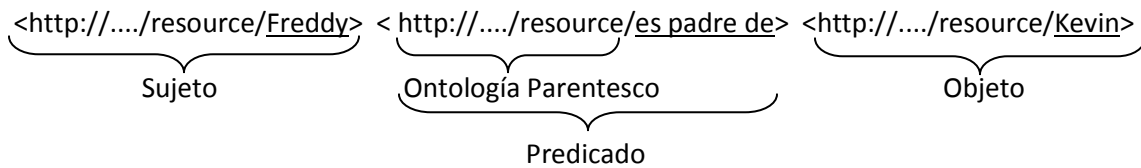
RDF (Resource Description Framework) es un modelo de datos para la representación de metadatos (descripción entre datos) en la Web Semántica, a través del cual se proporciona información descriptiva sobre los recursos que se encuentran en la Web. RDF se apoya en tres conceptos fundamentales (21):

- **Recurso (Resource):** Es la descripción de un objeto, el cual podría ser: una persona, una ciudad, un lugar, etc. Todo recurso está identificado por una URI.
- **Propiedad (Property):** Define una característica ó la relación entre recursos.

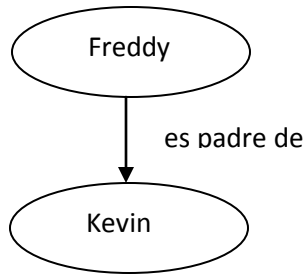
- **Sentencia (Statement):** Está formada por un sujeto, predicado y objeto. Siendo esta la estructura fundamental en la que se basa una tripleta RDF.

Adicionalmente, a estos conceptos individualmente se los conoce como: sujeto (recurso a describir), predicado (propiedad que define una característica del objeto) y el objeto (valor que toma la propiedad para el recurso que lo define).

Estas pueden ser expresadas de forma textual:



Ó, de forma gráfica, mediante un grafo dirigido:



RDF Schema es una ontología básica descrita con RDF la cual define propiedades y clases de recursos RDF (23), entre otros conceptos.

A continuación, utilizaremos un ejemplo propuesto por el W3C (ver ilustración 5), para exponer el uso de cómo RDF y RDFS describen cosas del mundo real. Centrémonos en “eg:Person” y “eg:Document”:

- “eg:Document” y “eg:Person” son clases, que pertenecen al dominio “eg:autor”, la cual a su vez se relaciona con un recurso de nivel superior (superclase) por medio de “rdfs:subClassOf”.
- “eg:Document” y “eg:Work” son subclases (clase de nivel inferior) de “rdfs:Resource”.
- Si analizamos la URI “http://.../Proposal”, podemos observar que este recurso es de tipo “eg:Document”, el cual a su vez es subclase de “eg:Work”.
- A través de la relación definida por la URI se conoce el título del documento: “Information Management: A Proposal”, y el nombre del autor “Tim Berners-Lee”, el cuál es de tipo persona, y subclase de “eg:Agent”.

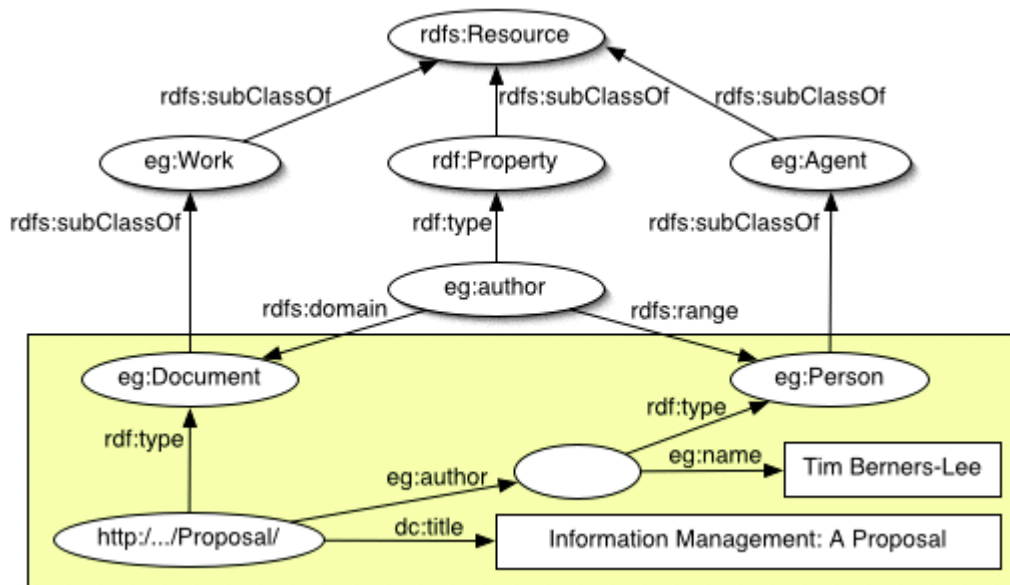


Ilustración 5: RDF Schema

Fuente: <http://www.w3.org/TR/2002/WD-rdf-schema-20021112/intro-example.jpg>

RDFS define un conjunto de recursos RDF, estos recursos pueden ser usados para describir algunas propiedades de otros recursos RDF (incluyendo sus propiedades), de esta forma se estaría dotando de semántica a todos los recursos utilizados. En la tabla 1 se muestran algunas descripciones (24):

Clases RDF:

El Recurso es una clase de la cual se heredan todos los objetos descritos en un documento RDF.

Nombre de la Clase	Comentario
rdfs:Resource	Todas las cosas descritas por RDF son llamadas "resource", y son miembros de una clase.
rdf:Property	Establece relaciones entre recursos o desde un recurso.
rdfs:Class	Corresponde a un tipo o categoría de recurso, así como también, podría ser un sub recurso de rdfs:Resource

Tabla 1: Especificaciones para definir clases RDFS

La tabla 2 muestran algunas especificaciones relacionadas al dominio y rango del vocabulario RDF, es decir, define cuál es el recurso al que se puede aplicar la propiedad y sobre qué rango de datos puede tomar valores dicha propiedad. (24):

Propiedades RDF:

Las propiedades RDF permiten relacionar:

- Clases y Propiedades.
- Clases y superclases.
- Entre propiedades.

Nombre de la Propiedad	Comentario	Dominio	Rango
rdf:type	Identifica el tipo de propiedad que puede tener un recurso.	rdfs:Resource	rdfs:Class
rdfs:subClassOf	Representa un nivel jerárquico entre clases, de aquí el término subclase	rdfs:Class	rdfs:Class
rdfs:range	Indica el rango que puede tener una propiedad	rdf:Property	rdfs:Class
rdfs:domain	Indica el dominio que puede tener una propiedad	rdf:Property	rdfs:Class

Tabla 2: Especificaciones para definir propiedades RDF

2.3.3. OWL

El Lenguaje de Ontologías Web (OWL) fue desarrollado por el Web Ontology Working Group del W3C (25). Está construido sobre RDF y RDF Schema, con la finalidad de dotarle de contenido semántico a las ontologías descritas por este lenguaje. Por otra parte:

- OWL provee de una mayor capacidad expresiva que XML, RDF y RDFS.
- OWL es una fusión del Lenguaje de Ontologías DAML y el lenguaje OIL (26).

En diciembre de 2012, el W3C recomendó una nueva versión del lenguaje de ontologías OWL, denominado OWL-2 (27).

2.3.3.1. Sub Lenguajes de OWL

OWL proporciona tres lenguajes, la principal diferencia entre cada uno de ellos es el nivel de expresividad con el que trabajan. A continuación se proporciona una breve descripción (23):

- **OWL Lite:** Está diseñado para una fácil implantación, ya que maneja niveles básicos de jerarquización y restricción en cuanto a la funcionalidad, es decir, provee a los usuarios de niveles básicos de expresividad. A cambio, se garantiza la computabilidad del modelo, esto es, cálculos en tiempos razonables.
- **OWL DL ("Description Logic"):** Habilita el razonamiento lógico en los sistemas, garantizando que todas las conclusiones sean computables. OWL DL, es el límite entre expresividad y capacidad de razonamiento, es decir, todos los cálculos se resolverán en un tiempo finito.
- **OWL Full:** Provee a los usuarios niveles máximos de expresividad, pero sin garantías computacionales, debido a que estos razonamientos pueden llevar tiempos infinitos en resolverlos.

2.3.4. SPARQL (SPARQL Protocol and RDF Query Language)

Es un lenguaje de consultas para datos RDF. En la ilustración 6 se puede observar un esquema básico de implantación, el cual muestra a SQL como un lenguaje de consulta, y la vinculación que tiene con un motor de almacenamiento y recuperación de datos. Ahora bien, si lo comparamos con SPARQL podemos darnos cuenta que el principio general es el mismo, la principal diferencia es el uso de RDF y OWL. El W3C recomienda su uso, desde el año 2008.

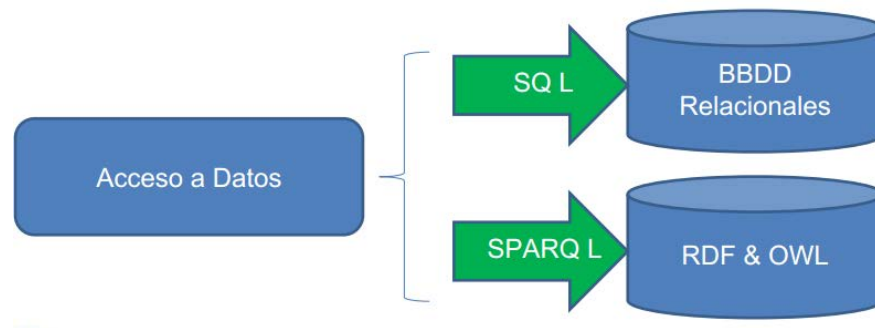


Ilustración 6: Arquitectura de BB.DD.

Fuente: <http://sinbad.dit.upm.es/docencia/doctorado/curso0809/HichamSPARQLrevCarmen.pdf>

2.3.5. Linked Data.

Son las diferentes interconexiones que tienen los nodos dentro de la Web Semántica, donde cada nodo puede ser considerado como una gran base de datos por el nivel de conectividad y distribución que tienen, esto se puede observar en la ilustración 7 (nivel de conectividad que tiene la DBpedia). De este modo las personas y las máquinas pueden reutilizar los datos de una forma eficiente. Un claro ejemplo de esto es DBpedia (29), la cual ocupa un lugar central dentro la Web Semántica (ver ilustración 9).

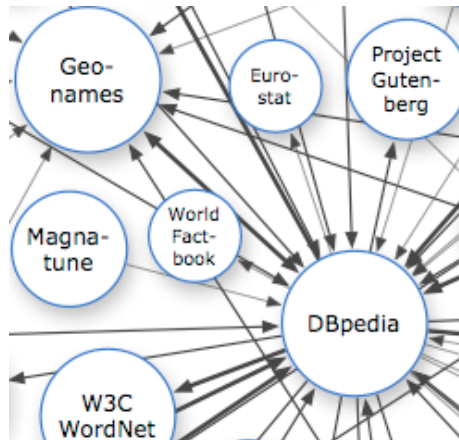


Ilustración 7: Interconexiones en Linked Data

Fuente: <http://wiki.dbpedia.org/Interlinking>

Linked Data (29) se basa en el uso de cuatro elementos básicos y necesarios:

- **Uso de URIs para identificar las cosas:** Este proceso ofrece un nivel de abstracción del lenguaje natural, es decir, fácil interpretación de los recursos dentro de una tripleta RDF.
- **Uso de URIs HTTP:** El uso de URIs sobre HTTP asegura que cualquier recurso pueda ser buscado y accedido mediante la Web. Recalcando que las URIs no son sólo direcciones, sino identificadores de recursos.
- **Información sobre los recursos usando RDF:** Una vez obtenida la información sobre un recurso mediante una URI y accedido vía HTTP, se obtendría información en formato RDF.
- **Inclusión de enlaces a otros URIs:** El principal elemento es la conexión de los datos, los cuales no pueden quedar aislados. Es necesario destacar que la utilización de enlaces o recursos provenientes de sitios especializados, ofrece un valor añadido a la información.

2.4. URL, URN y URI

URL (Uniform Resource Locator): Son localizadores que se usan para identificar recursos en Internet (páginas web) (30).

Ejemplo: <http://www.ecuador.travel/>

URN (Uniform Resource Name): Son cadenas de texto que nombran o identifican recursos en Internet, una URN no se usa como medio de localización, actualmente estas no son utilizadas (30).

Ejemplo: URN: isbn: 0451450523

Este corresponde a un identificador de libro titulado "The Last Unicorn".

URI (Uniform Resource Identifier): Son cadenas de texto que proporcionan un medio simple y extensible para identificar recursos. Por consiguiente, una URI engloba los dos conceptos anteriores (URL y URN). En la ilustración 8, se observa lo mencionado anteriormente (30).

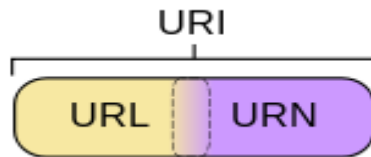


Ilustración 8: Estructura de una URI

Fuente: http://en.wikipedia.org/wiki/Uniform_resource_identifier

Una URI genérica, tiene la siguiente estructura (31):

Ejemplo: `http://www.hispamedios.com/?pagina=2#inicio`

Esquema - HTTP:

Identifica el protocolo a usar. Existe una variedad de protocolos como por ejemplo: FTP, TELNET, etc. Siendo el más usado HTTP.

Parte jerárquica - `www.hispamedios.com`:

Se refiere a la información del dominio, es decir, nombre y ruta del servidor para acceder al recurso.

Solicitud - `pagina=2`:

Son las variables que se transfieren a la página Web (recursos). Las cuales están separadas de la parte jerárquica mediante el signo de interrogación (?), y del fragmento mediante el signo numeral (#) si lo hubiese.

Fragmento - `inicio`:

Indica una subdirección dentro del recurso. La cual está delimitada por el signo numeral (#). Este fragmento es la diferencia entre una URL y una URI, por cuanto las URL no identifican fragmentos.

2.5. Ontologías

El término ontología ha sido una pieza clave para el campo de la filosofía y del conocimiento. En los últimos años este término ha obtenido un nuevo repunte debido a la incursión y transformación que ha tenido dentro de algunas áreas del conocimiento, por ejemplo: Inteligencia Artificial, Lingüística Computacional, Biblioteconomía.

Existen varias definiciones de ontología, a continuación se muestra las más relevantes (32):

- "Una ontología es un vocabulario acerca de un dominio (términos, relaciones, reglas de combinación)". Neches, 1991.
- "Una ontología es la especificación de una conceptualización común (Propuestas de Expertos)". Gruber, 1993.
- "Una ontología es una base de datos que describe los conceptos generales ó sobre un dominio, algunas de sus propiedades y cómo los conceptos se relacionan unos con otros". Weingand, 1997.

Complementando las definiciones mencionadas, podríamos decir que una ontología es una representación formal y consensuada de conceptos y de las relaciones que mantiene entre ellos. De hecho, una ontología describe los siguientes conceptos (33):

- **Clase:** Relación de dominio.
- **Relaciones:** Interacción entre las clases existentes.
- **Propiedades:** Atributos de las clases.

2.6.DBpedia

DBpedia ocupa un lugar central dentro de Linked Data debido a la relación que mantiene con múltiples fuentes de información, como se puede observar en la ilustración 9.

Por otro lado, DBpedia extrae información estructurada de la Wikipedia (34), basándose en la información contenida en los Infoboxes de Wikipedia (ver apartado 2.6.2). Algunos errores detectados en ciertos datos de DBpedia suelen deberse a errores tipográficos que Wikipedia posee dentro de los infoboxes.

De este modo, la DBpedia inglesa describe 3.77 millones de cosas, de las cuales 2.35 millones están clasificadas mediante la ontología DBpedia, esta ontología actualmente está en 111 idiomas. Todas estas versiones juntas describen 20.8 millones de "cosas", de las cuales 1.89 miles de millones son tripletas RDF, dónde, 400 millones fueron extraídas de la DBpedia inglesa y 1.46 miles de millones de todos los otros idiomas (34).

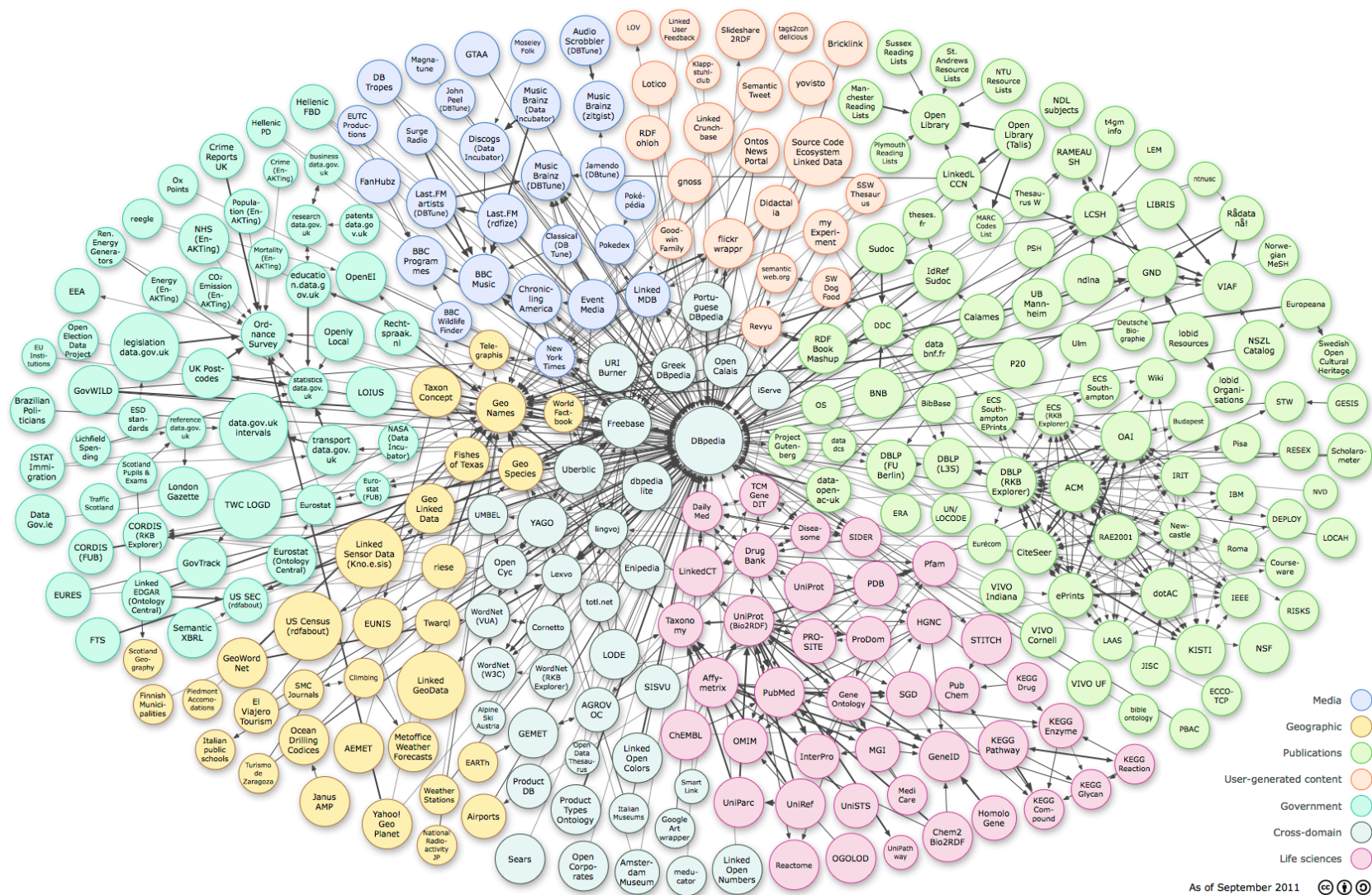


Ilustración 9: Linked Data
Fuente: <http://wiki.dbpedia.org/Interlinking>

2.6.1. Ontología DBpedia

La ontología DBpedia ha tenido algunos acontecimientos relevantes en los últimos años (34), (35), a continuación un resumen de los mismos:

- **DBpedia 3.2 (Octubre de 2008):** Se usaron nuevos métodos de extracción como reglas de granularidad fina, que es aplicada a los infoboxes para la generación de mapeos. Para esta versión la ontología DBpedia contaba con 170 clases y 940 propiedades.
- **DBpedia 3.5 (marzo de 2010):** Se introdujeron wikis públicos los cuales permitían realizar mapeos colaborativos. Para esta versión, la ontología DBpedia contaba con 259 clases y 1200 propiedades.
- **DBpedia 3.7 (Julio de 2011):** Se depura la ontología DBpedia con la finalidad de eliminar inconsistencias. Esta versión manejó 319 clases y 1541 propiedades.
- **DBpedia 3.8: (Junio del 2012):** Permite manejar un mayor número de infoboxes y mapeos, así como también involucra a más idiomas, esta versión maneja 359 clases y 1775 propiedades. Siendo esta la última versión que se encuentra operativa.

En la ilustración 10 se observan algunas clases contenidas dentro de DBpedia, las cuales se subdividen en varias subclases, como por ejemplo: “Activity” es una superclase de “Game” y “Sport”. Todas estas clases están contenidas dentro de “Owl Thing” la cual es la clase base de toda la ontología DBpedia.

Ontology Classes

- [owl:Thing](#)
 - [Activity](#) ([edit](#))
 - [Game](#) ([edit](#))
 - [BoardGame](#) ([edit](#))
 - [Sport](#) ([edit](#))
 - [Agent](#) ([edit](#))
 - [Organisation](#) ([edit](#))
 - [Band](#) ([edit](#))
 - [Broadcaster](#) ([edit](#))
 - [BroadcastNetwork](#) ([edit](#))
 - [RadioStation](#) ([edit](#))
 - [TelevisionStation](#) ([edit](#))

Ilustración 10: Clases de Ontología

Fuente: <http://mappings.dbpedia.org/server/ontology/classes/#owl:Thing>

Ahora bien, si analizamos la clase “Activity”, podemos encontrar todas las propiedades que podrían ser identificadas como miembros de esta clase. Por ejemplo: abbreviation, abstract, access, accessDate, entre otros, como muestra la ilustración 11:

Properties on *Activity*:

Name	Label	Domain	Range	Comment
abbreviation (edit)	abbreviation	owl:Thing	xsd:string	
abstract (edit)	has abstract	owl:Thing	xsd:string	Reserved for DBpedia.
access (edit)	access	owl:Thing	xsd:string	
accessDate (edit)	access date	owl:Thing	xsd:gYear	
activeYearsEndDate (edit)	active years end date	owl:Thing	xsd:date	
activeYearsEndYear (edit)	active years end year	owl:Thing	xsd:gYear	
activeYearsStartDate (edit)	active years start date	owl:Thing	xsd:date	
activeYearsStartYear (edit)	active years start year	owl:Thing	xsd:gYear	
album (edit)	album	owl:Thing	Album	

Ilustración 11: Propiedades contenidas dentro de la clase Activity

Fuente: <http://mappings.dbpedia.org/server/ontology/classes/Activity>

De la misma forma, si analizamos a la clase “Game” podemos observar algunas propiedades que describen a dicha clase, la única diferencia es el dominio al que pertenece. Como muestra la ilustración 12:

Properties on *Game*:

Name	Label	Domain	Range
equipment (edit)	equipment	Activity	owl:Thing
numberOfClubs (edit)	number of clubs	Activity	xsd:nonNegativeInteger
numberOfPeopleLicensed (edit)	number of licensed	Activity	xsd:nonNegativeInteger
numberOfPlayers (edit)	number of players	Activity	xsd:nonNegativeInteger
numberOfProfessionals (edit)	number of professionals	Activity	xsd:nonNegativeInteger

Ilustración 12: Propiedades de la clase Game

Fuente: <http://mappings.dbpedia.org/server/ontology/classes/Game>

El principal beneficio de usar la ontología DBpedia es reducir la redundancia en el uso de propiedades, (36), así como también, dotarles de semántica a los datos.

2.6.2. Mapeos en DBpedia



La ontología DBpedia ha sido creada manualmente y gracias a la colaboración de expertos en el tema. La mayor parte de los datos obtenidos por la DBpedia provienen de los infoboxes que utiliza Wikipedia. Estos infoboxes contienen información relevante de los artículos publicados en la Wikipedia, los cuales posteriormente podrían ser mapeados e incorporados a la ontología DBpedia (37). Este proceso ha permitido contar con datos mucho más claros y mejor estructurados que en los infoboxes originales (Wikipedia).

Por ejemplo: para mapear la información contenida dentro del Infobox que se relaciona al Ecuador como país (ver ilustración 13), se debe buscar una clase dentro de la ontología DBpedia la cuál contenga las propiedades descritas en el Infobox de Ecuador, para el ejemplo propuesto esta clase se llama "Country" (ver ilustración 14). Por consiguiente toda la información contenida dentro del Infobox de Ecuador podría ser mapeado usando esta clase para lo cual se debería identificar qué atributos podrían ser definidos como propiedades dentro del mapeo. Una vez obtenido el fichero respectivo, se procede al volcado de datos RDF (37).

Ilustración 13: Infobox de Ecuador

Fuente: <http://es.wikipedia.org/wiki/Ecuador>

A continuación, en la ilustración 14 se puede observar algunas de las propiedades usadas por la clase "Country". Como por ejemplo: agglomeration, areaCode, areaUrban, entre otros.

Properties on *Country*:

Name	Label	Domain	Range
agglomeration (edit)	agglomeration	PopulatedPlace	Agglomeration
agglomerationArea (edit)	agglomeration area	PopulatedPlace	Area
agglomerationDemographics (edit)	agglomeration demographics	PopulatedPlace	Demographics
alemmanicName (edit)	city allemmanic name	PopulatedPlace	<i>xsd:string</i>
anthem (edit)	anthem	PopulatedPlace	Work
apskritis (edit)	apskritis	PopulatedPlace	<i>xsd:string</i>
arberishtName (edit)	city arberisht name	PopulatedPlace	<i>xsd:string</i>
areaCode (edit)	area code	PopulatedPlace	<i>xsd:string</i>
areaRural (edit)	area rural	PopulatedPlace	Area
areaTotalRanking (edit)	total area ranking	PopulatedPlace	<i>xsd:positiveInteger</i>
areaUrban (edit)	area urban	PopulatedPlace	Area
arrondissement (edit)	arrondissement	PopulatedPlace	PopulatedPlace
barangays (edit)	barangays	PopulatedPlace	<i>xsd:string</i>
borough (edit)	borough	PopulatedPlace	PopulatedPlace
calabrianName (edit)	city calabrian name	PopulatedPlace	<i>xsd:string</i>
capital (edit)	capital	PopulatedPlace	City

Ilustración 14: Propiedades de la clase *Country*

3. Aplicación de las Tecnologías Semánticas para la generación de datos abiertos relacionados con los docentes de la EPS – UAM.

3.1 Introducción.

Freie Universität Berlin (38), junto a otras organizaciones (DERI, UCB, JP Morgan Chase, AGFA Healthcare, HP Labs, Johannes Kepler Universität Linz), han desarrollado D2RQ como herramienta para la creación de tripletas RDF a partir de Bases de Datos tradicionales, y su posterior consulta por medio de lenguaje SPARQL.

En 2003 fue lanzada la primera plataforma D2R, esta incorporaba instrucciones SQL dentro de sus reglas de mapeo.

En 2004 se lanza la versión D2RQ Server 0.1, esta versión incorporaba el uso de términos RDF y OWL (39) para la etapa de mapeo.

Actualmente se dispone de la versión 0.8, lanzada en diciembre de 2012. Entre lo más destacable de esta versión es el uso de esquemas RDFS/OWL, así como también la compatibilidad con otras BB.DD., como por ejemplo: Oracle, MySql, entre otras.

Como se puede observar en la ilustración 15, la plataforma D2RQ interactúa con los siguientes componentes (40):

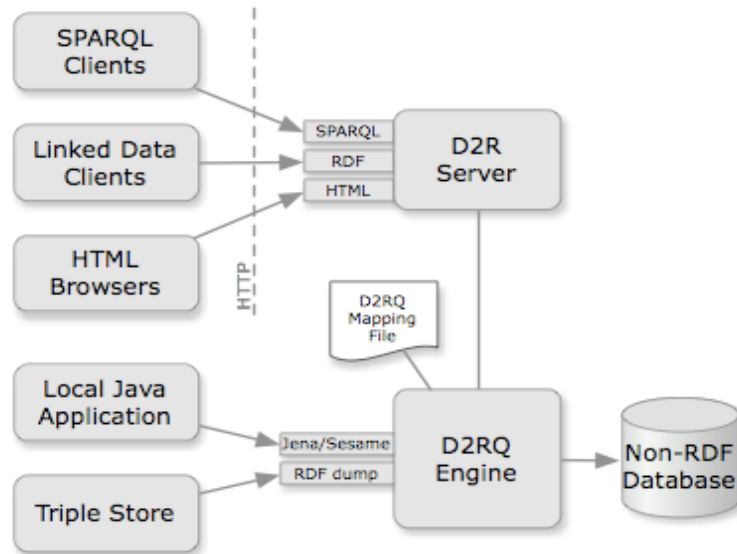


Ilustración 15: Estructura D2RQ

Fuente: <http://d2rq.org/>

- **D2RQ Mapping:** Lenguaje declarativo que describe la relación entre una ontología y un modelo de datos relacional.
- **D2RQ Engine:** Permite el acceso a la información de B.D. no RDF, usando el API de Apache Jena (sistema diseñado para trabajar con aplicaciones de la Web Semántica).
- **D2R Server:** Herramienta que permite el acceso a los datos mapeados por medio de un navegador, así como también, permite realizar consultas por medio de lenguaje SPARQL.
- **Linked Data:** Permite interrelacionar archivos RDF.
- **SPARQL:** Permite realizar consultas a B.D. por medio de lenguaje SPARQL.
- **HTML:** Proporciona acceso vía web a los datos.

3.2 Generación de tripletas a partir de una Base de Datos Relacional.

Para cumplir con los objetivos planteados se realizó la experimentación sobre la plataforma D2RQ, así como también, se recurrió a datos “no sensibles” de La Escuela Politécnica Superior (EPS - UAM), los mismos que están compuestos por cincuenta y un tablas, y algunos miles de datos entre registros y campos. La información suministrada proviene desde el año 2004 hasta el año 2012. Ha sido necesario modelar las relaciones entre los datos (mapeo) y su posterior transformación y obtención de tripletas RDF. De todos los datos disponibles lo publicado se refiere a la investigación que han realizado los profesionales de la EPS – UAM. Estos datos sirven para crear la memoria de investigación del

Departamento. Para esto, se han mapeado veinte y seis tablas, el resto de tablas no han podido ser incluidas por no contar con una estructura adecuada (tablas no relacionadas) que faciliten su reutilización.

La ilustración 16, muestra cómo se relacionan las seis tablas principales, por medio de sus claves primarias:

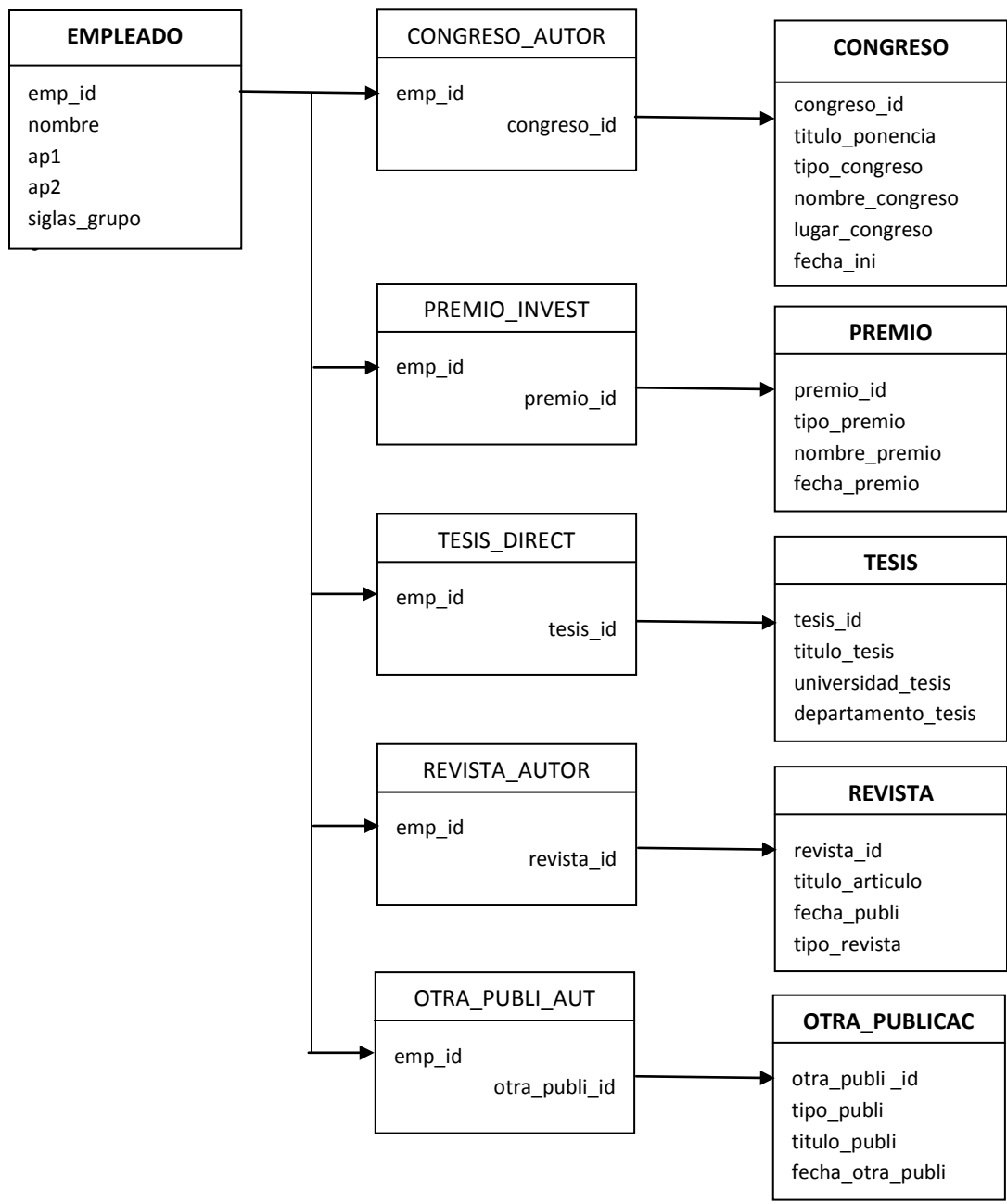


Ilustración 16: Vinculación entre tablas

La ilustración 16 muestra la estructura general que tiene cada una de las tablas, así como también, si analizamos las tablas Empleado y Congreso podemos observar que se necesita de una tercera tabla (Congreso_Autor) la cual permitiría relacionar a las dos anteriores (Empleado y Congreso). Por lo tanto, estas tres tablas (Empleado, Congreso y Congreso_Autor) serían las que debemos mapear posteriormente, este proceso se aplica para todo el resto de tablas a usar.

3.3 Descripción del Método utilizado.

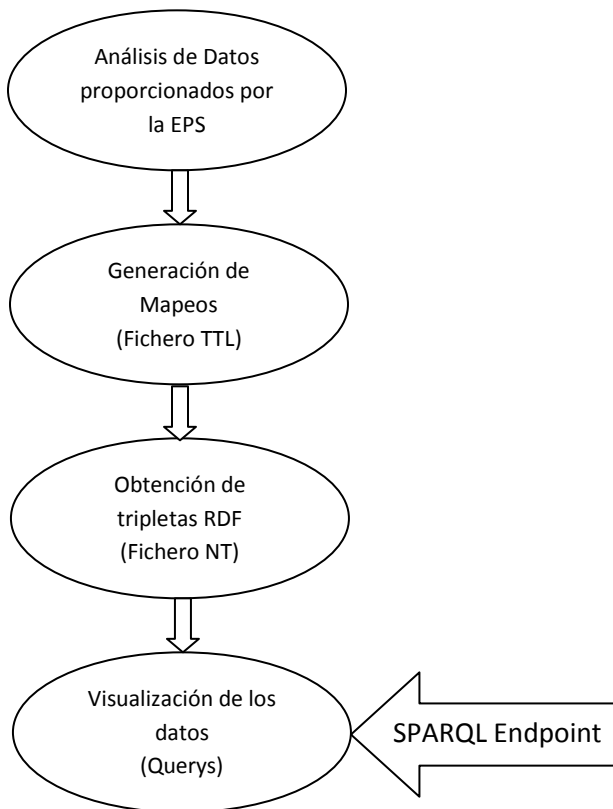


Ilustración 17: Obtención de tripletas - EPS

La ilustración 17, muestra el método utilizado para la obtención de datos RDF, el cual se compone de cuatro etapas. A continuación se proporciona una breve descripción de cada una de ellas:

1. Análisis de los Datos proporcionados por la EPS:

Como se ha mencionado anteriormente los datos suministrados por la EPS han sido cargados en su totalidad en un gestor de Base de Datos MySQL 5.5. Se debe tener cuidado con la integridad de los datos, por lo que se hicieron algunas pruebas con las tablas originales por medio de MySQL

Workbench 5.2, software que permitió revisar la estructura y contenidos de los nuevos y antiguos datos. (Ver ilustración 16).

2. Generación de mapeos (Fichero TTL):

Esta etapa es la más delicada, ya que aquí es donde se identifican los posibles datos a mapear (clases y propiedades), procedimiento que fue realizado manualmente.

Además, fue necesario identificar los vocabularios a utilizar, los mismos que representarían de mejor forma los datos a publicar. Igualmente, se han utilizado clases y propiedades propuestas por la ontología DBpedia y algunos otros estándares como por ejemplo: Dublin Core (DC), Friend of a Friend (FOAF), entre otros.

A continuación, la descripción de algunos aspectos requeridos para los mapeos:

- **Conexión a la Base de Datos:** En esta etapa se define algunos parámetros de conexión, como por ejemplo: nombre y clave de la B.D. Este mapeo permite la conexión a la B.D. (Ver ilustración 18).

```
map:database a d2rq:Database;
  d2rq:jdbcDriver "com.mysql.jdbc.Driver";
  d2rq:jdbcDSN "jdbc:mysql://127.0.0.1/eps?autoReconnect=true";
  d2rq:username "root";
  d2rq:password "*****";
  jdbc:autoReconnect "true";
  jdbc:zeroDateTimeBehavior "convertToNull";
  jdbc:keepAlive "3600";          # sends noop-query every 3600 seconds
.
```

Ilustración 18: Conexión a la B.D. Relacional

- **Mapeos individuales:** En esta etapa se identifican las clases y propiedades a mapear. En la ilustración 19 se muestra un ejemplo del mapeo realizado (map:empleado), los cuales se relacionan con la tabla empleado (Table empleado). A continuación se presenta una descripción de los mapeos realizados:

```

# Table empleado
map:empleado a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "empleado/@@empleado.emp_id@@";
  d2rq:class dbpedia-owl:Scientist;
.
map:empleado_Nombre a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property foaf:name;
  d2rq:property rdfs:label;
  d2rq:pattern "##empleado.ap1@@_##empleado.ap2@@_##empleado.nombre##";
.
map:empleado_lastmod a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dc:date;
  d2rq:column "empleado.lastmod";
.

```

Ilustración 19: Mapeo de la Tabla Empleado

- **“map:empleado”**, se refiere al asignar un nombre al mapeo, este nombre serviría en adelante para ir relacionándolo con el resto de relaciones.
 - **“# Table empleado”**, permite la incorporación de comentarios.
 - **“empleado/@@empleado.emp_id@@”**, representa un patrón de URI definido, el cual, permite enlazar la B.D. relacional con el mapeo a desarrollar.
 - **“Scientist”**, es la clase DBpedia a utilizar, esta clase relacionaría la tabla empleado con la ontología DBpedia.
 - **“map:empleado_Nombre”**, es asignar un nombre a otro mapeo, Por lo que, no se pueden repetir los nombres ya que al invocarlos existiría un error de llamada. Adicionalmente, no se debe olvidar que este otro mapeo pertenece al primer mapeo definido, en este caso, map:empleado.
 - **“foaf:name”**, es un vocabulario para describir Personas y Organizaciones.
 - **“empleado.lastmod”**, es la descripción de la columna donde está almacenado el dato date.
- **Vinculación entre tablas (uso de claves primarias):** Esta etapa tiene por objetivo realizar la vinculación entre las tres tablas identificadas anteriormente (ver apartado 3.2), para esto es necesario el uso de la instrucción “join”, como medio de vinculación. A continuación la descripción de los términos más importantes (Ver ilustración 20):

```
# Table inv_congreso_autor
map:inv_congreso_autor1 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:participant;
  d2rq:refersToClassMap map:inv_congreso;
  d2rq:condition "inv_congreso_autor.es_externo = 0";
  d2rq:join "inv_congreso_autor.congreso_id = inv_congreso.congreso_id";
  d2rq:join "inv_congreso_autor.emp_id = empleado.emp_id";
.
```

Ilustración 20: Uso del JOIN - Vinculación entre tablas

- **“map:inv_congreso_autor1”**, nombre del mapeo que realizará la vinculación entre tablas.
- **“map:empleado”**, llamada a uno de los mapeos definidos anteriormente.
- **“dbpedia_prop:participant”**, uso de la propiedad DBpedia como medio para expresar la vinculación entre los mapeos definidos.
- **“map:inv_congreso”**, invocamos al segundo mapeo.
- **“inv_congreso_autor.es_externo=0”**, Condicional de uso.
- **“join”**, al igual que en SQL esta instrucción permite combinar tablas, para lo cual se definen equivalencias entre claves.

3. Obtención de tripletas RDF (Fichero NT):

Una vez creado el fichero de mapeos TTL se debe realizar el volcado de los datos a formato RDF. Este fichero contiene toda la información de la B.D., la cual está representada en forma de tripletas, cumpliendo así el objetivo principal del proyecto. En la ilustración 21 se puede observar dos tripletas, las cuales nos indican que los empleados 55 y 56 (emp_id) han participado en el congreso número 1 (congreso_id).

```
<http://localhost:2020/empleado/55> <http://dbpedia.org/property/participant>
<http://localhost:2020/inv_congreso/1> .
<http://localhost:2020/empleado/56> <http://dbpedia.org/property/participant>
<http://localhost:2020/inv_congreso/1> .
```

Ilustración 21: Tripletas RDF - empleado

Ahora, si analizamos la ilustración 22, podemos observar que la tripleta obtenida hace referencia al congreso 147 (congreso_id).

Este congreso tiene las siguientes propiedades “location”, “startDate”, “endDate”, y una etiqueta RDF para el Nombre del Congreso, así como también usamos la ontología Dublin Core para asociarle al Título del Congreso, y por último observamos que todas estas propiedades están definidas dentro de la Clase DBpedia “Convention” la cual es del tipo RDF.

Todos estos elementos son los que le proveen de Semántica a los datos RDF.

```
<http://localhost:2020/inv_congreso/147>
<http://dbpedia.org/property/location> "Burgos, Spain" .
<http://localhost:2020/inv_congreso/147>
<http://dbpedia.org/property/startDate> "2006-09-20" .
<http://localhost:2020/inv_congreso/147> <http://www.w3.org/2000/01/rdf-
schema#label> "Intelligent Data Engineering and Automated Learning (IDEAL
2006)" .
<http://localhost:2020/inv_congreso/147>
<http://purl.org/dc/elements/1.1/title> "Evaluation of Decision Tree Pruning
with Subadditive Penalties"@en .
<http://localhost:2020/inv_congreso/147> <http://dbpedia.org/property/endDate>
"2006-09-23" .
<http://localhost:2020/inv_congreso/147> <http://www.w3.org/1999/02/22-rdf-
syntax-ns#type> <http://dbpedia.org/ontology/Convention> .
```

Ilustración 22: Tripleta RDF – Congreso 147

4. Visualización de los datos (Queries):

Para visualizar los datos se deben ejecutar consultas por medio de lenguaje SPARQL. Es necesario mencionar que para poder ejecutar estas instrucciones se debe contar con un SPARQL Endpoint. La plataforma D2RQ incluye esta herramienta (Ver ilustración 23).

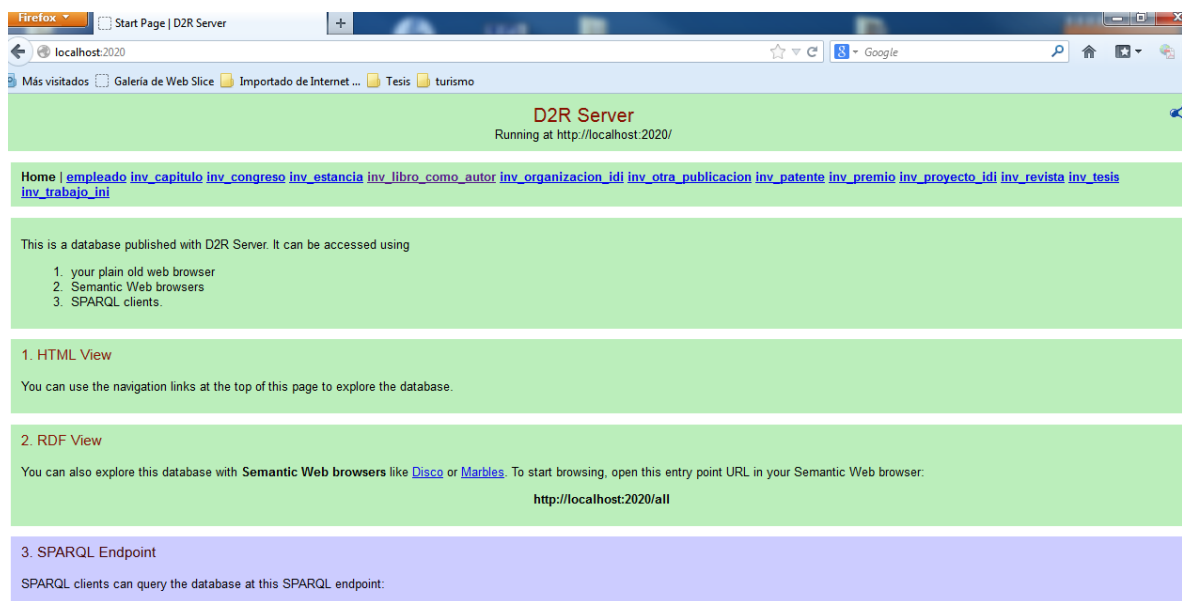


Ilustración 23: Pantalla Inicio D2R

Un SPARQL Endpoint proporciona a los clientes web un FrontEnd para que puedan realizar consultas por medio del lenguaje SPARQL. Este fue desarrollado en Ajax (**A**synchronous **J**ava **S**cript **A**nd **X**ML), el cual se complementa con Json (**J**ava **S**cript **O**bject **N**otation). Fue creado por Richard Cyganaik como parte del proyecto D2R Server. Actualmente esta licenciado bajo los términos de Apache-2 (41).

En la ilustración 24, se puede observar la estructura que maneja la pantalla principal (Snorql), la cual está dividida en tres bloques:

- El Bloque superior contiene todos los prefix o vocabularios definidos en la etapa de mapeo.
- En el bloque intermedio se ingresan las instrucciones SPARQL, es decir, instrucciones de búsqueda y filtrado. Por ejemplo:

```
SELECT ?Nombre ?Congreso
WHERE {
  ?emp foaf:name ?Nombre .
  ?emp dbpedia-prop:participant ?con .
  ?con dc:title ?Congreso .
  FILTER (?Congreso = "Machine Learning by Multi-feature Extraction Using Genetic Algorithms"@en)
}
```

- En el bloque inferior se muestran los resultados obtenidos. En el ejemplo propuesto anteriormente (instrucciones SPARQL), se desea saber qué investigadores han participado en el Congreso “Machine Learning by Multi-feature Extraction Using Genetic Algorithms”.
- El resultado obtenido permite concluir que cuatro docentes participaron en el mencionado Congreso (Ver ilustración 24).

Snorql: Exploring <http://localhost:2020/sparql>

SPARQL:

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX db: <http://localhost:2020/resource/>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dbpedia-prop: <http://dbpedia.org/property/>
PREFIX vcard: <http://www.w3.org/2001/vcard-rdf/3.0#>
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX map: <file:/Users/richard/D2RQ/workspace/D2RQ/doc/example/mapping.ttl#>
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX vocab: <http://localhost:2020/resource/vocab/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>

SELECT ?Nombre ?Congreso
WHERE {
  ?emp foaf:name ?Nombre .
  ?emp dbpedia-prop:participant ?con .
  ?con dc:title ?Congreso .
  FILTER (?Congreso = "Machine Learning by Multi-feature Extraction Using Genetic Algorithms"@en)
}
```

Results:

SPARQL results:

Nombre	Congreso
"Porta_Zamorano_Jordi"	"Machine Learning by Multi-feature Extraction Using Genetic Algorithms"@en
"Pulido_Cañabate_Estrella"	"Machine Learning by Multi-feature Extraction Using Genetic Algorithms"@en
"Pérez_Pérez_Eduardo"	"Machine Learning by Multi-feature Extraction Using Genetic Algorithms"@en
"Shafiq_Leila Shila"	"Machine Learning by Multi-feature Extraction Using Genetic Algorithms"@en

Ilustración 24: Pantalla de Ingreso y Obtención de datos

4. Conclusiones y Trabajos Futuros.

4.1. Conclusiones.

Este apartado incluye algunas recomendaciones a nivel general, así como también, algunas de forma específica (áreas), las cuales han sido clasificadas en dos grupos:

Web Semántica / Linked Data:

La Web Semántica se presenta como una nueva propuesta frente a la Web tradicional, destacando que la Web Semántica no pretende en ningún momento reemplazar a la Web tradicional, sino más bien complementarla; es decir, dotar de significado semántico a los datos.

La Web Semántica ha ido creciendo gracias al nivel de publicación y reutilización de los datos (RDF).

La ontología DBpedia define clases y propiedades que garantizan el nivel Semántico que tendrán los datos. Por esto DBpedia es una fuente referencial para muchos otros nodos (Linked Data), lo que hace la diferencia al momento de seleccionar a qué nodo nos vinculamos e intercambiamos información.

El uso de datos estructurados es un factor clave dentro de estas propuestas, ya que de esta forma los datos introducidos a la Web Semántica guardan estándares y patrones de usabilidad que facilitan su posterior reutilización.

Generación y publicación de datos:

En una B.D. tradicional los datos y el modelo son conocidos sólo por el administrador de la B.D. y por nadie más, a diferencia de una ontología.

Un fichero RDF es más flexible que los datos contenidos en un B.D. Por ejemplo: Añadir una columna o un dato a una B.D., es más complicado que incluirlo en una tripleta RDF ya que esta contiene toda la información de la B.D.

Los pasos descritos en el método utilizado fueron los adecuados para la obtención de datos RDF y su posterior publicación, para lo cual se tuvo en cuenta el origen de los datos y el tipo de información a publicar (5 estrellas).

D2RQ y SPARQL incluyen algunos principios relacionados a SQL con el objetivo de facilitar su uso, en especial SPARQL, permite la posibilidad de combinar diversos criterios de filtrado facilitando así, la obtención de resultados y familiarización de la plataforma.

Generales

El crecimiento de los datos ha sido el principal motivo para que exista mucho interés por parte de diversos grupos de investigación por analizar y recabar dichos datos. Esto ha influido para que el término Open Data haya ganado adeptos a nivel mundial.

Una ontología está representada por un modelo, procurando siempre la reutilización de dicho modelo, para el cual se deben definir grafos que representen al conocimiento de forma adecuada, enfatizando que si el modelo no está bien construido no se obtendrá una información coherente, es decir, tendríamos datos semánticos mal definidos.

La herramienta D2RQ no razona ni detecta inconsistencias semánticas en las tripletas generadas, ya que acepta como válidos los datos que no tienen ningún valor semántico, siendo esto una desventaja al momento de usar el fichero TTL.

No todos los datos pueden ser migrados a formato RDF, y en algunos casos es mejor conservarlos en formato relacional, ya que al ganar flexibilidad perdemos eficiencia. Aspecto clave al momento de tomar una decisión.

El uso de vocabularios estandarizados garantiza el nivel semántico que los datos poseen, así como, el nivel de reutilización.

El método utilizado cumple las recomendaciones propuestas por la herramienta D2RQ, como medio para la obtención de datos RDF a partir de una B.D. relacional.

Gracias al desarrollo de la guía didáctica y metodológica sobre el proceso de instalación y ejecución de la plataforma D2RQ, se ha conseguido ser ente multiplicador y llegar a tener contacto con otras personas del medio.

Se incursionó en el uso y manejo de la herramienta Google Refine, como otro medio para la obtención de tripletas RDF.

Actualmente el Ecuador no es parte de la iniciativa Open Data. Con el desarrollo del presente trabajo y colaboración de otros profesionales del medio, se podría conseguir introducir al país dentro de estas iniciativas mundiales.

4.2. Trabajos Futuros.

Con la investigación realizada en este trabajo, se abren los siguientes campos de acción:

- Dar a conocer la iniciativa Open Data dentro del Ecuador.
- Representar al Ecuador dentro de las iniciativas Open Data, las mismas que vienen dando sus primeros pasos en América Latina.

- Exponer las ventajas de publicar datos en formatos RDF, en especial dentro de algunas entidades gubernamentales del Ecuador.
- Cargar y actualizar datos dentro de la Wikipedia referentes a ciertas instituciones públicas del Ecuador, los cuales posteriormente podrían ser mapeados y publicados.

También se puede enumerar posibles investigaciones:

- Realizar un análisis de rendimiento entre las diferentes plataformas existentes, las cuales deben tener como principal característica que permitan manejar BB.DD. tradicionales. Este análisis debería ser realizado con gran cantidad de datos reales.
- Desarrollar un prototipo que permita almacenar y acceder a las queris generadas por grupos especializados, es decir, un repositorio de todas las instrucciones SPARQL generadas. Este prototipo serviría como medio de difusión y entrenamiento para las personas que desean incursionar en esta área.
- Realizar un estudio sobre los niveles de seguridad que pueden tener los datos publicados, es decir, nivel de confiabilidad de los datos origen.

5. Anexos

5.1 Ejemplos Significativos.

A continuación se describen las pruebas más significativas realizadas con los datos semánticos generados a partir de la B.D. de la EPS.

Ejercicio 1

Mostrar todos los investigadores que han obtenido un factor de impacto superior a 5, así como el nombre de la revista respectiva

```
SELECT ?Nombre ?Revista ?Factor
WHERE{
  ?emp foaf:name ?Nombre .
  ?emp dbpedia-prop:writer ?rev .
  ?rev dc:title ?Revista .
  ?rev dbpedia-prop:impactFactor ?Factor .
  FILTER (?Factor > "5")
}
```

SPARQL results:

Nombre	Revista	Factor
"Varona_Martínez_Pablo"	"Physical Review Letters"	"7.489"
"Sigüenza_Pizarro_Juan Alberto"	"PLoS Computational Biology"	"6.236"
"Varona_Martínez_Pablo"	"PLoS Computational Biology"	"6.236"
"Martínez_Muñoz_Gonzalo"	"IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGEN"	"5.96"
"Suárez_González_Alberto"	"IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGEN"	"5.96"
"Hernández_Lobato_Daniel"	"IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGEN"	"5.96"
"Martínez_Muñoz_Gonzalo"	"IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGEN"	"5.96"
"Suárez_González_Alberto"	"IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGEN"	"5.96"
"Hernández_Lobato_Daniel"	"IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGEN"	"5.96"
"González_Marcos_Ana María"	"BREAST CANCER RESEARCH"	"5.33"

Ejercicio 2

Mostrar cuál es el nombre del investigador que ha obtenido un factor de 7.489, así como el tipo y nombre de la revista donde han publicado, acompañado del factor de impacto correspondiente.

```
SELECT ?Nombre ?Revista ?Titulo ?Factor
WHERE{
  ?emp foaf:name ?Nombre .
  ?emp dbpedia-prop:writer ?prem .
  ?prem dbpedia-prop:category ?Revista .
  ?prem dc:title ?Titulo .
  ?prem dbpedia-prop:impactFactor?Factor .
  FILTER (?Factor = "7.489")
}
```

SPARQL results:

Nombre	Revista	Titulo	Factor
"Varona_Martinez_Pablo"	"Internacional"	"Physical Review Letters"	"7.489"

Ejercicio 3

Mostrar todos los investigadores que han publicado en revistas nacionales, así como, también el nombre de la revista.

```
SELECT ?Nombre ?Revista ?Tipo
WHERE{
  ?emp foaf:name ?Nombre .
  ?emp dbpedia-prop:writer ?prem .
  ?prem dc:title ?Revista .
  ?prem dbpedia-prop:category ?Tipo
  FILTER (?Tipo = "Nacional")
}
```

SPARQL results:

Nombre	Revista	Tipo
"Pérez_Marín_Diana Rosario"	"Revista Colombiana de Computación"	"Nacional"
"Acuña_Castillo_Silvia Teresita"	"Revista Colombiana de Computación"	"Nacional"
"Macías_Iglesias_José Antonio"	"Novática"	"Nacional"
"Rodríguez_Marín_Pilar"	"Novática"	"Nacional"
"Pérez_Marín_Diana Rosario"	"Novática"	"Nacional"
"Alfonseca_Moreno_Manuel"	"Religión y Cultura"	"Nacional"
"Alfonseca_Moreno_Manuel"	"BIT"	"Nacional"
"Alfonseca_Moreno_Manuel"	"BIT"	"Nacional"
"Alfonseca_Moreno_Manuel"	"BIT"	"Nacional"
"Alfonseca_Moreno_Manuel"	"BIT"	"Nacional"
"Alfonseca_Moreno_Manuel"	"BIT"	"Nacional"
"Pascual_Nieto_Ismael"	"Sociedad Española para el Procesamiento del Lenguaje Natural"	"Nacional"
"Pascual_Nieto_Ismael"	"Revista de la Sociedad Española para el Procesamiento del Le"	"Nacional"
"Koroutchev_Kostadin"	"C/IR/2007/002"	"Nacional"
"Rodríguez_Marín_Pilar"	"Revista de la Sociedad Española de PLN (SEPLN)"	"Nacional"
"Pérez_Marín_Diana Rosario"	"Revista de la Sociedad Española de PLN (SEPLN)"	"Nacional"
"Pascual_Nieto_Ismael"	"Revista de la Sociedad Española de PLN (SEPLN)"	"Nacional"
"López_Buedo_Sergio"	"Boletín de la red nacional de I+D.RedIRIS"	"Nacional"
"Pedro_Sánchez_Luis de"	"Boletín de la red nacional de I+D.RedIRIS"	"Nacional"
"López de Vergara_Méndez_Jorge E."	"Boletín de la red nacional de I+D.RedIRIS"	"Nacional"
"Aracil_Rico_Javier"	"Boletín de la red nacional de I+D.RedIRIS"	"Nacional"
"Hernández_Gutiérrez_José Alberto"	"Boletín de la red nacional de I+D.RedIRIS"	"Nacional"
"García_Dorado_José Luis"	"Boletín de la red nacional de I+D.RedIRIS"	"Nacional"
"Acuña_Castillo_Silvia Teresita"	"Revista Faz"	"Nacional"

Ejercicio 4

Mostrar todos los investigadores que han publicado en alguna revista, en un año dado (2010), así como, nombre y tipo de la revista.

```

SELECT ?Nombre ?Revista ?Tipo ?Fecha
WHERE{
  ?emp foaf:name ?Nombre .
  ?emp dbpedia-prop:writer ?prem .
  ?prem dc:title ?Revista .
  ?prem dbpedia-prop:category ?Tipo .
  ?prem dc:date ?Fecha .
  FILTER (?Fecha > "2010-01-01" && ?Fecha < "2010-12-31" )
}

```

SPARQL results:

Nombre	Revista	Tipo	Fecha
"Castro_Martin_Angel de"	"IEEE Transactions on Power Electronics"	"Internacional"	"2010-04-30"
"Castro_Martin_Angel de"	"IEEE Transactions on Power Electronics"	"Internacional"	"2010-05-31"
"Molina_Vela_Javier"	"IEEE Transactions on Power Electronics"	"Internacional"	"2010-05-31"
"Iglesias_Cavada_Juan"	"IEEE Transactions on Power Electronics"	"Internacional"	"2010-05-31"
"Mori_Pelaez_Fiorella"	"IEEE Transactions on Power Electronics"	"Internacional"	"2010-05-31"
"Castro_Martin_Angel de"	"IEEE Transactions on Industrial Electronics"	"Internacional"	"2010-09-30"
"Iglesias_Cavada_Juan"	"IEEE Transactions on Industrial Electronics"	"Internacional"	"2010-09-30"
"Santiago_del Rio_Pedro Maria"	"IEEE Transactions on Industrial Electronics"	"Internacional"	"2010-09-30"
"Garcia_Garcia_Miguel Angel"	"Pattern Recognition"	"Internacional"	"2010-11-30"

Ejercicio 5

Mostrar todos los investigadores que han participado en alguna ponencia o han escrito algún libro, mostrando el año.

```

SELECT ?nombre ?lugar ?Ponencia ?Libro ?Fecha_Libro
WHERE {
  { ?emp foaf:name ?nombre .
    ?emp dbpedia-prop:participant ?pon .
    ?pon dbpedia-prop:location ?lugar .
    ?pon dc:title ?Ponencia.
  }
  UNION
  {
    ?emp foaf:name ?nombre .
    ?emp dbpedia-prop:writer ?lib .
    ?lib dc: ?Libro .
    ?lib dc:date ?Fecha_Libro}
}
order by ?nombre

```


"Anguiano_Rey_Eloy"	"Stuttgart, Alemania"	"Graph domain decomposition for heterogeneous cluster programming"@en	-	-
"Anguiano_Rey_Eloy"	"Cracovia, Polonia"	"Parallel Metropolis-Montecarlo simulation for Potts model using an adaptable network topology based on dynamic graph partitioning"@en	-	-
"Anguiano_Rey_Eloy"	"Lyon, France"	"Parallel simulation of NEPs on clusters"@en	-	-
"Anguiano_Rey_Eloy"	"Burgos, Spain"	"Using Preferences to Solve Student-Class Allocation Problem"@en	-	-
"Anguiano_Rey_Eloy"	-	-	"Blended Learning"	"2008-01-01"
"Anguiano_Rey_Eloy"	-	-	"LINUX: guía de aprendizaje"	"2008-01-01"
"Anguiano_Rey_Eloy"	-	-	"PATTERN ANALYSIS AND APPLICATIONS"	"2009-11-30"
"Angulo_Pinedo_José María"	"Las Vegas, EEUU"	"Detection of Unusual Objects and Temporal Patterns in EEG Video Recordings"@en	-	-
"Antona_Montoro_Ana María"	"Sofia, Bulgaria"	"Allocating Educational Resources Through Happiness Optimization"@en	-	-
"Antona_Montoro_Ana María"	"La Palma, Canary Islands, Spain"	"Dynamics of a Three Neurons Network with Excitatory-inhibitory Interactions"@en	-	-
"Antona_Montoro_Ana María"	"Burgos, Spain"	"Using Preferences to Solve Student-Class Allocation Problem"@en	-	-
"Aparicio_Samuel"	-	-	"Annals of Telecommunications"	"2006-12-31"
"Aparicio_Samuel"	-	-	"IEEE Transactions on Information Forensics and Security"	"2007-11-30"
"Aparicio_Samuel"	-	-	"IEEE Trans. on Information Forensics and Security"	"2008-05-31"

Ejercicio 6

Mostrar quiénes han participado en una ponencia específica, mostrando el lugar y nombres de los profesionales involucrados:

```
SELECT ?nombre ?lugar ?titulo
WHERE{
  ?emp foaf:name ?nombre .
  ?emp dbpedia-prop:participant ?pon .
  ?pon dbpedia-prop:location ?lugar .
  ?pon dc:title ?titulo .
  FILTER(regex(?titulo, 'Pruning in Ordered Bagging Ensembles ')) .
}
```

SPARQL results:

nombre	lugar	titulo
"Martínez_Muñoz_Gonzalo"	"Carnegie Mellon Univ., Pittsburgh, USA"	"Pruning in Ordered Bagging Ensembles"@en
"Suárez_González_Alberto"	"Carnegie Mellon Univ., Pittsburgh, USA"	"Pruning in Ordered Bagging Ensembles"@en

Ejercicio 7

Mostrar el título de todas las ponencias en las que hayan participado todos los investigadores de la EPS, así como, lugar y fechas dadas (2007).

```
SELECT ?nombre ?titulo ?lugar ?inicio ?fin
WHERE{
  ?emp foaf:name ?nombre .
  ?emp dbpedia-prop:participant ?pon .
  ?pon dbpedia-prop:location ?lugar .
  ?pon dc:title ?titulo .
  ?pon dbpedia-prop:startDate ?inicio .
  ?pon dbpedia-prop:endDate ?fin .
}
```

FILTER (?inicio > "2007-01-01" && ?fin < "2007-12-31")

}

SPARQL results:

nombre	titulo	lugar	inicio	fin
"Macías_Iglesias_José Antonio"	"Interfaces Basados en la Web Semántica para la Gestión de Contenidos Multimedia"@en	"Zaragoza"	"2007-09-11"	"2007-09-14"
"Tiburzi_Paramio_Fabrizio"	"Interfaces Basados en la Web Semántica para la Gestión de Contenidos Multimedia"@en	"Zaragoza"	"2007-09-11"	"2007-09-14"
"Anguiano_Rey_Eloy"	"Architecture based on dynamic graphs for the dynamic reconfiguration of farms of computers"@en	"Viena, Austria"	"2007-05-25"	"2007-05-27"
"Garrigos_Aniorte_Gustavo"	"A Problem-Oriented Method for Supporting AEH Authors through Data Mining"@en	"Creta, Greece"	"2007-09-17"	"2007-09-20"
"Polop_Jorda_Celia"	"A Problem-Oriented Method for Supporting AEH Authors through Data Mining"@en	"Creta, Greece"	"2007-09-17"	"2007-09-20"
"Esteve_Elizalde_Cristina"	"A Problem-Oriented Method for Supporting AEH Authors through Data Mining"@en	"Creta, Greece"	"2007-09-17"	"2007-09-20"
"García_del_Amo_Ignacio José"	"A Problem-Oriented Method for Supporting AEH Authors through Data Mining"@en	"Creta, Greece"	"2007-09-17"	"2007-09-20"
"Piqueras_Piqueras_Juan"	"A Problem-Oriented Method for Supporting AEH Authors through Data Mining"@en	"Creta, Greece"	"2007-09-17"	"2007-09-20"
"Carro_Salas_Rosa María"	"Empowering AEH Authors Using Data Mining Techniques"@en	"Corfu, Greece"	"2007-06-25"	"2007-06-29"
"Estébanez_Rodríguez_Pablo"	"Empowering AEH Authors Using Data Mining Techniques"@en	"Corfu, Greece"	"2007-06-25"	"2007-06-29"
"Martínez_Sánchez_José María"	"Adapting Surveillance Video to Small Displays via Object-based Video Cropping"@en	"Santorini, Grecia"	"2007-05-31"	"2007-05-31"
"Herranz_Aribas_Luis"	"Adapting Surveillance Video to Small Displays via Object-based Video Cropping"@en	"Santorini, Grecia"	"2007-05-31"	"2007-05-31"
"Martínez_Sánchez_José María"	"Multimedia Content Adaptation Modelled as a Constraints Matching Problem with Optimisation"@en	"Santorini, Grecia"	"2007-05-31"	"2007-05-31"
"López_Hernández_Fernando"	"Multimedia Content Adaptation Modelled as a Constraints Matching Problem with Optimisation"@en	"Santorini, Grecia"	"2007-05-31"	"2007-05-31"
"Martínez_Sánchez_José María"	"On-line Video Skimming Based on Histogram Similarity"@en	"Augsbrugo, Alemania"	"2007-08-31"	"2007-08-31"
"García_García_Miguel Angel"	"Comparative Evaluation of Classical Methods, Optimized Gabor Filters and LBP for Texture Feature Selection and Classification"@en	"Viena, Austria"	"2007-08-27"	"2007-08-29"
"Vukotic_Jovsic_Dragan"	"Graph-Based Perceptual Segmentation of Stereo Vision 3D Images at Multiple Abstraction Levels"@en	"Alicante, España"	"2007-05-31"	"2007-05-31"
"García_García_Miguel Angel"	"Graph-Based Perceptual Segmentation of Stereo Vision 3D Images at Multiple Abstraction Levels"@en	"Alicante, España"	"2007-05-31"	"2007-05-31"
"García_García_Miguel Angel"	"Supervised Pixel-Based Texture Classification with Gabor Wavelet Filters"@en	"Bielefeld, Alemania"	"2007-02-28"	"2007-02-28"
"Bescós_Cano_Jesús"	"Content-driven adaptation of on-line video"@en	"Burdeos, Francia"	"2007-05-31"	"2007-05-31"
"Martínez_Sánchez_José María"	"Content-driven adaptation of on-line video"@en	"Burdeos, Francia"	"2007-05-31"	"2007-05-31"
"Herranz_Aribas_Luis"	"Content-driven adaptation of on-line video"@en	"Burdeos, Francia"	"2007-05-31"	"2007-05-31"
"Tiburzi_Paramio_Fabrizio"	"Content-driven adaptation of on-line video"@en	"Burdeos, Francia"	"2007-05-31"	"2007-05-31"
"Martínez_Sánchez_José María"	"On the effect of motion segmentation techniques in description based adaptive video transmission"@en	"Londres, Reino Unido"	"2007-08-31"	"2007-08-31"
"San Miguel_Avedillo_Juan Carlos"	"On the effect of motion segmentation techniques in description based adaptive video"	"Londres, Reino Unido"	"2007-08-31"	"2007-08-31"

Ejercicio 8

Complementando el ejercicio anterior, pero ahora se desea saber para un investigador dado.

```
SELECT ?nombre ?titulo ?lugar ?inicio ?fin
WHERE{
  ?emp foaf:name ?nombre .
  FILTER(regex(?nombre,'García_García_Miguel Angel')) .
  ?emp dbpedia-prop:participant ?pon .
  ?pon dbpedia-prop:location ?lugar .
  ?pon dc:title ?titulo .
  ?pon dbpedia-prop:startDate ?inicio .
  ?pon dbpedia-prop:endDate ?fin .
  FILTER (?inicio > "2007-01-01" && ?fin < "2007-12-31")
}
```

SPARQL results:

nombre	titulo	lugar	inicio	fin
"García_García_Miguel Angel"	"Comparative Evaluation of Classical Methods, Optimized Gabor Filters and LBP for Texture Feature Selection and Classification"@en	"Viena, Austria"	"2007-08-27"	"2007-08-29"
"García_García_Miguel Angel"	"Graph-Based Perceptual Segmentation of Stereo Vision 3D Images at Multiple Abstraction Levels"@en	"Alicante, España"	"2007-05-31"	"2007-05-31"
"García_García_Miguel Angel"	"Supervised Pixel-Based Texture Classification with Gabor Wavelet Filters"@en	"Bielefeld, Alemania"	"2007-02-28"	"2007-02-28"

Ejercicio 9

De un investigador dado, mostrar el titulo de la ponencia, así como el lugar donde fue impartida.

```
SELECT ?nombre ?lugar ?titulo
WHERE{
  ?emp foaf:name ?nombre .
  FILTER(regex(?nombre,'Taha_Ahmed_Bazil')) .
  ?emp dbpedia-prop:participant ?pon .
  ?pon dbpedia-prop:location ?lugar .
  ?pon dc:title ?titulo
}
```

SPARQL results:

nombre	lugar	titulo
"Taha_Ahmed_Bazil"	"Madrid"	"IMPACTO DE LOS SISTEMAS ULTRA WIDE BAND (UWB) EN LOS SISTEMAS WIMAX A 3.5 GHZ"@en

Ejercicio 10

Desplegar que tipo de reconocimiento que ha recibido un investigador dado, así como también la entidad que emitió tal reconocimiento y la fecha.

```
SELECT ?nombre ?premio ?entidad ?fecha
WHERE{
  ?emp foaf:name ?nombre .
  FILTER(regex(?nombre,'Taha_Ahmed_Bazil')) .
  ?emp dbpedia-prop:award ?prem .
  ?prem dbpedia-prop:category ?premio .
  ?prem dbpedia-prop: foundedBy?entidad .
  ?prem dc:date ?fecha .
}
```

SPARQL results:

nombre	premio	entidad	fecha
"Taha_Ahmed_Bazil"	"Premio"	"Universidad Autónoma de Madrid"	"2006-06-29"

Ejercicio 11

Desplegar Apellidos y Nombres de todos los investigadores que han recibido alguna "Beca", para lo cual se desea saber el nombre de la Entidad, en una fecha dada.

```

SELECT ?nombre ?premio ?entidad ?fecha
WHERE{
  ?emp foaf:name ?nombre .
  ?prem dbpedia-prop: category?premio .
  FILTER (?premio = "Beca")
  ?emp dbpedia-prop:award ?prem .
  ?prem dbpedia-prop: foundedBy ?entidad .
  ?prem dc:date ?fecha,"2007-04-01"
}
ORDER BY (?name)

```

SPARQL results:

nombre	premio	entidad	fecha
"Fiérrez_Aguilar_Julián"	"Beca"	"European Commission"	"2007-04-01"

Ejercicio 12

Desplegar Apellidos y Nombres de todos los investigadores que han recibido alguna Beca.

SPARQL results:

name	premio
"Abejón_González_Alejandro"	"Beca"
"Acuña_Castillo_Silvia Teresita"	"Beca"
"Alonso_Fernández_Fernando"	"Beca"
"Bellogín_Kouki_Alejandro"	"Beca"
"Bescós_Cano_Jesús"	"Beca"
"Bravo_Agapito_Javier"	"Beca"
"Camacho_Fernández_David"	"Beca"
"Castro_Martín_Angel de"	"Beca"
"Cobos_Pérez_Ruth"	"Beca"
"Fernández_Álvarez_Antonio Carlos"	"Beca"
"Fiérrez_Aguilar_Julián"	"Beca"
"Franco_Pedroso_Javier"	"Beca"
"Galbally_Herrero_Javier"	"Beca"
"González_Domínguez_Javier"	"Beca"

```

SELECT Distinct ?name ?premio
WHERE{
  ?emp foaf:name ?name .
  ?prem dbpedia-prop:category ?premio .
  FILTER (?premio = "Beca")
  ?emp dbpedia-prop:award ?pre
}
ORDER BY (?name)

```

Ejercicio 13

Desplegar Apellidos y Nombres de todos los investigadores, así como también el grupo de investigación al que pertenecen.

SPARQL results:

Nombre	Grupo
"Abejón_González_Alejandro"	"ATVS"
"Acero_González_Víctor"	""
"Acuña_Castillo_Silvia Teresita"	"GHIA"
"Aguayo_Navarrete_Estanielao"	"DSLlab"
"Aguilar_Morales_Antonio"	""
"Aguirre_Maeso_Carlos"	"GNB"
"Alamán_Roldán_Xavier"	"GHIA"
"Alarcón_Rodríguez_Miren Idoia"	""
"Alaiz_Gudín_Carlos María"	"GAA"
"Albertos_Bofarull_Ana"	""
"Alcedo_Moreno_Borja"	""
"Alfaya_Sánchez_David"	""
"Alfonseca_Cubero_Enrique"	""
"Alfonseca_Moreno_Manuel"	"GHIA"
"Aliev_Timour"	""
"Almeida_Cano_Elena"	""
"Alonso_Fernández_Fernando"	"ATVS"

```
SELECT ?Nombre ?Grupo
WHERE{
  ?emp foaf:name ?Nombre .
  ?emp dbpedia-prop:pseudonym ?Grupo .
}
ORDER BY (?Nombre)
```

Ejercicio 14

Desplegar cuál es el mayor factor obtenido entre todos los docentes.

SPARQL results:

.1
"7.489"

```
SELECT MAX(?Factor)
WHERE{
  ?rev dbpedia-prop:impactFactor ?Factor .
}
```

Ejercicio 15

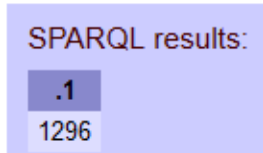
Mostrar qué docente dirigió la siguiente tesis: "Matrix Graph Grammars".

```
SELECT ?Nombre ?Tesis ?Universidad
WHERE{
  ?emp foaf:name ?Nombre .
  ?emp dbpedia-prop:projectCoordinator ?dir .
  ?dir dc:title ?Tesis .
  FILTER (?Tesis = "Matrix Graph Grammars") .
  ?dir dbpedia-prop:University ?Universidad .
}
```

Nombre	Tesis	Universidad
"Lara_Jaramillo_Juan de"	"Matrix Graph Grammars"	"Universidad Autónoma de Madrid"

Ejercicio 16

Cuántos investigadores han publicado en revistas internacionales.



```
SELECT count(?Tipo)
WHERE{
  ?emp foaf:name ?Nombre .
  ?emp dbpedia-prop:writer ?prem .
  ?prem dc:title ?Revista .
  ?prem dbpedia-prop:category ?Tipo .
  FILTER (?Tipo = "Internacional")
}
```

5.2 Recomendaciones y Buenas Prácticas sobre D2RQ.

Estas recomendaciones fueron concebidas con la finalidad de proveer una guía más didáctica y metodológica sobre el proceso de instalación y ejecución de la plataforma D2RQ sobre ambientes Windows. La información publicada en el sitio web oficial (<http://d2rq.org/>) es muy compleja de entender, más aun para personas que por primera vez se involucran con esta herramienta, así como también, sobre otros procesos relacionados a la instalación de la plataforma D2RQ.

Es por ello que esta guía de buenas prácticas ha sido desarrollada en idioma español y tratando de usar términos simples y fáciles de seguir en aspectos como:

- Instalación
- Mapeos
- Creación de tripletas
- Consultas

Adicionalmente, este documento ha sido publicado dentro del portal de la DBpedia en Español (<http://es.dbpedia.org/Wiki.jsp?page=Generaci%C3%B3n%20de%20datos%20sem%C3%A1nticos%20a%20partir%20de%20una%20base%20de%20datos>), con el objetivo de contribuir al crecimiento de la Web Semántica.

Es necesario mencionar que gracias a este link se ha podido interactuar con algunas personas hispanas que les interesa la herramienta D2RQ, así como también les atrae la idea del Open Data. Por lo tanto, gracias a esta guía didáctica se ha conseguido ser ente multiplicador de conocimiento.

5.3 Fichero TTL

A continuación se muestra el fichero .TTL, el cual contiene toda la información de los mapeos realizados a la B.D. de la EPS – UAM, mediante este fichero de mapeos se obtuvieron todas las tripletas RDF. Para mayor detalle ver el apartado 3.2.

```
@prefix map:
<file:/Users/richard/D2RQ/workspace/D2RQ/doc/example/mapping.ttl#> .
@prefix db: <> .
@prefix vocab: <vocab/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix d2rq: <http://www.wiwiw.fu-berlin.de/suhl/bizer/D2RQ/0.1#> .
@prefix jdbc: <http://d2rq.org/terms/jdbc/> .
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix skos: <http://www.w3.org/2004/02/skos/core#> .
@prefix vcard: <http://www.w3.org/2001/vcard-rdf/3.0#> .
@prefix dbpedia-owl: <http://dbpedia.org/ontology/> .
@prefix dbpedia-prop: <http://dbpedia.org/property/> .
@prefix dbpedia: <http://dbpedia.org/resource/> .

map:database a d2rq:Database;
  d2rq:jdbcDriver "com.mysql.jdbc.Driver";
  d2rq:jdbcDSN "jdbc:mysql://127.0.0.1/eps?autoReconnect=true";
  d2rq:username "root";
  d2rq:password "*****";
  jdbc:autoReconnect "true";
  jdbc:zeroDateTimeBehavior "convertToNull";
  jdbc:keepAlive "3600";          # sends noop-query every 3600 seconds
.

# Table empleado
map:empleado a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "empleado/@@empleado.emp_id@";
  d2rq:class dbpedia-owl:Scientist;
.

map:empleado_Nombre a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property foaf:name;
  d2rq:property rdfs:label;
  d2rq:pattern "@@empleado.ap1@@_@@empleado.ap2@@_@@empleado.nombre@";
.

map:empleado_lastmod a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dc:date;
  d2rq:column "empleado.lastmod";
```

```

.
map:empleado_siglas a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:pseudonym;
  d2rq:column "empleado.siglas_grupo";
.

# Table inv_congreso
map:inv_congreso a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "inv_congreso/@@inv_congreso.congreso_id@";
  d2rq:class dbpedia-owl:Convention;
  d2rq:classDefinitionLabel "Congreso";
.
map:inv_congreso_titulo_ponencia a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_congreso;
  d2rq:property dc:title;
  d2rq:column "inv_congreso.titulo_ponencia";
  d2rq:lang "en";
.
map:inv_congreso_nombre_congreso a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_congreso;
  d2rq:property rdfs:label;
  d2rq:column "inv_congreso.nombre_congreso";
.
map:inv_congreso_lugar_congreso a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_congreso;
  d2rq:property dbpedia-prop:location;
  d2rq:column "inv_congreso.lugar_congreso";
.
map:inv_congreso_fecha_ini a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_congreso;
  d2rq:property dbpedia-prop:startDate;
  d2rq:column "inv_congreso.fecha_ini";
.
map:inv_congreso_fecha_fin a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_congreso;
  d2rq:property dbpedia-prop:endDate;
  d2rq:column "inv_congreso.fecha_fin";
.

# Table inv_congreso_autor
map:inv_congreso_autor1 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:participant;
  d2rq:belongsToClassMap map:inv_congreso;
  d2rq:condition "inv_congreso_autor.es_externo = 0";
  d2rq:join "inv_congreso_autor.congreso_id = inv_congreso.congreso_id";
  d2rq:join "inv_congreso_autor.emp_id = empleado.emp_id";
.
map:inv_congreso_autor2 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:participant;
  d2rq:belongsToClassMap map:inv_congreso;
  d2rq:condition "inv_congreso_autor.es_externo = 1";
  d2rq:join "inv_congreso_autor.congreso_id = inv_congreso.congreso_id";
  d2rq:join "inv_congreso_autor.emp_id = empleado.emp_id";

```



```

.
# Table inv_congreso_editor
map:inv_congreso_editor1 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:participant;
  d2rq:refersToClassMap map:inv_congreso;
  d2rq:condition "inv_congreso_editor.es_externo = 0";
  d2rq:join "inv_congreso_editor.congreso_id =
inv_congreso.congreso_id";
  d2rq:join "inv_congreso_editor.emp_id = empleado.emp_id";
.
map:inv_congreso_editor2 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:participant;
  d2rq:refersToClassMap map:inv_congreso;
  d2rq:condition "inv_congreso_editor.es_externo = 1";
  d2rq:join "inv_congreso_editor.congreso_id =
inv_congreso.congreso_id";
  d2rq:join "inv_congreso_editor.emp_id = empleado.emp_id";
.

# Table inv_premio
map:inv_premio a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "inv_premio/@@inv_premio.premio_id@";
  d2rq:class dbpedia-owl:Award;
  d2rq:classDefinitionLabel "Premio";
.
map:inv_premio_fecha_premio a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_premio;
  d2rq:property dc:date;
  d2rq:column "inv_premio.fecha_premio";
  d2rq:propertyDefinitionLabel "Fecha Premio";
.
map:inv_premio_entidad_premio a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_premio;
  d2rq:property dbpedia-prop:foundedBy ;
  d2rq:column "inv_premio.entidad_premio";
  d2rq:propertyDefinitionLabel "Entidad Premio";
.
map:inv_premio_tipo_premio a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_premio;
  d2rq:property dbpedia-prop:category;
  d2rq:column "inv_premio.tipo_premio";
  d2rq:propertyDefinitionLabel "Tipo Premio";
.

# Table inv_premio_invest
map:inv_premio_invest1 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:award;
  d2rq:refersToClassMap map:inv_premio;
  d2rq:condition "inv_premio_invest.es_externo = 0";
  d2rq:join "inv_premio_invest.premio_id = inv_premio.premio_id";
  d2rq:join "inv_premio_invest.emp_id = empleado.emp_id";

```

```

.
map:inv_premio_invest2 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:award;
  d2rq:refersToClassMap map:inv_premio;
  d2rq:condition "inv_premio_invest.es_externo = 1";
  d2rq:join "inv_premio_invest.premio_id = inv_premio.premio_id";
  d2rq:join "inv_premio_invest.emp_id = empleado.emp_id";
.

# Table inv_libro_como_autor
map:inv_libro_como_autor a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern
"inv_libro_como_autor/@@inv_libro_como_autor.isbn_libro@@";
  d2rq:class dbpedia-owl:Book;
  d2rq:classDefinitionLabel "Libro";
.

map:inv_libro_como_autor_titulo_libro a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_libro_como_autor;
  d2rq:property dc:title;
  d2rq:column "inv_libro_como_autor.titulo_libro";
.

map:inv_libro_como_autor_anyo_libro a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_libro_como_autor;
  d2rq:property dc:date;
  d2rq:column "inv_libro_como_autor.anyo_libro";
.

map:inv_libro_como_autor_editorial_libro a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_libro_como_autor;
  d2rq:property dbpedia-prop:editorTitle;
  d2rq:column "inv_libro_como_autor.editorial_libro";
  d2rq:propertyDefinitionLabel "Editorial";
.

map:inv_libro_como_autor_lugar_libro a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_libro_como_autor;
  d2rq:property dbpedia-prop:place;
  d2rq:column "inv_libro_como_autor.lugar_libro";
  d2rq:propertyDefinitionLabel "Lugar";
.

# Table inv_libro_autor
map:inv_libro_autor1 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:writer;
  d2rq:refersToClassMap map:inv_libro_como_autor;
  d2rq:condition "inv_libro_autor.es_externo = 1";
  d2rq:join "inv_libro_autor.isbn_libro =
inv_libro_como_autor.isbn_libro";
  d2rq:join "inv_libro_autor.emp_id = empleado.emp_id";
.

map:inv_libro_autor2 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:writer;
  d2rq:refersToClassMap map:inv_libro_como_autor;
  d2rq:condition "inv_libro_autor.es_externo = 0";

```

```

d2rq:join "inv_libro_autor.isbn_libro =
inv_libro_como_autor.isbn_libro";
d2rq:join "inv_libro_autor.emp_id = empleado.emp_id";
.

# Table inv_capitulo
map:inv_capitulo a d2rq:ClassMap;
d2rq:dataStorage map:database;
d2rq:uriPattern "inv_capitulo/@@inv_capitulo.isbn_libro@";
d2rq:class dbpedia-owl:Book;
d2rq:classDefinitionLabel "Libro";
.

map:inv_capitulo_titulo_libro a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:inv_capitulo;
d2rq:property dc:title;
d2rq:column "inv_capitulo.titulo_libro";
d2rq:classDefinitionLabel "Libro";
.

map:inv_capitulo_anyo_libro a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:inv_capitulo;
d2rq:property dc:date;
d2rq:column "inv_capitulo.anyo_libro";
.

map:inv_capitulo_editorial_libro a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:inv_capitulo;
d2rq:property dbpedia-prop:editorTitle;
d2rq:column "inv_capitulo.editorial_libro";
d2rq:propertyDefinitionLabel "Editorial";
.

map:inv_capitulo_ciudad_libro a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:inv_capitulo;
d2rq:property dbpedia-prop:city;
d2rq:column "inv_capitulo.ciudad_libro";
d2rq:propertyDefinitionLabel "Lugar";
.

map:inv_capitulo_titulo_capitulo a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:inv_capitulo;
d2rq:property dbpedia-prop:projectObjective;
d2rq:column "inv_capitulo.titulo_capitulo";
d2rq:propertyDefinitionLabel "Capitulo";
.

# Table inv_capitulo_autor
map:inv_capitulo_autor1 a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:empleado;
d2rq:property dbpedia-prop:writer;
d2rq:refersToClassMap map:inv_capitulo;
d2rq:condition "inv_capitulo_autor.es_externo = 1";
d2rq:join "inv_capitulo_autor.isbn_libro =
inv_capitulo.isbn_libro";
d2rq:join "inv_capitulo_autor.emp_id = empleado.emp_id";
.

map:inv_capitulo_autor2 a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:empleado;
d2rq:property dbpedia-prop:writer;
d2rq:refersToClassMap map:inv_capitulo;

```

```

d2rq:condition "inv_capitulo_autor.es_externo = 0";
d2rq:join "inv_capitulo_autor.isbn_libro =
inv_capitulo.isbn_libro";
d2rq:join "inv_capitulo_autor.emp_id = empleado.emp_id";
.

# Table inv_estancia
map:inv_estancia a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "inv_estancia/@@inv_estancia.estancia_id@";
  d2rq:class dbpedia-owl:Event;
  d2rq:classDefinitionLabel "Estancia";
.
map:inv_estancia_centro_estancia a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_estancia;
  d2rq:property dbpedia-prop:center;
  d2rq:column "inv_estancia.centro_estancia";
.
map:inv_estancia_tipo_estancia a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_estancia;
  d2rq:property dbpedia-prop:category;
  d2rq:column "inv_estancia.tipo_estancia";
.
map:inv_estancia_tema_estancia a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_estancia;
  d2rq:property dbpedia-prop:academicDiscipline;
  d2rq:column "inv_estancia.tema_estancia";
.
map:inv_estancia_lugar_estancia a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_estancia;
  d2rq:property dbpedia-prop:place;
  d2rq:column "inv_estancia.lugar_estancia";
.
map:inv_estancia_fecha_ini a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_estancia;
  d2rq:property dbpedia-prop:startDate;
  d2rq:column "inv_estancia.fecha_ini";
.
map:inv_estancia_fecha_fin a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_estancia;
  d2rq:property dbpedia-prop:endDate;
  d2rq:column "inv_estancia.fecha_fin";
.

# Table inv_estancia_investigador
map:inv_estancia_investigador1 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:place;
  d2rq:refersToClassMap map:inv_estancia;
  d2rq:join "inv_estancia_investigador.estancia_id =
inv_estancia.estancia_id";
  d2rq:join "inv_estancia_investigador.emp_id = empleado.emp_id";
.

# Table inv_otra_publicacion
map:inv_otra_publicacion a d2rq:ClassMap;

```

```

        d2rq:dataStorage map:database;
        d2rq:uriPattern
"inv_otra_publicacion/@@inv_otra_publicacion.otra_public_id@";
        d2rq:class dbpedia-owl:Event;
        d2rq:classDefinitionLabel "Publicacion";
        .
map:inv_otra_publicacion_titulo_publici a d2rq:PropertyBridge;
        d2rq:belongsToClassMap map:inv_otra_publicacion;
        d2rq:property dc:title;
        d2rq:column "inv_otra_publicacion.titulo_publici";
        d2rq:classDefinitionLabel "Titulo";
        .
map:inv_otra_publicacion_fecha_otra_publici a d2rq:PropertyBridge;
        d2rq:belongsToClassMap map:inv_otra_publicacion;
        d2rq:property dc:date;
        d2rq:column "inv_otra_publicacion.fecha_otra_publici";
        .

# Table inv_otra_publici_autor
map:inv_otra_publici_autor1 a d2rq:PropertyBridge;
        d2rq:belongsToClassMap map:empleado;
        d2rq:property dbpedia-prop:writer;
        d2rq:refersToClassMap map:inv_otra_publicacion;
        d2rq:condition "inv_otra_publici_autor.es_externo = 1";
        d2rq:join "inv_otra_publici_autor.otra_public_id =
inv_otra_publicacion.otra_public_id";
        d2rq:join "inv_otra_publici_autor.emp_id = empleado.emp_id";
        .

map:inv_otra_publici_autor2 a d2rq:PropertyBridge;
        d2rq:belongsToClassMap map:empleado;
        d2rq:property dbpedia-prop:writer;
        d2rq:refersToClassMap map:inv_otra_publicacion;
        d2rq:condition "inv_otra_publici_autor.es_externo = 0";
        d2rq:join "inv_otra_publici_autor.otra_public_id =
inv_otra_publicacion.otra_public_id";
        d2rq:join "inv_otra_publici_autor.emp_id = empleado.emp_id";
        .

# Table inv_organizacion_idi
map:inv_organizacion_idi a d2rq:ClassMap;
        d2rq:dataStorage map:database;
        d2rq:uriPattern
"inv_organizacion_idi/@@inv_organizacion_idi.organizacion_id@";
        d2rq:class dbpedia-owl:Organisation;
        d2rq:classDefinitionLabel "Organizacion";
        .
map:inv_organizacion_idi_actividad_org a d2rq:PropertyBridge;
        d2rq:belongsToClassMap map:inv_organizacion_idi;
        d2rq:property dc:title;
        d2rq:column "inv_organizacion_idi.actividad_org";
        d2rq:classDefinitionLabel "Actividad";
        .
map:inv_organizacion_cargo_org a d2rq:PropertyBridge;
        d2rq:belongsToClassMap map:inv_organizacion_idi;
        d2rq:property dbpedia-prop:position;

```

```

    d2rq:column "inv_organizacion_idi.cargo_org";
    .
map:inv_organizacion_fecha_org a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_organizacion_idi;
    d2rq:property dc:date;
    d2rq:column "inv_organizacion_idi.fecha_org";
    .
map:inv_organizacion_lugar_org a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_organizacion_idi;
    d2rq:property dbpedia-prop:place;
    d2rq:column "inv_organizacion_idi.lugar_org";
    .

# Table inv_organizacion_idi_invest
map:inv_organizacion_idi_invest a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:empleado;
    d2rq:property dbpedia-prop:participant;
    d2rq:refersToClassMap map:inv_organizacion_idi;
    d2rq:join "inv_organizacion_idi_invest.organizacion_id =
    inv_organizacion_idi.organizacion_id";
    d2rq:join "inv_organizacion_idi_invest.emp_id = empleado.emp_id";
    .

# Table inv_revista
map:inv_revista a d2rq:ClassMap;
    d2rq:dataStorage map:database;
    d2rq:uriPattern "inv_revista/@@inv_revista.revista_id@";
    d2rq:class dbpedia-owl:Magazine;
    d2rq:classDefinitionLabel "Revista";
    .
map:inv_revista_titulo_articulo a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_revista;
    d2rq:property dbpedia-prop:literaryGenre;
    d2rq:column "inv_revista.titulo_articulo";
    .
map:inv_revista_titulo_revista a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_revista;
    d2rq:property dc:title;
    d2rq:column "inv_revista.titulo_revista";
    .
map:inv_revista_fecha_publi a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_revista;
    d2rq:property dc:date;
    d2rq:column "inv_revista.fecha_publi";
    .
map:inv_revista_tipo_revista a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_revista;
    d2rq:property dbpedia-prop:category;
    d2rq:column "inv_revista.tipo_revista";
    .
map:inv_revista_factor_revista a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_revista;
    d2rq:property dbpedia-prop:impactFactor;
    d2rq:column "inv_revista.factor_revista";
    .

```

```

# Table inv_revista_autor
map:inv_revista_autor1 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:writer;
  d2rq:refersToClassMap map:inv_revista;
  d2rq:condition "inv_revista_autor.es_externo = 1";
  d2rq:join "inv_revista.revista_id =
inv_revista_autor.revista_id";
  d2rq:join "inv_revista_autor.emp_id = empleado.emp_id";
.
map:inv_revista_autor2 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:writer;
  d2rq:refersToClassMap map:inv_revista;
  d2rq:condition "inv_revista_autor.es_externo = 0";
  d2rq:join "inv_revista.revista_id =
inv_revista_autor.revista_id";
  d2rq:join "inv_revista_autor.emp_id = empleado.emp_id";
.

# Table inv_tesis
map:inv_tesis a d2rq:ClassMap;
  d2rq:dataStorage map:database;
  d2rq:uriPattern "inv_tesis/@@inv_tesis.tesis_id@";
  d2rq:class dbpedia-owl:ResearchProject;
  d2rq:classDefinitionLabel "Tesis";
.
map:inv_tesis_titulo_tesis a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_tesis;
  d2rq:property dc:title;
  d2rq:column "inv_tesis.titulo_tesis";
.
map:inv_tesis_universidad_tesis a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_tesis;
  d2rq:property dbpedia-prop:University;
  d2rq:column "inv_tesis.universidad_tesis";
.
map:inv_tesis_fecha_tesis a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:inv_tesis;
  d2rq:property dc:date;
  d2rq:column "inv_tesis.fecha_tesis";
.

# Table inv_tesis_director
map:inv_tesis_director1 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:projectCoordinator;
  d2rq:refersToClassMap map:inv_tesis;
  d2rq:condition "inv_tesis_director.es_externo_dir = 1";
  d2rq:join "inv_tesis_director.tesis_id = inv_tesis.tesis_id";
  d2rq:join "inv_tesis_director.emp_id = empleado.emp_id";
.
map:inv_tesis_director2 a d2rq:PropertyBridge;
  d2rq:belongsToClassMap map:empleado;
  d2rq:property dbpedia-prop:projectCoordinator;
  d2rq:refersToClassMap map:inv_tesis;
  d2rq:condition "inv_tesis_director.es_externo_dir = 0";

```

```

    d2rq:join "inv_tesis_director.tesis_id = inv_tesis.tesis_id";
    d2rq:join "inv_tesis_director.emp_id = empleado.emp_id";
    .

# Table inv_trabajo_ini
map:inv_trabajo_ini a d2rq:ClassMap;
    d2rq:dataStorage map:database;
    d2rq:uriPattern "inv_trabajo_ini/@@inv_trabajo_ini.trabajo_ini_id@";
    d2rq:class dbpedia-owl:ResearchProject;
    d2rq:classDefinitionLabel "Trabajo";
    .

map:inv_trabajo_ini_titulo_trabajo a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_trabajo_ini;
    d2rq:property dc:title;
    d2rq:column "inv_trabajo_ini.titulo_trabajo";
    .

map:inv_trabajo_ini_universidad_trabajo a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_trabajo_ini;
    d2rq:property dbpedia-prop:University;
    d2rq:column "inv_trabajo_ini.universidad_trabajo";
    .

map:inv_trabajo_ini_fecha_trabajo a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_trabajo_ini;
    d2rq:property dc:date;
    d2rq:column "inv_trabajo_ini.fecha_trabajo";
    .

# Table inv_trabajo_tutor
map:inv_trabajo_tutor a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:empleado;
    d2rq:property dbpedia-prop:projectCoordinator;
    d2rq:refersToClassMap map:inv_trabajo_ini;
    d2rq:join "inv_trabajo_ini.trabajo_ini_id =
inv_trabajo_tutor.trabajo_ini_id";
    d2rq:join "inv_trabajo_tutor.emp_id = empleado.emp_id";
    .

# Table inv_proyecto_idi
map:inv_proyecto_idi a d2rq:ClassMap;
    d2rq:dataStorage map:database;
    d2rq:uriPattern
"inv_proyecto_idi/@@inv_proyecto_idi.referencia_proyecto@";
    d2rq:class dbpedia-owl:ResearchProject;
    d2rq:classDefinitionLabel "Proyecto";
    .

map:inv_proyecto_idi_titulo_proyecto a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_proyecto_idi;
    d2rq:property dc:title;
    d2rq:column "inv_proyecto_idi.titulo_proyecto";
    d2rq:classDefinitionLabel "Titulo";
    .

map:inv_proyecto_idi_fecha_ini a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_proyecto_idi;
    d2rq:property dbpedia-prop:startDate;
    d2rq:column "inv_proyecto_idi.fecha_ini";
    .

map:inv_proyecto_idi_fecha_fin a d2rq:PropertyBridge;

```



```

    d2rq:belongsToClassMap map:inv_proyecto_idi;
    d2rq:property dbpedia-prop:endDate;
    d2rq:column "inv_proyecto_idi.fecha_fin";
    .
map:inv_proyecto_idi_tipo_proyecto a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_proyecto_idi;
    d2rq:property dbpedia-prop:category;
    d2rq:column "inv_proyecto_idi.tipo_proyecto";
    .

# Table inv_proyecto_investigador
map:inv_proyecto_investigador1 a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:empleado;
    d2rq:property dbpedia-prop:projectCoordinator;
    d2rq:refersToClassMap map:inv_proyecto_idi;
    d2rq:condition "inv_proyecto_investigador.es_externo = 1";
    d2rq:join "inv_proyecto_investigador.referencia_proyecto =
inv_proyecto_idi.referencia_proyecto";
    d2rq:join "inv_proyecto_investigador.emp_id = empleado.emp_id";
    .

map:inv_proyecto_investigador2 a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:empleado;
    d2rq:property dbpedia-prop:projectCoordinator;
    d2rq:refersToClassMap map:inv_proyecto_idi;
    d2rq:condition "inv_proyecto_investigador.es_externo = 0";
    d2rq:join "inv_proyecto_investigador.referencia_proyecto =
inv_proyecto_idi.referencia_proyecto";
    d2rq:join "inv_proyecto_investigador.emp_id = empleado.emp_id";
    .

# Table inv_patente
map:inv_patente a d2rq:ClassMap;
    d2rq:dataStorage map:database;
    d2rq:uriPattern "inv_patente/@@inv_patente.registro_patente@";
    d2rq:class dbpedia-owl:ResearchProject;
    d2rq:classDefinitionLabel "Patente";
    .

map:inv_patente_titulo_patente a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_patente;
    d2rq:property dc:title;
    d2rq:column "inv_patente.titulo_patente";
    d2rq:classDefinitionLabel "Titulo";
    .

map:inv_patente_anyo_patente a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_patente;
    d2rq:property dc:date;
    d2rq:column "inv_patente.anyo_patente";
    .

map:inv_patente_entidad_patente a d2rq:PropertyBridge;
    d2rq:belongsToClassMap map:inv_patente;
    d2rq:property dbpedia-prop:organisation;
    d2rq:column "inv_patente.entidad_patente";
    .

# Table inv_patente_autor
map:inv_patente_autor1 a d2rq:PropertyBridge;

```

```
d2rq:belongsToClassMap map:empleado;
d2rq:property dbpedia-prop:registration;
d2rq:refersToClassMap map:inv_patente;
d2rq:condition "inv_patente_autor.es_externo = 1";
d2rq:join "inv_patente_autor.registro_patente =
inv_patente.registro_patente";
d2rq:join "inv_patente_autor.emp_id = empleado.emp_id";
.

map:inv_patente_autor2 a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:empleado;
d2rq:property dbpedia-prop:registration;
d2rq:refersToClassMap map:inv_patente;
d2rq:condition "inv_patente_autor.es_externo = 0";
d2rq:join "inv_patente_autor.registro_patente =
inv_patente.registro_patente";
d2rq:join "inv_patente_autor.emp_id = empleado.emp_id";
.
```

6. Referencias

1. **IBM.** Big Data at the Speed of Business. [Online] <http://www-01.ibm.com/software/data/bigdata/>.
2. **Torres i Viñals, J.** Del cloud computing al big data. [Online] UPC Barcelona. http://www.jorditorres.org/wp-content/uploads/2012/03/Del.Cloud_.Computing.al_.Big_.Data_.JordiTorres.ES_.pdf.
3. **W3C.** Semantic Web. [Online] 2013. http://www.w3.org/2001/sw/wiki/Main_Page.
4. **Conatel and Senatel.** Plan Nacional de Desarrollo de las Telecomunicaciones 2007 - 2012. [Online] 2007. http://www.comunidadandina.org/telec/Plan_telecomunicaciones_ecuador.pdf.
5. E-Government Survey 2012. [Online] United Nations Department of Economic and Social Affairs, 2012. <http://unpan1.un.org/intradoc/groups/public/documents/un/unpan048065.pdf>.
6. **Corojan, A. and Campos, E.** GOBIERNO ABIERTO: ALCANCE E IMPLICACIONES. [Online] FEPS Foundation for European Progressive Studies. http://www.fundacionideas.es/sites/default/files/pdf/DT-Gobierno_abierto-Pol.pdf.
7. Open Data Navarra. [Online] 2011. http://www.navarra.es/home_es/Open-Data/Que-es-Open-Data/.
8. Open Government Partnership. [Online] 2012. <http://www.opengovpartnership.org/about>.
9. **Velasco, J.J.** Open Data: compartir y reutilizar datos para aportar valor. *Think Big*. [Online] Telefonica, 2013. <http://blogthinkbig.com/open-data-aportar-valor/>.
10. **De la Fuente, C. and Álvarez E., M.** La Promesa del Gobierno Abierto. *Open linked data: la nueva frontera*. [Online] 2013. <http://www.lapromesadelgobiernoabierto.info/lpga.pdf>.
11. **Pereira, D.** Las 5 estrellas de los Datos Abiertos. [Online] Open Data for Development in Latin America and the Caribbean, 2012. <http://www.od4d.org/2012/10/08/las-5-estrellas-de-los-datos-abiertos/>.
12. Open Knowledge Foundation. [Online] <http://okfn.org/about/>.
13. **W3C.** W3C SEMANTIC WEB ACTIVITY. [Online] 2013. <http://www.w3.org/2001/sw/>.
14. **OMG.** OMG WE SET STANDARD. [Online] <http://www.omg.org/gettingstarted/gettingstartedindex.htm>.
15. Sunlight Foundation. [Online] <http://sunlightfoundation.com/about/>.
16. Escuela Politécnica Superior. [Online] <http://www.uam.es/ss/Satellite/EscuelaPolitecnica/es/la-escuela-2/Page/contenidoFinal/presentacion-3.html>.

17. **W3C**. Sobre W3C. [Online] <http://www.w3c.es/Consortio/>.
18. **e-Kontsulta**. Web Semántica. [Online] Gobierno Vasco. http://www.ekontsulta.net/ekontsulta/wiki/index.php/Web_Sem%C3%A1ntica.
19. Guía Breve de Web Semántica. [Online] W3C - España. <http://www.w3c.es/Divulgacion/GuiasBreves/WebSemantica>.
20. **Perissé, M.** Gestión del Conocimiento: estructuración de la información científica para la Web Semántica. [Online] UNIVERSIDAD NACIONAL DE LA MATANZA, 2009.
21. **Palacios, V.** Sistema de Recuperación Conceptual mediante Niveles Semánticos en la representación de Esquemas de Metadatos. [Online] Universidad Carlos III de Madrid, 2010. <http://e-archivo.uc3m.es/handle/10016/9332>.
22. **Castells, P.** La web semántica. [Online] Universidad Autónoma de Madrid, EPS. <http://arantxa.ii.uam.es/~castells/publications/castells-uclm03.pdf>.
23. **W3C**. Lenguaje de Ontologías Web (OWL) Vista General. [Online] 2004. <http://www.w3.org/2007/09/OWL-Overview-es.html>.
24. —. RDF Vocabulary Description Language 1.0: RDF Schema. [Online] <http://www.w3.org/TR/2002/WD-rdf-schema-20021112/>.
25. —. Web-Ontology (WebOnt) Working Group. [Online] <http://www.w3.org/2001/sw/WebOnt/>.
26. —. OWL Web Ontology Language Guide. [Online] <http://www.w3.org/TR/owl-guide/>.
27. —. OWL 2 Web Ontology Language Document Overview (Second Edition). [Online] 2012. <http://www.w3.org/TR/owl2-overview/#sec-syn>.
28. **Qaissi, H.** Lenguaje De Recuperación De Datos en la Web Semántica SPARQL. [Online] Universidad Politécnica de Madrid, 2009. <http://sinbad.dit.upm.es/docencia/doctorado/curso0809/HichamSPARQLrevCarmen.pdf>.
29. **W3C, España**. Guía Breve de Linked Data. [Online] <http://www.w3c.es/Divulgacion/GuiasBreves/LinkedData>.
30. **W3C**. Uniform Resource Identifier (URI): Generic Syntax. [Online] 2005. <http://www.ietf.org/rfc/rfc3986.txt>.
31. **HispaMedios**. URL y URI. [Online] <http://www.hispamedios.com/articles/id34-url-y-uri>.
32. **Lamarca L., M.** Hipertexto: El nuevo concepto de documento en la cultura de la imagen. [Online] 2011. <http://www.hipertexto.info/documentos/ontologias.htm>.

33. **W3C**. OWL Web Ontology Language Use Cases and Requirements. [Online] <http://www.w3.org/TR/webont-req/#onto-def>.
34. **DBpedia**. [Online] 2009. <http://wiki.dbpedia.org/About>.
35. —. DBpedia 3.8 released, including enlarged Ontology and additional localized Versions. [Online] 2012. <http://blog.dbpedia.org/2012/08/06/dbpedia-38-released-including-enlarged-ontology-and-additional-localized-versions/>.
36. —. Mapping Guide. [Online] 2011. http://mappings.dbpedia.org/index.php/Mapping_Guide.
37. —. DBpedia Mappings. [Online] 2012. http://mappings.dbpedia.org/index.php/Main_Page.
38. Freie Universität Berlin. [Online] <http://www.fu-berlin.de/>.
39. **Campaña G, Jesus**. Representación y Tratamiento Semántico de información Imprecisa en B.D. [Online] <http://hera.ugr.es/tesisugr/20081054.pdf>, 2011. <http://hera.ugr.es/tesisugr/20081054.pdf>.
40. D2RQ. [Online] Freie Universität Berlin. <http://d2rq.org/>.
41. kurtjx/SNORQL. [Online] GitHub, 2013. <https://github.com/kurtjx/SNORQL>.
42. DBpedia. [Online] team at a face-to-face meeting, 08 08, 2011. <http://wiki.dbpedia.org/UseCases?v=1rx>.
43. **Mazzo I., R**. Linked Open Data: qué es y ejemplos en el mundo. [Online] Biblioteca Nacional del Congreso de Chile, 2010. <http://www.bcn.cl/de-que-se-habla/open-data-link-data>.
44. **W3C**. Resource Description Framework (RDF): Concepts and Abstract Syntax. [Online] 2002. <http://www.w3.org/TR/2002/WD-rdf-concepts-20021108/>.
45. Escuela Politécnica Superior. [Online] <http://www.uam.es/ss/Satellite/EscuelaPolitecnica/es/la-escuela-2/Page/contenidoFinal/presentacion-3.htm>.
46. **CTIC**. Open Data . [Online] 2013. <http://datos.fundacionctic.org/sandbox/catalog/faceted/>.
47. **Berners-Lee, Tim**. Semantic Web - XML2000. [Online] W3C. <http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide11-0.html>.
48. **DBpedia**. Interlinking DBpedia with other Data Sets. [Online] 2013. <http://wiki.dbpedia.org/Interlinking>.
49. **Wikipedia**. Uniform resource identifier. [Online] 2013. http://en.wikipedia.org/wiki/Uniform_resource_identifier.
50. **DBpedia**. Ontology Classes. [Online] <http://mappings.dbpedia.org/server/ontology/classes/#owl:Thing>