



Uso de Wikis Semánticas

Juana María Perea Pérez
2o cicle de Ingeniería en Informática

Felipe Geva Urbano

Junio 2013



Esta obra está sujeta a una licencia de [Reconocimiento-NoComercial 3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc/3.0/es/)

FICHA DEL TRABAJO FINAL

| | |
|--|------------------------------|
| Título del trabajo: | Uso de Wikis Semánticas |
| Nombre del autor: | Juana María Perea Pérez |
| Nombre del consultor: | Felipe Geneva Urbano |
| Fecha de Entrega: | 06/2013 |
| Área del Trabajo Final: | XML y Web Semática |
| Titulación: | 2o ciclo de Ing. Informática |
| Resumen del Trabajo : | |
| <p>En este trabajo se ha realizado una tarea de investigación con la finalidad de construir una ontología para cargarla en una wiki semántica y así analizar su funcionamiento. Para ello se ha documentando:</p> <ul style="list-style-type: none"> - Los lenguajes que intervienen en la codificación como son el XML, RDL y OWL. - El software que se ha utilizado para implementar nuestra ontología. - El proceso que se ha realizado para el desarrollo de la ontología. - Que es una wiki semántica, se han comparado los dos motores OntoWiki y Semantic Mediawiki. - Se ha documentado el proceso de instalación de OntoWiki y su utilización (subida de la ontología, creación de consultas, modificación de etiquetas, etc). <p>El resultado final de todo este trabajo ha sido una ontología que se puede consultar desde OntoWiki desde la url http://ec2-46-137-55-124.eu-west-1.compute.amazonaws.com/ontowiki/proyecto/</p> | |
| Abstract: | |
| <p>This project has made a research task to build an ontology and upload it to a semantic wiki, and then analyze its performance.</p> <p>This project has made a research task to build an ontology and upload it to a semantic wiki, and then analyze its performance.</p> <p>Has been documented:</p> <ul style="list-style-type: none"> - The languages involved in encoding (XML, RDL, OWL, etc..). - The software has been used to implement our ontology. - The process that has been done for the development of the ontology. - What is a semantic wiki, we have compared the two engines OntoWiki and Semantic Mediawiki. - It has been documented OntoWiki installation process and use (rise of ontology query secretion, modification of labels, etc). <p>As a result of all this work has been an ontology that can be queried from OntoWiki from the url http://ec2-46-137-55-124.eu-west-1.compute.amazonaws.com/ontowiki/proyecto/</p> | |
| Palabras clave: | |
| Ontología, Wiki Semantica, OWL, XML, RDF, OntoWiki | |

Índice

| | | |
|---------------------------|------------------------------------|----|
| <u>1</u> | <u>Introducción</u> | 8 |
| 1.1 | <u>Justificación de TFC</u> | 8 |
| 1.2 | <u>Objetivos de TFC</u> | 8 |
| 1.3 | <u>Enfoque y método seguido</u> | 8 |
| 1.4 | <u>Planificación del proyecto</u> | 9 |
| 1.4.1 | <u>Tareas</u> | 9 |
| 1.4.2 | <u>Hitos</u> | 10 |
| 1.4.4 | <u>Planificación</u> | 10 |
| 1.4.5 | <u>Diagrama de Gantt</u> | 12 |
| 1.5 | <u>Productos Obtenidos</u> | 12 |
| 1.6 | <u>Estructura de la memoria</u> | 13 |
| <u>2</u> | <u>XML y XML-Schema</u> | 14 |
| 2.1 | <u>El lenguaje XML</u> | 14 |
| 2.1.1 | <u>Documentos XML</u> | 14 |
| 2.1.2 | <u>Reglas del documento XML</u> | 16 |
| 2.1.3 | <u>Parsers XML</u> | 16 |
| 2.2 | <u>DTD y XML-Schema</u> | 16 |
| 2.2.1 | <u>DTD</u> | 16 |
| 2.2.2 | <u>XML-Schema</u> | 17 |
| <u>3</u> | <u>RDF y RDF-Schema</u> | 18 |
| 3.1 | <u>RDF</u> | 18 |
| 3.1.1 | <u>URI</u> | 18 |
| 3.1.2 | <u>RDF/XML</u> | 18 |
| 3.2 | <u>RDF-Schema</u> | 19 |
| <u>4</u> | <u>SPARQL</u> | 21 |
| <u>5</u> | <u>OWL</u> | 23 |
| 5.1 | <u>Sublenguajes de OWL</u> | 23 |
| 5.2 | <u>Elección de un sublenguaje</u> | 23 |
| <u>6</u> | <u>Creación de una ontología</u> | 24 |
| 6.1 | <u>Que es una ontología</u> | 24 |
| 6.1.1 | <u>Como crear una ontología</u> | 24 |
| 6.2 | <u>WordNet</u> | 25 |
| 6.3 | <u>Protégé</u> | 25 |
| 6.4 | <u>Diseño ontología</u> | 26 |
| 6.4.1 | <u>Proceso general</u> | 26 |
| 6.4.2 | <u>Definición de la ontología</u> | 26 |
| Superclases | | 27 |
| Individuos de las clases: | | 29 |
| Propiedades de las clases | | 31 |
| 6.4.3 | <u>Diseño ontología en Protégé</u> | 32 |
| <u>7</u> | <u>Wiki Semántica</u> | 37 |
| 7.1 | <u>Introducción</u> | 37 |
| 7.2 | <u>Motores de Wiki semánticas</u> | 37 |
| 7.3 | <u>Instalación OntoWiki</u> | 38 |
| 7.4 | <u>Carga de una ontología</u> | 46 |
| 7.5 | <u>Uso OntoWiki</u> | 48 |

| | |
|---|----|
| 7.5.1 Selección de lenguajes | 48 |
| 7.5.2 Creación de consultas | 50 |
| 7.5.3 Etiquetas en diferentes idiomas | 54 |
| 7.6 Problemas Detectados | 57 |
| 8.Conclusiones | 60 |
| Glosario | 61 |
| Bibliografía | 63 |
| Webs Consultadas | 65 |
| Anexo I-Manual del usuario | |
| Anexo II - Código Ontología | |

Índice de Figuras

| | |
|---|----|
| Figura 1 Diagrama de Gantt | 12 |
| Figura 2 Cabecera XML | 14 |
| Figura 3 Elemento | 14 |
| Figura 4 Elemento Vacío | 14 |
| Figura 5 Elemento con Atributo | 14 |
| Figura 6 Ejemplo Documento XML | 15 |
| Figura 7 Ejemplo CDATA | 15 |
| Figura 8 Ejemplo Comentario | 15 |
| Figura 9 Ejemplo XML-Schema | 17 |
| Figura 10 Ejemplo XML-Schema namespaces | 17 |
| Figura 11 Ejemplo RDF/URI | 18 |
| Figura 12 Ejemplo documento RDF | 18 |
| Figura 13 Ejemplo documento RDF-Schema | 19 |
| Figura 14 Tabla Prefijos | 21 |
| Figura 15 Ejemplo Prefijos | 22 |
| Figura 16 Superclases Municipio | 27 |
| Figura 17 Superclases Playa | 27 |
| Figura 18 Superclases Localidad | 28 |
| Figura 19 Superclases Hotel | 28 |
| Figura 20 Clases y relaciones | 31 |
| Figura 21 Vista Classes | 32 |
| Figura 22 Vista Classes Propiedades | 33 |
| Figura 23 Vista Entities | 33 |
| Figura 24 Vista Individuals | 34 |
| Figura 25 Vista Object Properties | 34 |
| Figura 26 Consulta SPARQL Municipio | 35 |
| Figura 27 Consulta SPARQL Localidades | 35 |
| Figura 28 Ejemplo Código RDF/OWL | 36 |
| Figura 29 Instalación LOD2 | 39 |
| Figura 30 Instalación LOD2 | 40 |
| Figura 31 Instalación OntoWiki-Mysql | 41 |
| Figura 32 Contraseña Mysql | 41 |
| Figura 33 Confirmación contraseña Mysql | 42 |
| Figura 34 Usuario/Pass SuperAdmin | 42 |
| Figura 35 Acceso SuperAdmin | 42 |
| Figura 36 Acceso OntoWiki | 42 |
| Figura 37 OntoWiki System Configuration | 43 |
| Figura 38 OntoWiki User | 43 |
| Figura 39 OntoWiki configuración password Admin | 43 |
| Figura 40 OntoWiki Add Property | 43 |
| Figura 41 OntoWiki userPassword | 44 |
| Figura 42 OntoWiki Logout | 44 |
| Figura 43 Acceso Admin | 44 |
| Figura 44 OntoWiki Preferences | 45 |
| Figura 45 OntoWiki Cambio Password Admin | 45 |
| Figura 46 OntoWiki Admin Properties | 45 |

| | |
|---|----|
| Figura 47 OntoWiki Botón Edit | 46 |
| Figura 48 OntoWiki Botón Create Knowledge Base | 46 |
| Figura 49 OntoWiki Create Knowledge Base | 46 |
| Figura 50 OntoWiki Upload Ontowiki | 47 |
| Figura 51 OntoWiki Model Configuration | 47 |
| Figura 52 OntoWiki Ontología Cargada | 48 |
| Figura 53 OntoWiki Extras | 48 |
| Figura 54 OntoWiki Configurar Select Lenguaje | 48 |
| Figura 55 OntoWiki añadir lenguajes en Select Lenguajes | 49 |
| Figura 56 OntoWiki Configuración Select Lenguajes | 49 |
| Figura 57 Código RDF clase Construcción | 49 |
| Figura 58 OntoWiki Queries | 50 |
| Figura 59 OntoWiki Query Editor | 50 |
| Figura 60 OntoWiki Query Editor Consulta | 51 |
| Figura 61 OntoWiki Query Editor Submit Query | 51 |
| Figura 62 OntoWiki Query Editor Error Save | 52 |
| Figura 63 OntoWiki Query Editor Save Queries | 52 |
| Figura 64 OntoWiki Query | 52 |
| Figura 65 OntoWiki Query Instances | 53 |
| Figura 66 OntoWiki Query Properties | 53 |
| Figura 67 OntoWiki Edit Resource | 55 |
| Figura 68 OntoWiki Edit Resource añadir label | 55 |
| Figura 69 OntoWiki View Resource | 56 |
| Figura 70 OntoWiki Source | 56 |
| Figura 71 OntoWiki Error caracteres especiales | 57 |
| Figura 72 OntoWiki Error | 57 |
| Figura 73 OntoWiki Select Lenguaje Correcto | 58 |
| Figura 74 OntoWiki Saved Queries | 59 |

1 Introducción

En este capítulo se encuentran los conceptos generales del proyecto realizado, como es la justificación del trabajo, sus objetivos y el producto obtenido. También se incluyen las tareas y hitos identificados en la planificación del proyecto y para finalizar se proporciona una visión global de la estructura de la esta memoria.

1.1 Justificación de TFC

El campo de la Web Semántica es un campo relativamente nuevo que está teniendo un gran impacto tanto en la práctica como en la investigación. La filosofía de la Web Semántica es añadir conocimiento a la web. Si se consiguiera este objetivo búsquedas muy complejas se podrían realizar sin complicaciones. Por ejemplo: *Playa con bandera azul dentro del municipio de Calviá con hotel de 5 estrellas*.

Actualmente esta tecnología se aplica en Wikis Semánticas, que integran ontologías que permiten realizar las búsquedas complejas anteriormente comentadas. Para ello se requiere un trabajo de creación e implementación de la ontología que refleje el campo o materia que se ofrece al usuario. Estas Wikis Semánticas son herramientas altamente colaborativas, eso hace que los usuarios que consulten los datos puedan ampliarlos, borrarlos o modificarlos.

En este proyecto se ha desarrollado un primer esbozo de lo que podría llegar a ser una ontología turística aplicada en el contexto del Municipio de Calviá. La ontología desarrollada ofrecerá al usuario que la consulta la opción de realizar algunas consultas implementadas sobre ella y el colaborar en lo desarrollado para ampliarla.

1.2 Objetivos de TFC

El objetivo final de este trabajo es desarrollar una ontología y aplicarla mediante el uso de una wiki semántica (Semantic-Mediawiki). Profundizar en las características de esta wiki, su funcionalidad, como almacena información, como se añade información semántica, como importar ontologías, etc. Después de esto, desarrollar una ontología y desplegarla, validando la utilidad de la herramienta, sus puntos fuertes y sus carencias.

También se han de alcanzar unos objetivos generales en el desarrollo de este trabajo, son los siguientes:

- Estudiar los conceptos básicos de la web semántica.
- Entender que es una ontología.
- Conocer algunos de los lenguajes de representación de ontologías más utilizados (OWL, RDF...).
- Utilizar y conocer algunas de las herramientas de edición y gestión de ontologías mas conocidas.
- Aprender a utilizar ontologías para hacer aplicaciones semánticamente más ricas.
- Realización de casos prácticos de estas tecnologías, implementando una aplicación web sencilla.

1.3 Enfoque y método seguido

El proyecto siguiendo los objetivos del apartado anterior ha sido dividido en dos partes diferenciadas:

- Una primera parte que es una introducción a las técnicas y tecnologías que proporcionan una base para la construcción de ontologías (capítulos del 2 al 5). Se incluye un pequeño estudio de los lenguajes XML, XML-Schema, RDF, RDF- Schema ya que son la base del lenguaje OWL.
- La segunda parte se constituye de la creación de una ontología, describiendo la técnica y el software utilizado para ello. Por ese motivo, se ha realizado una basándose algunas zonas turísticas de Baleares, reflejando algunas características de estas, por ejemplo, número de hoteles y playas, tipos de hoteles (categoría, capacidad, servicios), descripción de las playas (longitud, si posee bandera azul, limpieza de aguas, etc). Todos estos datos son accesibles desde las páginas webs turísticas y guías existentes en Internet. Se ha creado la ontología necesaria para realizar el proyecto con el software PROTÉGÉ (capítulo 6)
- Para finalizar se estudia la estructura de las dos wikis semánticas sugeridas en el plan de estudios del TFC (Semantic MediaWiki y OntoWiki). Se realiza la importación de la ontología en OntoWiki, reflejando los pasos que se realizan, los problemas detectados y su usabilidad (capítulo 7).

1.4 Planificación del proyecto

En esta sección se exponen las tareas que se han realizado. El tiempo dedicado a estas tareas ha estado predefinido por las limitaciones temporales del TFC y por las fechas de entrega de las diferentes PAC's obligatorias, esta temporalización se refleja en una temporalización que se incluye en este apartado.

1.4.1 Tareas

Se han a realizado las siguientes actividades.

- 1 Realización PAC1 “Plan de Trabajo”.
 - 1.1 Búsqueda de información de los temas a desarrollar en el TFC.
 - 1.2 Recoger algunos ejemplos bibliográficos que se utilizaran en el trabajo.
 - 1.3 Redacción del plan de trabajo.
 - 1.4 Entrega PAC1.
- 2 Instalar el software necesario para el desarrollo del proyecto. En este caso se instalará el PROTÉGÉ. Queda pendiente elegir que wiki semántica se va a utilizar.
- 3 Realización PAC2
 - 3.1 Realización capítulos 2, 3, 4, 5 y 6 del TFC.
 - 3.1.1 Leer y redactar el apartado referente XML y XML-Schema
 - 3.1.2 Leer y redactar el apartado referente RDF y RDF-Schema
 - 3.1.3 Leer y redactar el apartado referente a SPARQL
 - 3.1.4 Leer y redactar el apartado referente a OWL
 - 3.1.5 Creación de una ontología
 - 3.1.5.1 Que es una Ontología

- 3.1.5.2 WordNet
- 3.1.5.3 PROTÉGÉ
- 3.1.5.4 Diseño de la ontología propuesta
- 3.2 Entrega borrador PAC2
- 3.3 Corrección de errores detectados PAC2
- 3.4 Entrega PAC2
- 4 Realización PAC3
 - 4.1 Realización capítulo 7 del TFC
 - 4.1.1 Introducción (Definición wiki semántica)
 - 4.1.2 Motores de Wiki Semánticas
 - 4.1.3 Instalación y uso de OntoWiki
 - 4.1.4 Integración de la ontología
 - 4.2 Entrega de borrador PAC3
 - 4.3 Corrección de errores PAC3
 - 4.4 Entrega PAC3
- 5 Redacción de conclusiones, manual del usuario, glosario y bibliografía
- 6 Revisión del trabajo realizado y corrección de errores.
- 7 Resumir la información. Realizar presentación del proyecto con el resumen realizado.
- 8 Presentación de la memoria y la presentación.
- 9 Debate virtual.

1.4.2 Hitos

Durante el desarrollo de este trabajo ha existido una serie de entregas parciales con fechas predefinidas. A estas fechas se añaden las fechas de los borradores de las entregas parciales que se ha intentado realizar una semana antes de la fecha de entrega oficial. Así las fechas importantes han quedado como sigue:

| Hito | Fecha |
|----------------|------------------------|
| Entrega Pac1 | 11/03/13 |
| Borrador Pac2 | 08/04/13 |
| Entrega Pac2 | 15/04/13 |
| Borrador Pac3 | 14/05/13 |
| Entrega Pac3 | 21/05/13 |
| Entrega Final | 11/06/13 |
| Debate virtual | 25 al 28 de Junio 2013 |

1.4.4 Planificación

Aquí se refleja la planificación por tareas.

| Tarea | Fechas | Horas |
|--|----------------------|------------------|
| 1 Realización PAC1 “Plan de Trabajo”. | 05/03 → 11/03 | 23 horas |
| 1.1 Búsqueda de información de los temas a desarrollar en el TFC. | 05/03/13 | 3 horas |
| 1.2 Recoger algunos ejemplos bibliográficos que se utilizaran en el trabajo. | 06/03/13 | 3 horas |
| 1.3 Redacción del plan de trabajo. | 07/03 → 11/03 | 14 horas |
| 1.4 Entrega PAC1. | 11/03/13 | 0 |
| 2 Instalar el software necesario para el desarrollo del proyecto. | 12/03/13 | 3 horas |
| 3 Realización PAC2 | 13/03 → 15/04 | 120 horas |
| 3.1 Realización de los capítulos 2, 3, 4, 5 y 6 del TFC. | 13/03 → 28/03 | |
| 3.1.1 Leer y redactar el apartado referente XML y XML-Schema | 13/03 → 16/03 | 13 horas |
| 3.1.2 Leer y redactar el apartado referente RDF y RDF-Schema | 17/03 → 20/03 | 13 horas |
| 3.1.3 Leer y redactar el apartado referente a SPARQL | 21/03 → 22/03 | 6 horas |
| 3.1.4 Leer y redactar el apartado referente a OWL | 23/03 → 24/03 | 8 hora |
| 3.1.5 Creación de una ontología | 25/03 → 08/04 | |
| 3.1.5.1 Que es una Ontología | 25/03 → 26/03 | 6 horas |
| 3.1.5.2 WordNet | 27/03 → 28/03 | 7 horas |
| 3.1.5.3 PROTÉGÉ | 29/03 → 30/03 | 8 horas |
| 3.1.4 Diseño de la ontología propuesta | 31/03 → 08/04 | 31 horas |
| 3.3 Entrega borrador PAC2 | 08/04/13 | |
| 3.4 Corrección de errores detectados PAC2 | 11/04 → 15/04 | 20 horas |
| 3.5 Entrega PAC2 | 15/04/13 | |
| 4 Realización PAC3 | 16/04 → 21/05 | 113 horas |
| 4.1 Realización capítulo 7 del TFC | | 96 horas |
| 4.1.1 Wiki Semántica | 16/04 → 21/04 | 20 horas |
| 4.1.2 Instalación y uso de la wiki semántica | 22/04 → 26/04 | 15 horas |
| 4.1.3 Integración de la ontología | 27/04 → 14/05 | 61 horas |
| 4.2 Entrega de borrador PAC3 | 14/05/13 | |

| | | |
|--|---------------|----------|
| 4.3 Corrección de errores PAC3 | 17/05 → 21/05 | 17 horas |
| 4.4 Entrega PAC3 | 21/05/13 | |
| 5 Redacción de conclusiones, glosario, manual del usuario y bibliografía | 22/05 → 25/05 | 13 horas |
| 6 Revisión del trabajo realizado y corrección de errores. | 26/05 → 28/05 | 10 horas |
| 7 Resumir la información. Realizar presentación del proyecto con el resumen realizado. | 29/05 → 09/06 | 40 horas |
| 8 Presentación de la memoria y la presentación. | 11/06/13 | |
| 9 Debate virtual | 25/06 → 28/06 | |

1.4.5 Diagrama de Gannt

El diagrama de Gannt generado con la planificación del apartado anterior.

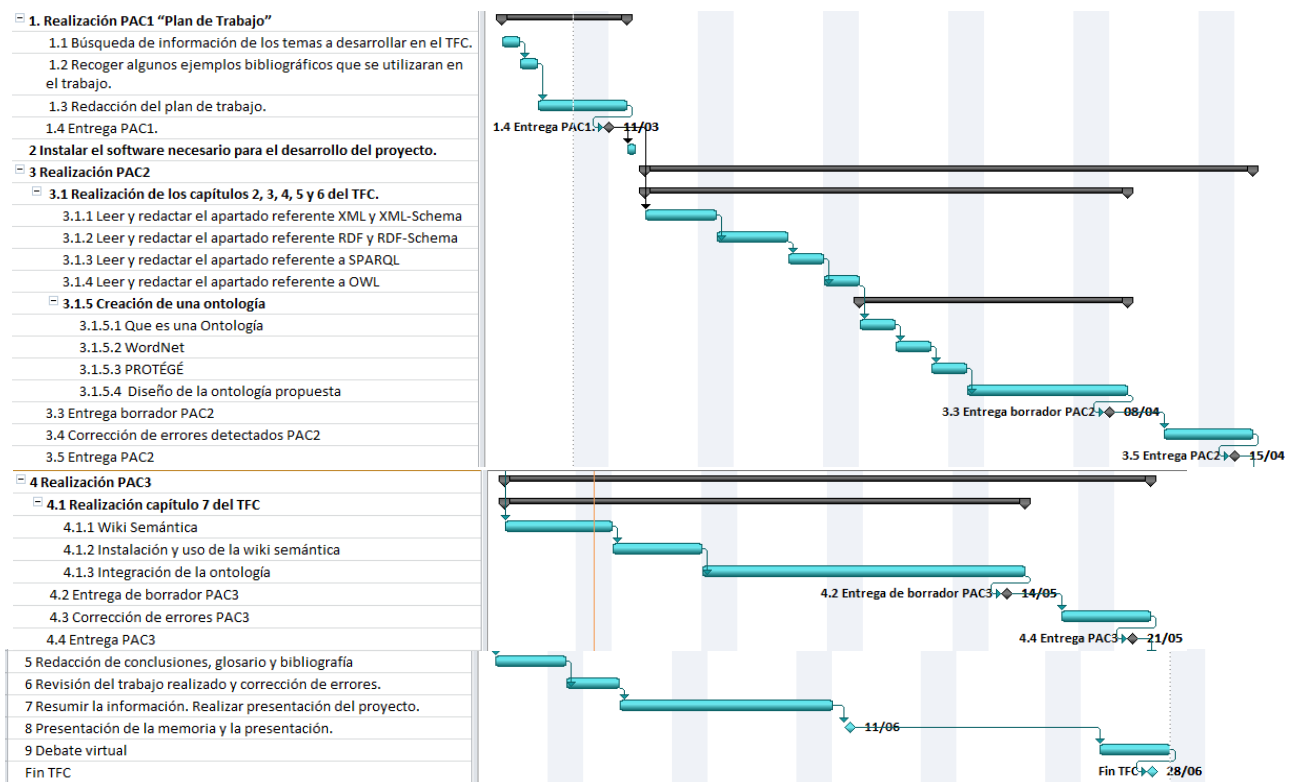


Figura 1 Diagrama de Gannt

1.5 Productos Obtenidos

Al termino del proyecto los productos obtenidos son los siguientes:

- Memoria del proyecto: Es el presente documento que se entregará en formato PDF.
- Presentación del Proyecto: Se trata de una presentación digital en formato vídeo para su exposición. Contendrá tanto diapositivas como la grabación de uso de web's con la explicación y su formato de entrega será en un formato de vídeo compatible con todos los

sistemas.

- Productos del caso práctico: El producto final es la ontología subida a la OntoWiki accesible desde la URL <http://ec2-46-137-55-124.eu-west-1.compute.amazonaws.com/ontowiki/proyecto/>

También se incluye el código rdf que codifica la ontología creada así como un manual para el usuario externo que explica como utilizar la OntoWiki.

1.6 Estructura de la memoria

La memoria se ha organizado con el objetivo de proporcionar una aproximación gradual al tema central del TFC: que es desarrollar una ontología y aplicarla mediante el uso de una wiki semántica (Semantic-mediawiki)

En los primeros capítulos se realiza una introducción a los diferentes lenguajes que contribuyen directa o indirectamente en la construcción de una ontología.

Así en el capítulo 2 se realiza una descripción de los lenguajes XML y de XML-Schema, en el capítulo 3 los lenguajes de RDF y RDF-Schema, en el capítulo 4 del lenguaje de consulta SPARQL y para finalizar en el capítulo 5 se trata el lenguaje OWL.

En el capítulo 6 se describe como se ha desarrollado la ontología., explicando el uso del WordNet y Protégé y para que se han utilizado en este caso,

En el siguiente, capítulo 7, se realiza un pequeño estudio sobre las wikis semánticas, comparando dos de ellas (OntoWiki y Semantic MediaWiki). También se describe como se ha instalado OntoWiki, como se utiliza (subida y modificación de una ontología) y los problemas que se han tenido en su utilización

Para finalizar en el capítulo 8 presentan las conclusiones del proyecto.

2 XML y XML-Schema

En este capítulo se verá una descripción general de los lenguajes XML y XML-Schema. En el primer apartado se hablará del lenguaje XML y sus características, en los siguientes se tratará el DTD y la evolución en el XML-Schema. También se mostrarán algunas de sus aplicaciones y se explicarán las ventajas e inconvenientes de usar estos lenguajes.

2.1 El lenguaje XML

El XML (Extensible Markup Language) [1] es un metalenguaje, es decir, permite describir otros lenguajes. Esto quiere decir que es un formato estándar para la estructuración de datos y no un lenguaje de programación propiamente dicho.

Este lenguaje es un derivado del estándar SGML (Standard Generalized Markup Language) [2] y su última definición [3] es la dada por W3C (World Wide Web Consortium) [4] .

La característica principal de este lenguaje es la utilización de etiquetas (tag's) y la posibilidad de establecer relaciones estructurales entre ellas. Esto permite que los usuarios de este lenguaje puedan crear sus propias etiquetas y definir el intercambio y validación de información entre sistemas.

2.1.1 Documentos XML

Los elementos básicos de los documentos XML son los *elementos* y las *etiquetas*. Los elementos definen el contenido del documento, y las etiquetas describen y estructuran a los elementos. En resumen un documento XML es una estructura jerarquía de etiquetas que organiza los elementos.

Los documentos XML se componen de las siguientes partes:

- **Prologo:** elemento no obligatorio, que contiene una declaración XML, donde se declara el documento como documento XML, y se pueden incluir comentarios e instrucciones de procesamiento. El formato habitual incluye:
 - *Version:* Indica la versión XML del documento.
 - *Encoding:* Indica la codificación que se va a utilizar para interpretar los caracteres del documento XML.

```
<?xml version="1.0" encoding="UTF-8"?>
```

Figura 2 Cabecera XML

- **Cuerpo:** parte obligatoria del documento. Para que el cuerpo esté bien formado unicamente ha de contener un elemento raíz, a partir de la cual se estructuran los demás elementos.
- **Elementos:** Un elemento viene definido por dos etiquetas una de inicio y otra de fin si este posee contenido, caracteres u otros elementos.

```
<elemento_1>contenido</elemento_1>
```

Figura 3 Elemento

O por una sola etiqueta si su contenido está vacío.

```
<elemento_2/>
```

Figura 4 Elemento Vacío

- **Atributos:** contenido opcional de los elementos que les incorpora de características o de propiedades identificativas. Estos atributos irán entre comillas.

```
<elemento_1 atributo_1="valor_atributo_1" >
```

Figura 5 Elemento con Atributo

Este ejemplo refleja todo lo explicado anteriormente y se ve su aplicación practica.

```
<?xml version="1.0" encoding="UTF-8"?>
<libro>
  <autor>George R. R. Martin </autor>
  <titulo edición="primera">Juego de Tronos</titulo>
  <coleccion>
    <volumen>1</volumen>
  </coleccion>
</libro>
```

Figura 6 Ejemplo Documento XML

- **Entidades predefinidas:** son marcas XML utilizadas para representar caracteres especiales para que no sean interpretados al procesar el documento XML. En esta tabla se reflejan las referencias a estas entidades.

| Referencia de entidad | Carácter |
|-----------------------|----------|
| & | & |
| < | < |
| > | > |
| ' | ' |
| " | " |

- **Secciones CDATA:** estas secciones indican al procesador de XML que no interprete el texto que contenga. Estas secciones comienzan con los caracteres <!CDATA [y se termina con]>. A continuación se ve un ejemplo:

```
<!CDATA[
  <HTML>
    <!DOCTYPE HTML PUBLIC "-//W3c/ DTD HTML 3.2 Final / EN">
    <HEAD>
      <TITLE>Ejemplo</TITLE>
    </HEAD>
    <BODY>
      <p> Ejemplo </p>
    </BODY>
  </HTML>
]]>
```

Figura 7 Ejemplo CDATA

- **Comentarios:** permiten añadir notas o información por parte del programador, que no serán mostrados por los navegadores ni procesados por una aplicación XML. Los comentarios se incluyen entre las cadenas de caracteres <!-- y -->, tal y como se ve a continuación:

```
<!-- Comentario TFC -->
```

Figura 8 Ejemplo Comentario

2.1.2 Reglas del documento XML

Para que un documento XML sea sintácticamente correcto ha de cumplir una serie de normas básicas:

- Un documento XML contiene uno o más elementos.
- El elemento raíz no ha de aparecer en el contenido de ningún otro elemento.
- La etiqueta de inicio y fin de un elemento ha de estar dentro del mismo elemento que lo contiene.
- El nombre de las etiquetas ha de seguir las siguientes convenciones:
 - El nombre ha de comenzar por una letra o por el carácter `_`. Las cifras y los signos de puntuación únicamente se pueden utilizar a partir del segundo carácter.
 - El nombre no puede contener espacios en blanco.
 - El nombre no puede contener el carácter `:`.
 - El nombre no puede comenzar por la cadena `xml`.
 - El nombre se ha de especificar justo después del carácter `<` que marca el inicio de la etiqueta.
 - El nombre que aparece la etiqueta de inicio al ser el mismo nombre que aparece en la etiqueta final.
- XML es sensible a la utilización de mayúsculas y minúsculas.
- Los atributos no pueden contener espacios en blanco, son declarados dentro de la etiqueta inicio del elemento y no pueden aparecer más de una vez dentro del mismo elemento.
- Un elemento puede poseer diversos atributos.
- Un documento XML está compuesto por un prologo y la instancia del documento, donde el prologo es opcional y solo puede aparecer al principio del documento.

2.1.3 Parsers XML

Un parser XML es el software que procesa los documentos XML. Esa herramienta es la que se encarga de leer, crear o modificar un documento XML, por tanto cualquier aplicación que trate documentos XML ha de contener un parser. Este procesador también se encarga de comprobar que el documento está bien formado y si es válido.

2.2 DTD y XML-Schema

Los documentos XML han de ser válidos además de estar bien formados. Para que estos documentos sean válidos han de seguir una estructura definida por el tipo de documento DTD o por un XML-Schema. En los apartados siguientes se verá una descripción general de estos dos tipos de definición.

2.2.1 DTD

Las DTD definen que elementos se pueden incluir dentro de un documento XML, sus relaciones y

los atributos que se les puede asignar. Para incluir una DTD se ha de especificar en una línea al inicio del documento XML, con el formato siguiente:

```
<!DOCTYPE tfc SYSTEM "tfc.dtd">
```

donde tfc.dtd es una URL o un directorio donde se encuentra la definición de la DTD. La definición de la DTD es utilizada por el documento XML para comprobar que la gramática utilizada es correcta y por tanto un documento XML válido.

Si se desea ampliar los conocimientos sobre los documentos DTD consultar [3] ya que en este trabajo no se va a tratar con más profundidad al no ser el objetivo del proyecto.

2.2.2 XML-Schema

EL XML-Schema [5, 6] esta substituyendo al DTD y es la nueva solución recomendada por W3C para la definición de tipos de documento y de nuevos tipos de datos . Este nueva especificación de documentos XML se diseñó para solucionar los problemas que tenía el DTD, utiliza lenguaje propio diferente al XML y limitaciones en la especificación de los tipos de datos.

A continuación se verán alguna de las características del XML-Schema:

- Utiliza sintaxis XML para la definición de esquemas.
- Define los elementos y atributos que pueden aparecer en el documento.
- Define cuales son los elementos hijos, el orden y el número de estos.
- Define si un elemento puede estar o no vacío, el tipo de datos que contiene tanto él como sus atributos.
- Permite utilizar conjuntos de elementos.
- Permite definir elementos con el mismo nombre y diferente contenido.
- Permite utilizar namespaces.

Un ejemplo de estructura de estructura xsd que contiene las características anteriores es el siguiente:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="libro">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Titulo" type="xsd:string"/>
        <xsd:element name="Autor" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

Figura 9 Ejemplo XML-Schema

Como ya se ha comentado, el XML-Schema permite definir namespaces (espacios de nombres) dentro de un xml. Esto significa que se pueden definir etiquetas propias como se ve a continuación.

```
<?xml version="1.0" encoding="UTF-8" ?>
<bl:biblioteca xmlns:xsd="http://www.biblioteca.org/biblioteca">
  <bl:libro id="Juego de Tronos">
    <bl:autor>George R. R. Martin </bl:autor/>
  </bl:libro/>
</bl:biblioteca>
```

Figura 10 Ejemplo XML-Schema namespaces

3 RDF y RDF-Schema

En este capítulo es una introducción a los lenguajes RDF y RDF-Schema. En el primer punto se hablará del lenguaje RDF y sus características, en el siguiente se tratará el RDF-Schema para terminar con las aplicaciones de estos lenguajes.

3.1 RDF

El RDF (Resource Description Framework) define un modelo de datos que permite la integración ágil de diferentes fuentes de información utilizando conceptos semánticos. Este lenguaje utiliza el XML para la descripción de recursos de la World Wide Web, y en este apartado se describirá sus características principales [7,8,9].

3.1.1 URI

Una URI [10] es una cadena de caracteres que sirve para identificar un recurso de manera independiente del contexto y si no es accesible o no desde Internet. El RDF utiliza las URI's para identificar los recursos y describirlos en forma de propiedades sencillas y valores de propiedades. Esto permite representar las declaraciones de los recursos como un grafo dirigido de nodos y arcos que representan los recursos, sus propiedades y valores. Aquí se ve un ejemplo que ilustra lo anterior:

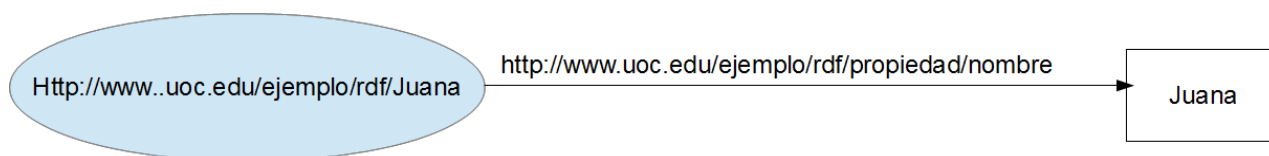


Figura 11 Ejemplo RDF/URI

3.1.2 RDF/XML

El formato RDF/XML [11] permite representar los grafos RDF como un documento XML que se denominará documento RDF.

Un documento RDF básicamente se compone de un elemento `rdf:RDF` que contiene un conjunto de descripciones, como se puede ver a continuación.

```
<?xml version="1.0"?>
  <rdf:RDF xmlns:rdf="http://www.w3org/1999/02/22-rdf-syntax-ns#" (1)
    xmlns:biblioteca="http://www.uoc.edu/rdf/biblioteca/"> (2)
    <rdf:Description rdf:about="http://www.uoc.edu/rdf/biblioteca/Juego de Tronos"
      propietats:titulo="Juego de Tronos"
      propietats:autor="George R. R. Martin" />
    <rdf:type rdf:resource="http://www.uoc.edu/rdf/biblioteca#libro"/>
  </rdf:Description>
</rdf:RDF>
```

Figura 12 Ejemplo documento RDF

Como ya se ha comentado RDF utiliza URI para nombrar recursos y propiedades, en el ejemplo anterior se utilizan dos (1) y (2) que permiten abreviar las referencias a las propiedades particulares de RDF y las que hemos definido nosotros.

El elemento `rdf:Description` es utilizado para describir un recurso y utiliza el atributo `rdf:about` para

especificar el recurso que se está describiendo.

Las propiedades de los recursos se define con *rdf:resource* y para indicar su tipo se utiliza el elemento *rdf:type*.

La mejora del documento RDF respecto al XML es su mayor capacidad para expresar significado semántico, aunque el XML sea el ideal para compartir información y realizar intercambio de datos.

3.2 RDF-Schema

El RDF-Schema es una recomendación de W3C [12] descrita por la siguiente sentencia:

“RDF Schema es un lenguaje de representación de conocimiento extensible que podemos utilizar para crear un vocabulario para describir clases, subclases, y propiedades de los recursos RDF”

El RDF-Schema se codifica como el RDF en XML y es considerado como una ampliación semántica de RDF.

Las primitivas básicas para describir esquemas RDF-Schema son:

- Definición de clase (*rdfs:Class*) y subclase (*rdfs:subClassOf*), que permite la definición de la jerarquía de clases.

```
<?xml version="1.0"?>
  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:biblioteca="http://www.w3.org/200/01/rdf-schema#">
    <rdf:Description rdf:ID="Libro">
      <rdf:type rdf:resource="http://www.w3.org/200/01/rdf-schema#Class"/>
      <rdfs:subClassOf rdf:resource="http://www.w3.org/200/01/rdf-schema#Resource" />
    </rdf:Description>
    <rdf:Description rdf:ID="Literatura">
      <rdf:type rdf:resource="http://www.w3.org/200/01/rdf-schema#Class"/>
      <rdfs:subClassOf rdf:resource="http://www.w3.org/200/01/rdf-schema#Libro"/>
    </rdf:Description>
    <rdf:Description rdf:ID="Literatura norteamericana">
      <rdf:type rdf:resource="http://www.w3.org/200/01/rdf-schema#Class"/>
      <rdfs:subClassOf rdf:resource="http://www.w3.org/200/01/rdf-
schema#Literatura"/>
    </rdf:Description>
  </rdf:RDF>
```

Figura 13 Ejemplo documento RDF-Schema

Como se puede ver en el ejemplo anterior, las clases se representan con la etiqueta *<rdf:Description rdf:ID="...">*, especificando el tipo de recurso (*rdf:type*) que en este caso sería *Class*. Para indicar subclase se utiliza la etiqueta *rdfs:subClassOf*. En el ejemplo anterior se observa que *Literatura Norteamericana* es subclase de *Literatura* que a su vez es subclase de *Libro*.

- Especificación de las propiedades (*rdf:Property*) y subpropiedades (*rdfs:subPropertyOf*) que permiten jerarquizar las propiedades de la misma forma que las clases.
- Establecer las combinaciones entre las clases y propiedades utilizando las etiquetas de dominio (*rdfs:domini*) y rango (*rdfs:range*).

- Declarar instancias de clases mediante la declaraciones de tipo (rdf:type) tal y como se ve en el ejemplo anterior.

Con las primitivas aquí descritas se pueden desarrollar esquemas para dominios específicos.

4 SPARQL

El lenguaje SPARQL [13] es uno de los lenguajes de consulta más usado para RDF. El lenguaje SPARQL incluye IRIs, un subconjunto de Referencias URI RDF que omiten los espacios.

La mayoría de las formas de consulta en SPARQL contienen un conjunto de patrones de tripleta (triple patterns) denominadas *patrón de grafo básico*, son similares a las tripletas RDF, excepto que cada sujeto, predicado y objeto puede ser una variable.

Una consulta simple por ejemplo tendrá la siguiente estructura:

- sujeto: <http://example.org/book/book1>
- predicado: <http://purl.org/dc/elements/1.1/title>
- objeto: "SPARQL Tutorial"

Consulta:

```
SELECT ?title
WHERE
{
  <http://example.org/book/book1>
    <http://purl.org/dc/elements/1.1/title> ?title .
}
```

Resultado:

| Title |
|-------------------|
| "SPARQL Tutorial" |

Como se puede observar se obtiene una lista de soluciones ordenadas que puede contener varios, uno o ningún resultado.

La consulta anterior refleja el uso de la terna descrita de sujeto, predicado y objeto. Para este tipo de consulta se han de definir una serie de prefijos (Prefix) que relacionan un nombre a un espacio de nombres. Como se ve a continuación:

| Prefijo | IRI |
|---------|---|
| rdf: | http://www.w3.org/1999/02/22-rdf-syntax-ns# |
| rdfs: | http://www.w3.org/2000/01/rdf-schema# |
| xsd: | http://www.w3.org/2001/XMLSchema# |
| fn: | http://www.w3.org/2005/xpath-functions# |

Figura 14 Tabla Prefijos

Un ejemplo de una consulta que contiene una referencia a un espacio de nombres es la siguiente:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
```

```
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX pr: <http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#>
SELECT ?subject
      WHERE { ?subject pr:esLocalidad pr:Calvià}
```

Figura 15 Ejemplo Prefijos

Como se puede observar en esta consulta se devolverán las localidades de Calvià que estarán definidas dentro del espacio de nombres **pr**.

5 OWL

OWL son las siglas de Ontology Web Language [14, 15] y fue aprobado el 10 de Febrero por el consorcio World Wide Web para ser utilizado en el desarrollo de webs semánticas. Este lenguaje proporciona una manera común para procesar la información en la web. Ha sido diseñado para ser interpretado por aplicaciones informáticas y puede ser utilizado para representar explícitamente el significado de términos en vocabularios y la relación de estos términos. Esta representación es lo que se llama Ontología.

OWL posee mayor capacidad para reflejar significado y semántica de XML, RDF y RDF-Schema, lo que significa que puede representar contenido interpretable por un ordenador.

5.1 Sublenguajes de OWL

OWL se divide en tres sublenguajes OWL-Lite, OWL-DL y OWL-Full, siendo el más sencillo OWL-Lite y el más completo OWL-Full.

- OWL Lite: diseñado para usuarios con requerimientos básicos, como son clasificaciones jerárquicas y restricciones simples.
- OWL DL (Description Logics): este lenguaje da mayor grado de complejidad y expresividad compatible con los requisitos computacionales. Incluye todas las posibilidades del lenguaje OWL pero con restricciones.
- OWL Full: dirigido a usuarios que requieren la máxima expresividad y libertad sintáctica que ofrece RDF sin garantía computacional.

Como se puede observar cada lenguaje es un extensión del anterior y lo expresado en uno es válido en su sucesor pero no a la inversa. Es decir:

- Una ontología legal en OWL Lite es legal OWL DL.
- Una ontología legal en OWL DL es legal OWL Full.

5.2 Elección de un sublenguaje

En nuestro caso la elección del lenguaje que se utilizará para desarrollar la ontología lo determinará el nivel de complejidad que acepte la wiki semántica que se va a utilizar, Ontowiki. Pero en este apartado se dará una ligera pincelada de como elegir un lenguaje o otro.

La elección entre OWL Lite y OWL DL está supeditada a la expresividad que se quiere dar a las construcciones, siendo OWL DL el que proporciona una mayor expresividad.

La elección entre OWL DL y OWL Full depende de la necesidad de los recursos de modelado del esquema RDF.

OWL Full se puede considerar como una extensión de RDF, y OWL Lite y OWL DL como extensiones de un RDF restringido. Por este motivo un documento OWL siempre será un documento RDF, un documento RDF será OWL Full pero no un OWL Lite o DL.

Teniendo en cuenta el motivo anterior se tendrá especial cuidado a la hora de migrar documentos RDF a OWL, ya que han de cumplir las especificaciones del sublenguaje OWL elegido.

6 Creación de una ontología

En este apartado se verá las diferentes herramientas que se van a usar para el desarrollo de la ontología de este trabajo.

En el primer apartado se explicará que es una ontología y los pasos más comunes para generar una. En el siguiente se hablará del WordNet que se utilizará como ayuda a la hora de desarrollar la ontología. Y para finalizar se describirá el software Protégé que se utilizará para implementar la ontología.

6.1 Que es una ontología

Una ontología [16] informática hace referencia a la formulación de un exhaustivo y riguroso esquema conceptual dentro de uno o varios dominios dados; con la finalidad de facilitar la comunicación y el intercambio de información entre diferentes sistemas y entidades. Se aplica a un dominio específico, y describe y representa un área concreta de conocimiento.

Las ontologías contienen conceptos y las relaciones entre estos. Estas relaciones se reflejan de forma jerárquica, donde los conceptos más generales están situados en la zona más alta (superclases) y los más específicos en la zona más baja (subclases).

También están incluidas dentro de una ontología las propiedades, estas describen características y atributos de los conceptos y pueden ser utilizadas para relacionar clases.

6.1.1 Como crear una ontología

Para desarrollar una ontología uno de los métodos más populares es el creado por Noy y McGuinness [17]. Los pasos a seguir son los siguientes:

1. Definir el dominio y el alcance de la ontología.

Para ello se han de responder las siguientes preguntas:

- ¿Que dominio está relacionado con la ontología?
 - El propósito de la ontología
 - A que preguntas ha de responder la ontología.
 - Quien va a ser el usuario de la ontología.
2. Se ha de intentar reutilizar la ontologías ya existentes. Esta opción ha tenerse en cuenta antes de acometer un desarrollo, para no realizar desarrollos ya existentes o si nuestra ontología se va a relacionar con otras ya existentes.
 3. Enumerar los términos importantes en la ontología. Esto facilita la definición de clases y sus jerarquía.
 4. Definición de las clases y sus jerarquías. Para ello se proponen tres formas:
 - Top-down, consiste en definir las clases más generales y a partir de estas las especializadas.
 - Botton-up, en esta caso se realiza el proceso contrario al punto anterior. Se comienzan por las clases más específicas y se termina por las generales.

- Combinar los dos procesos anteriores, utilizando uno u otro a medida que se define la ontología.
5. Definir las propiedades de las clases, estas pueden ser del tipo:
 - Intrínsecas, son las inherentes de una clase.
 - Extrínsecas, son externas de una clase.
 - Partes, elementos que son partes del objeto que representa la clase.
 - Relaciones con otras clases.
 6. Añadir restricciones a las propiedades. Esto se refiere a los valores que puede admitir, su cardinalidad, su dominio, su rango, etc.
 7. Crear las instancias. Es un paso opcional ya que dentro del documento de la definición de la ontología no es necesario tenerlas incluidas.

6.2 WordNet

WordNet [18] es una base de datos léxica en Inglés que ha sido desarrollada por Cognitive Science Laboratory de la universidad de Princeton bajo la dirección del profesor George A. Miller. En esta base de datos los sustantivos, verbos, adjetivos y adverbios se agrupan en conjuntos de sinónimos cognitivos (synsets), cada uno expresando un concepto distinto. Synsets están vinculados entre sí por medio de relaciones conceptuales, semánticos y léxicos. Por este motivo esta estructura de datos que ofrece WordNet hace que sea una herramienta útil para la lingüística computacional y procesamiento del lenguaje natural.

El *WordNet* ofrece tanto un software para instalarlo en el ordenador como una consulta *on line* de la base de datos. Esto la convierte en una herramienta muy útil que ha sido punto de partida para otros proyectos relacionados con ontologías, como son:

- **Euro WordNet [19]**: es una base de datos multilingüe con wordnets para varios idiomas europeos (holandés, italiano, español, alemán, francés, checo y estonio).
- **Open Multilingual Wordnet [20]**: Proyecto similar al anterior que incluye idiomas como el Malayo/Indonesio, Danés, Gallego, Catalán.
- **SUMO (Suggested Upper Merged Ontology) [21]**: pretende ser una base ontológica para sistemas informáticos de procesamiento de información. Originalmente fue desarrollado por la Corporación Teknowledge y ahora es mantenido por Software Articulate. Es un candidato para la " ontología estándar superior "que IEEE 1600.1.

6.3 Protégé

Protégé 4.2.0 [22] es un software utilizado para desarrollar sistemas basados en el conocimiento. Esta herramienta permite:

- Diseñar una ontología de clases que describen un dominio.
- Crear herramientas de adquisición del conocimiento por acumulación de este.
- Introducir instancias específicas dentro de un dominio y crear una base del conocimiento.
- Ejecutar las aplicaciones.

Esta aplicación permite introducir de manera sistemática y fácil los datos necesarios para desarrollar y el tratamiento de ontologías. También permite la exportación e importación de documentos RDF, RDF Schema y OWL.

Hay que tener en cuenta ciertas algunas restricciones al usar Protégé [23] y crear documentos RDF.

- **Miembros multiclases:** RDF permite que un recurso pueda ser instanciado por más de una clase, Protégé no permite esta característica.
- **Slots:** las propiedades en RDF se llaman slots dentro del Protégé.
- **Rangs:** en RDF el rango de una propiedad ha de ser una clase, en cambio en Protégé permite especificar mas de una clase para un rango determinado.

En el desarrollo practico de este trabajo se va realizar con la última versión del Protégé 4.2.0 donde se implementará la ontología que se publicara en le wiki semántica.

6.4 Diseño ontología

En este apartado se describirá el proceso que se va a seguir para definir nuestra propia ontología. Para ello se utilizará como apoyo el WordNet on line para ampliar la semántica que se va a utilizar y el Protégé para implementar la ontología.

6.4.1 Proceso general

Como ya se ha comentado se utilizará el WordNet on line [24] para ampliar los términos que se incluirán dentro de la ontología, para ello el proceso será el siguiente:

- Se elegirán algunos términos relevantes del dominio ontológico.
- Se buscarán estos términos en WordNet on line, y se escogerán los resultados relacionados con el dominio seleccionado.
- Se consultarán tanto los hiperónimos para detectar superclases, los términos coordinados que devuelven las clases que se encuentran al mismo nivel que la consultada y los hipónimos que devuelve las subclases.
- Este proceso se realizará tantas veces como se necesite para dibujar el mapa completo de las clases necesarias que completará la ontología que se desarrolla.

Durante este proceso se tendrá un especial cuidado en escoger las clases devueltas por el WordNet, ya que elegirán las que pertenezcan al mismo dominio y en las traducciones de los conceptos seleccionados (WordNet trabaja con semántica inglesa).

6.4.2 Definición de la ontología

Para la implementación de esta ontología se ha elegido el dominio turístico, restringido a los alojamientos y playas que pertenecen al municipio de Calvià (Islas Baleares).

Esta ontología es útil a nivel turístico ya que al realizar búsquedas vacacionales, si se selecciona un municipio en concreto se tendrán todos sus alojamientos organizados por la playa o zona donde se ubican, teniendo también como respuesta por ejemplo el equipamiento del que dispone la playa o si posee o no bandera azul (atributos de la playa).

Superclases

- El primer elemento en consultar dentro del WordNet a sido **Municipio** para encontrar sus superclases. En el esquema siguiente se muestra el resultado obtenido:

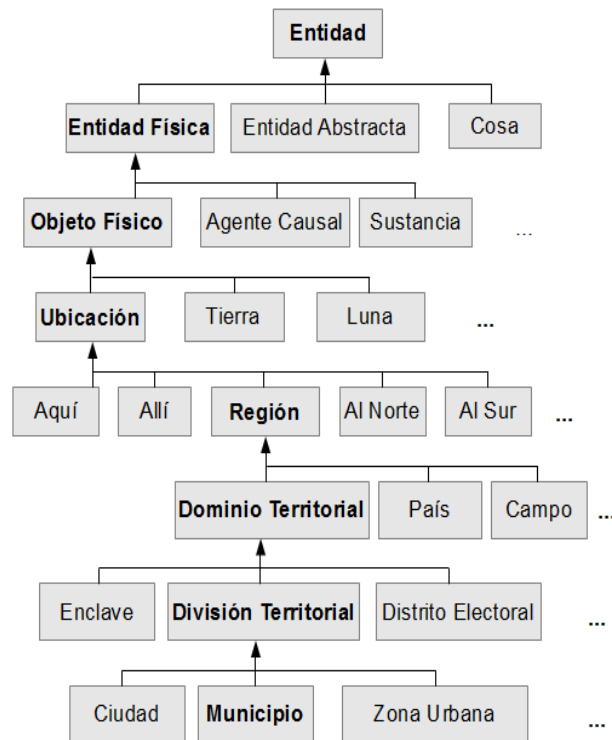


Figura 16 Superclases Municipio

- El segundo elemento a consultar es **Playa** para encontrar sus superclases. Este es el esquema del resultado obtenido:

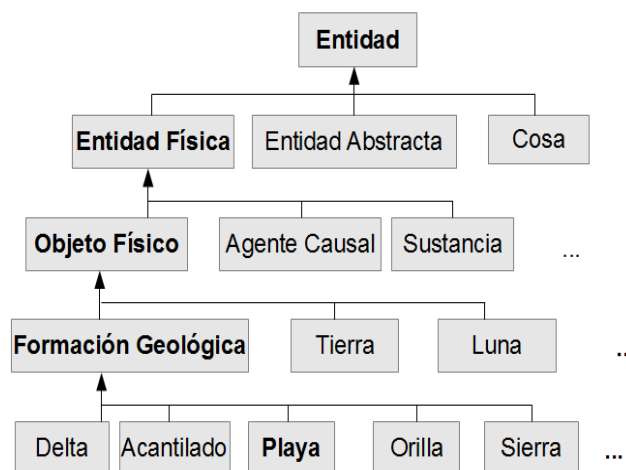


Figura 17 Superclases Playa

- Para el siguiente elemento consultado, **Localidad**, este es el esquema de las superclases:

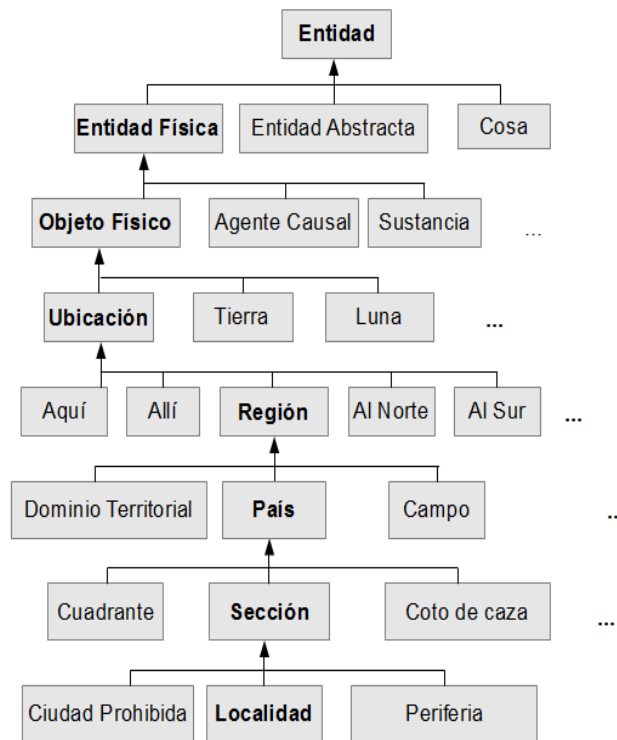


Figura 18 Superclases Localidad

- Para finalizar se mostrarán las superclases de la clase **Hotel** :

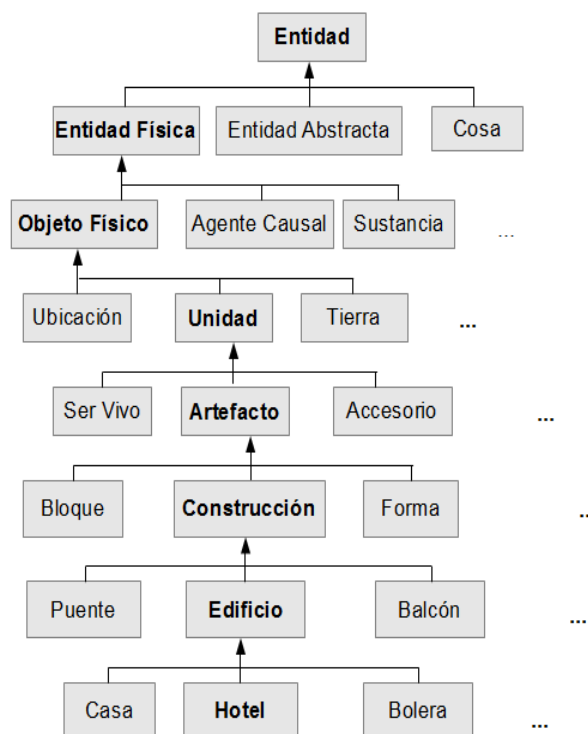


Figura 19 Superclases Hotel

Individuos de las clases:

En este apartado se describen todas las instancias de las clases desarrolladas.

- Para la clase **Municipio** se ha elegido una única instancia que será **Calvià**. Se ha elegido un único municipio para no sobre cargar de datos la ontología.
- Las instancias de **Localidad**, serán las siguientes:

| | |
|-------------------|--------------|
| Bendinat | Illetes |
| Calvià Vila | Magaluf |
| Cas Català | Palmanova |
| Costa de la Calma | Peguera |
| Costa d'en Blanes | Portals Nous |
| El Toro | Santa Ponça |
| Es Capdellà | Son Ferrer |
| Galatzó | |

Estas han sido elegidas al ser las divisiones poblacionales del municipio anterior.

- Para la clase **Playa** las instancias encontradas son las siguientes. Se muestran clasificadas por la zona donde pertenecen, esta relación se ha de indicar cuando se implemente la ontología con Protégé y su codificación se mostrará en los apartados siguientes.

Cas Català

Cala Brogi

Es Genetó

Rocamarina

Buguenvil·lia

El Toro

Cala de ses Penyes Roges

Costa de la Calma

Playa Monte de Oro

Cala Blanca

Illetes

Illetes de sa Torre

Cala Contesa

Platja d'Illetes

Costa d'en Blanes

Platja de Costa d'en Blanes

Magaluf

Portals Vells I i II

Portals Vells III

El Mago

Cala Aixada

Caló de la Bella Dona

Cala Falcó

Cala Vinyes

Platja de Magaluf

Peguera

Caló ses Llisses

Caló d'en Monjo

Cala Fornells

Palmira

Torà

La Romana

Portals Nous

Platja de l'Oratori

Platja de Portals Nous

Son Ferrer

Cala Figuera

Santa Ponça

Ses Rotes Velles

Platja de Santa Ponça

Caló d'en Pellisser

- Para finalizar se mostraran los individuos elegidos para la clase *Hotel*. En este caso se ha realizado una pequeña selección de 5 hoteles por localidad, ya que al ser una zona turística la cantidad de hoteles en la zona es bastante importante.

Cas Català

Hospes Maricel

Hotel Meliá de Mar

Hotel Zhero

El Toro

Hotel Port Adriano

Palmanova

Viva Palmanova

Son Maties Beach

Fergus Bahia Palma Nova

Son Caliu Hotel Spa Oasis

The St. Regis Mardavall

Mallorca Resort

Santa Ponça

Hesperia Playas de Mallorca

Iberostar Jardín del Sol

Hotel Casablanca

Club Santa Ponsa

Hotel Playas del Rey

Costa de la Calma

Luabay Galatzó

Illetes

Hotel Roc Illetas Playa

Barceló Illetas Albatros

Hotel Europe Playa Marina

Bon Sol

Hotel Riu Palace Bonanza Playa

Peguera

Valentín Paguera

Palmira Peguera

Hesperia Villamil

Hotel Cala Fornells

Costa d'en Blanes

H10 Punta Negra Boutique Hotel

Magaluf

Barceló Cala Viñas

Sol Magaluf Park

Sol Trinidad

Hotel Flamboyán Caribe

Hotel Samos

Portals Nous

Sallés Hotel Marina Portals

Hotel Costa Portals

Ola Aparthotel Tomir

Lindner Golf & Wellness Resort

Portals Nous

Hotel HSM Maria Luisa

Propiedades de las clases

Se van a definir algunas propiedades que definirán las relaciones de las clases.

– **ObjectProperty:**

- *esLocalidad* (dominio: Localidad/ rango: Municipio) define la relación entre el municipio y las localidades que le pertenecen.
- *esPlayaL* (dominio: Playa/ rango: Localidad) define la relación entre la localidad y las playas que le pertenecen.
- *esPlayaM* (dominio: Playa / rango: Municipio) define la relación entre el municipio y las playas que le pertenecen.
- *esHotelL* (dominio: Hotel/ rango: Localidad) define la relación entre la localidad y los hoteles que le pertenecen.
- *esHotelM* (dominio: Hotel / rango: Municipio) define la relación entre el municipio y los hoteles que le pertenecen.

– **inverseOf:**

- Se define la propiedad *esMunicipioL* (dominio: Municipio / rango: Localidad) como inversa de *esLocalidad*.
- Se definen las inversas de *esPlayaL*, *esPlayaM* como *esLocaPl* (dominio: Localidad / rango: Playa) y *esMunPl* (dominio: Municipio / rango: Playa).
- Se definen las inversas de *esHotelL*, *esHotelM* como *esLocalH* (dominio: Localidad / rango: Hotel) y *esMunH*(dominio: Municipio / rango: Hotel).

– **DatatypeProperty:**

- *nHabitantes* (dominio: Municipio / rango: reales no negativos) identifica el número de habitantes de un Municipio.
- *nEstrellas* (dominio: Hotel / rango: enteros) mostrará la categoría del hotel de 1 a 5 estrellas.
- *bAzul* (dominio: Playa / rango: booleano) contendrá si o no dependiendo si la playa posee o no bandera azul.

En la siguiente figura se representa las relaciones entre las clases con sus propiedades.

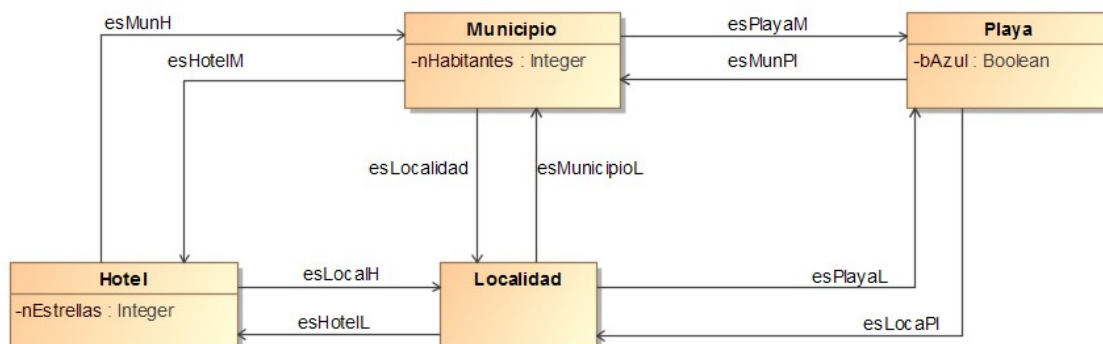


Figura 20 Clases y relaciones

– **Preguntas que se deberían de responder:**

Las relaciones y las propiedades de las clases deberían poder dar respuesta a las preguntas que se planteen sobre ellas. Como por ejemplo:

- ¿Cual es el municipio de una localidad?
- Las localidades de Calvià.
- ¿Que playas pertenecen a una localidad? Y ¿Cuales tiene bandera azul?

La implementación de estas preguntas y su respuesta se verá en el apartado siguiente en la Vista SPARQL.

6.4.3 Diseño ontología en Protégé

Una vez introducidos en Protégé todos los datos del apartado anterior que diseñan nuestra ontología este software nos proporciona diferentes interfaces gráficas donde se reflejan los datos introducidos. En la siguientes imágenes se verán las interfaces que se han utilizado y que facilitan la creación de la ontología en este software.

- **Vista Classes:** En esta primera vista se visualizan todas las clases y sus propiedades, muestra la jerarquía de clases creadas a partir de la superclase Thing que genera automáticamente Protégé. Como se puede observar se han introducido las clases estrictamente necesarias para que la ontología sea consistente.

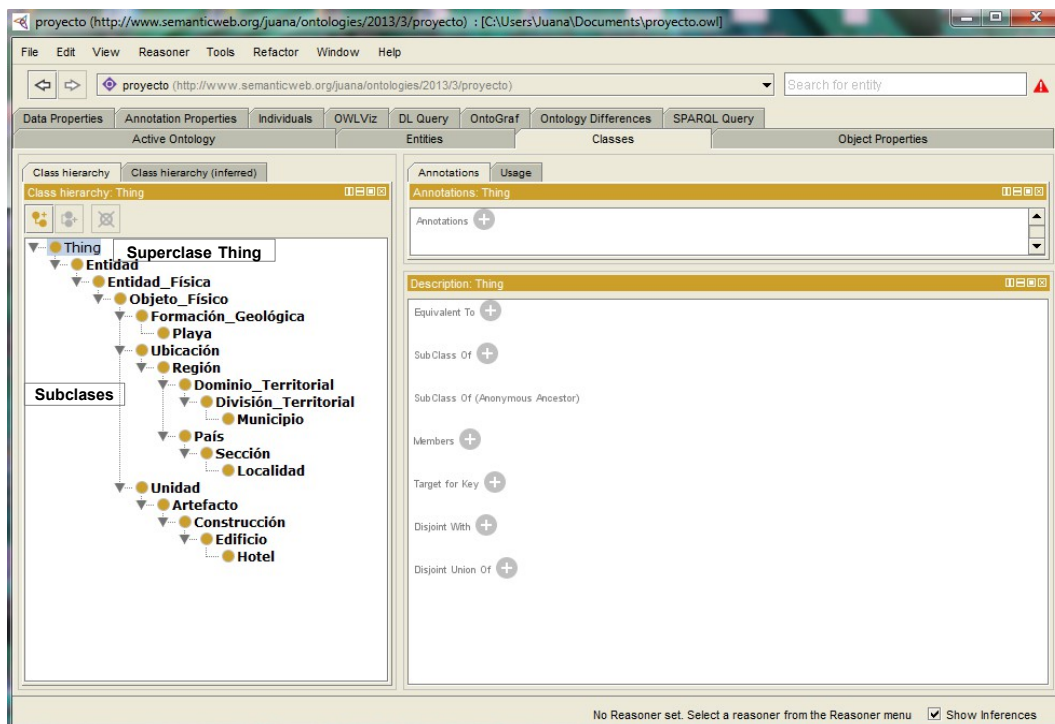


Figura 21 Vista Classes

En este mismo entorno realizando doble clic sobre una clase se obtiene su descripción general así como sus individuales, aunque para mayor claridad se ha utilizado la vista Entities.

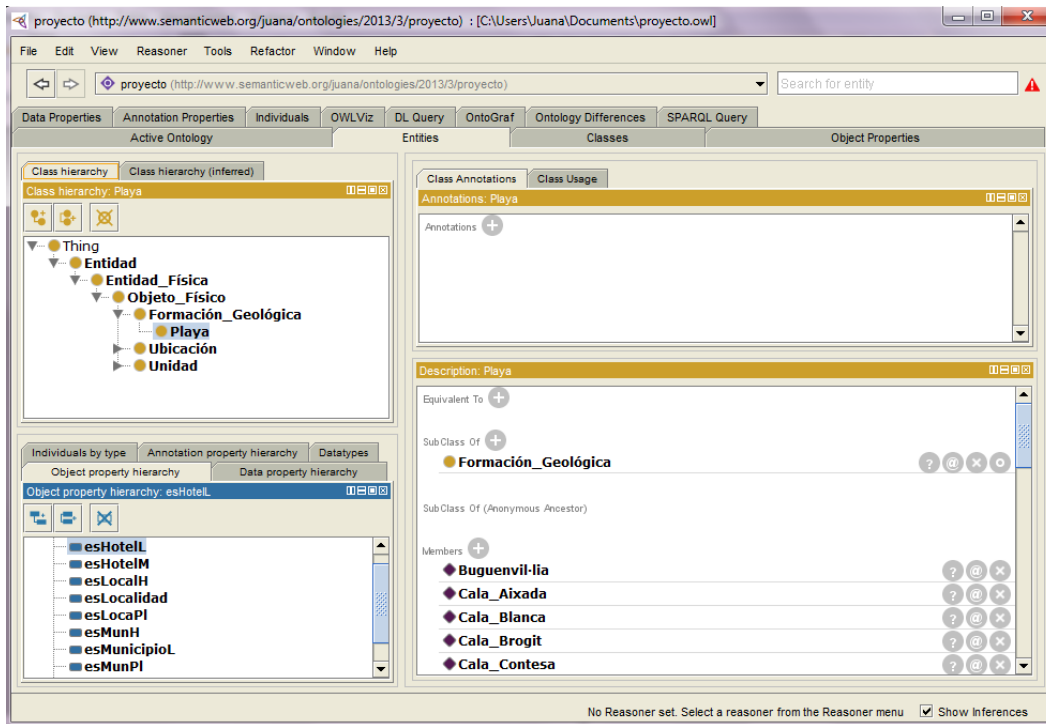


Figura 22 Vista Classes Propiedades

- **Vista Entities:** En esta vista se visualizan con más detalle la estructura de una clase, en ella se pueden definir sus ObjectProperty y DatatypeProperty, ver donde es utilizada y cuantas veces y sus miembros.

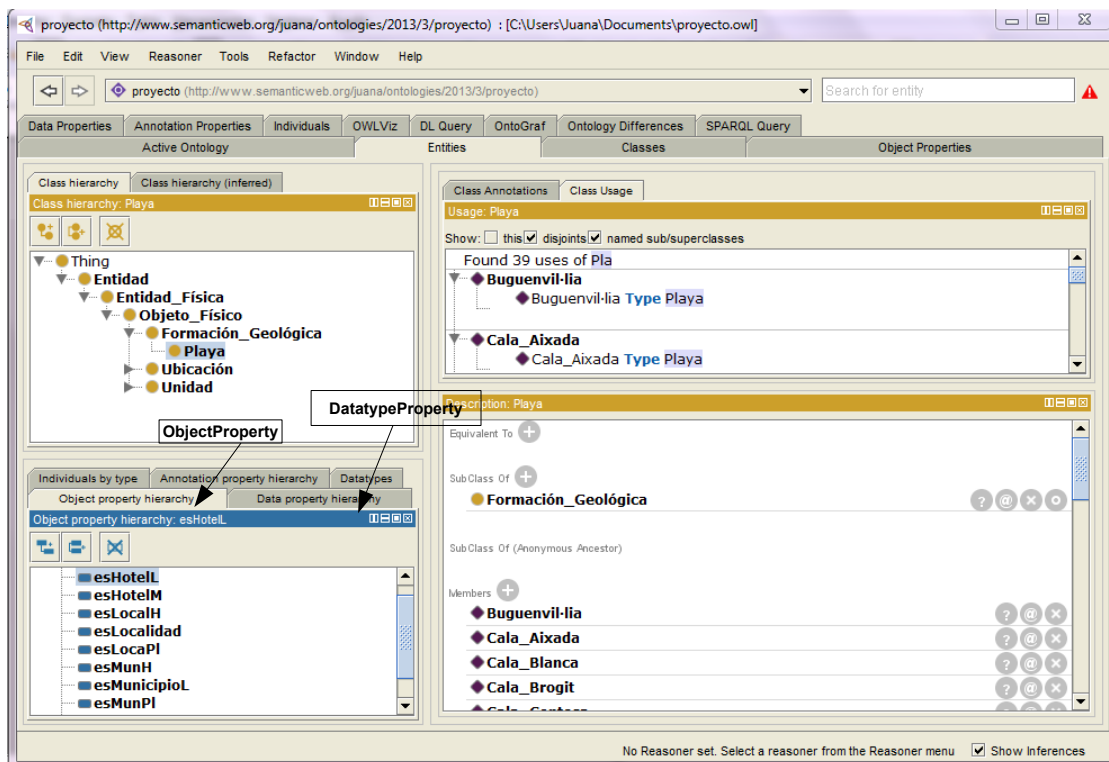


Figura 23 Vista Entities

- **Vista Individuals:** En esta pestaña se gestión con más facilidad los individuales de cada clase las propiedades que se le aplican a cada uno.

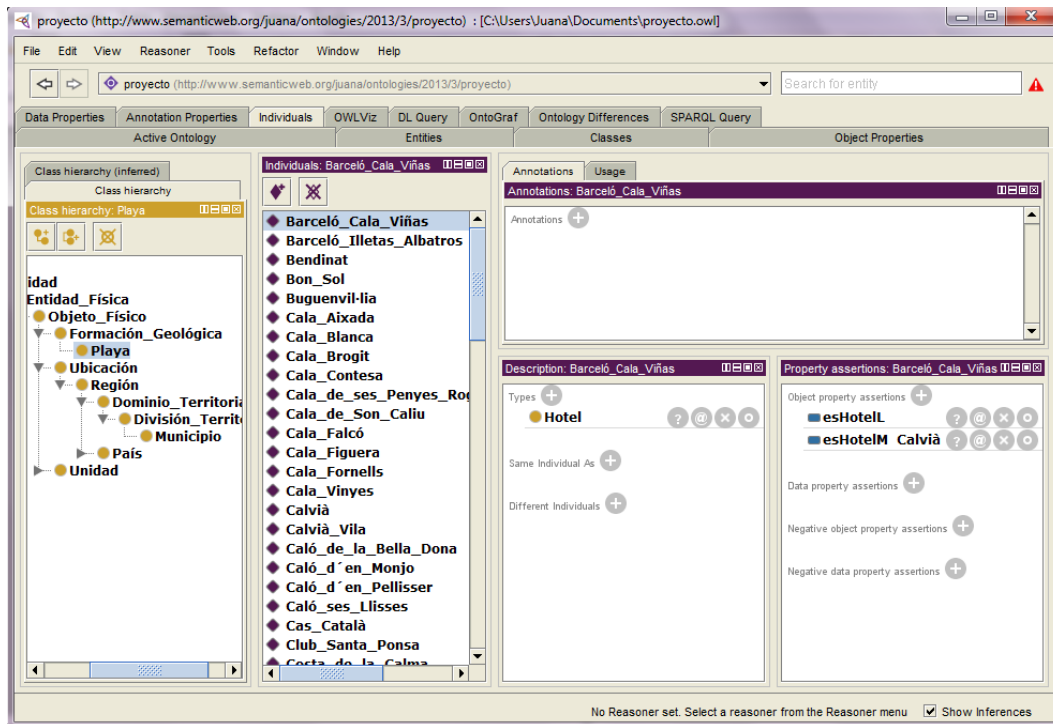


Figura 24 Vista Individuals

- **Vista Object Properties:** Para finalizar se muestra una imagen de la vista desde donde se facilita el trabajo de gestión de las propiedades de las clases ya que permite ir una por una viendo su uso y definiendo su dominio y rango.

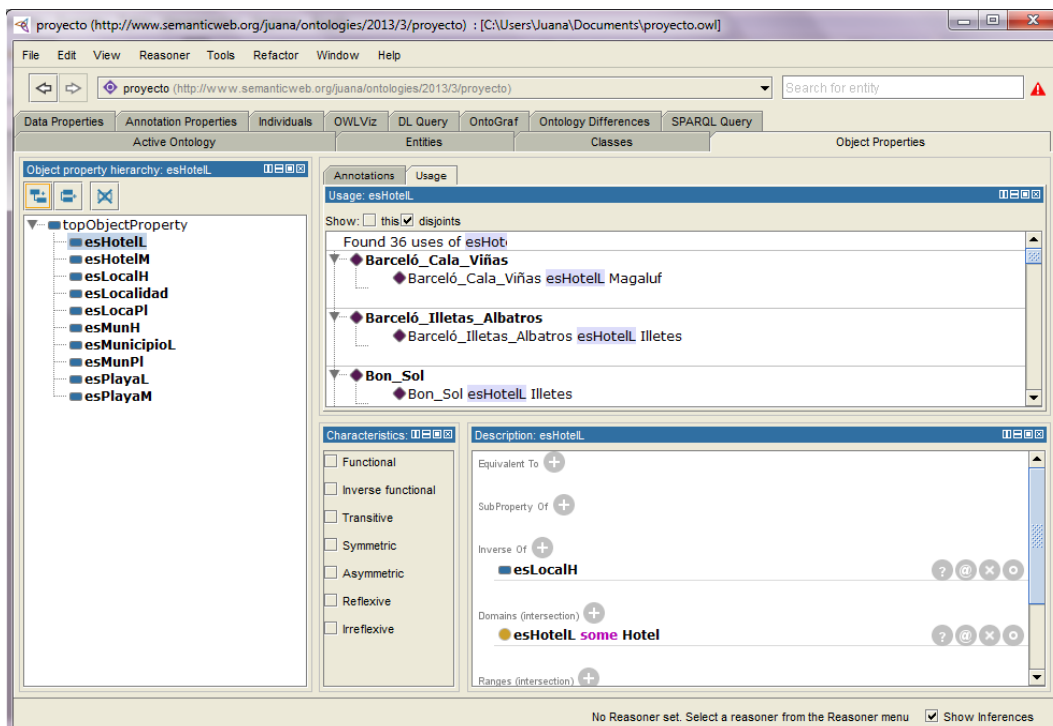


Figura 25 Vista Object Properties

- **Vista SPARQL:** En esta vista se ejecutan las consultas en SPARQL que han de responder esta ontología. Veremos como ejemplo alguna de las preguntas anteriormente formuladas:
 - ¿Cual es el municipio de una localidad?

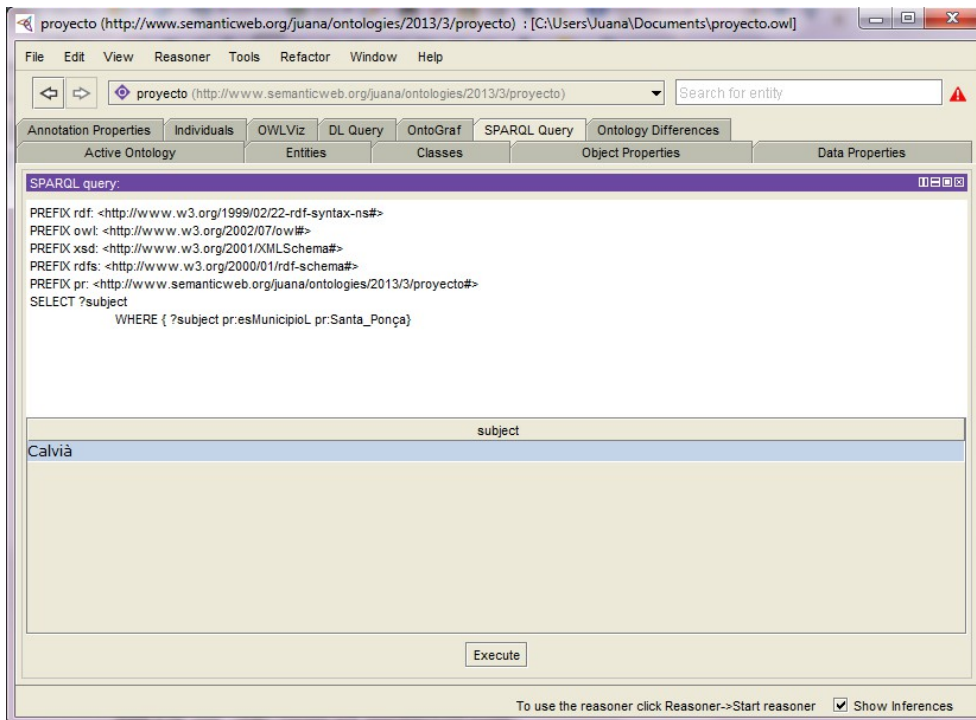


Figura 26 Consulta SPARQL Municipio

- Las localidades de Calvià.

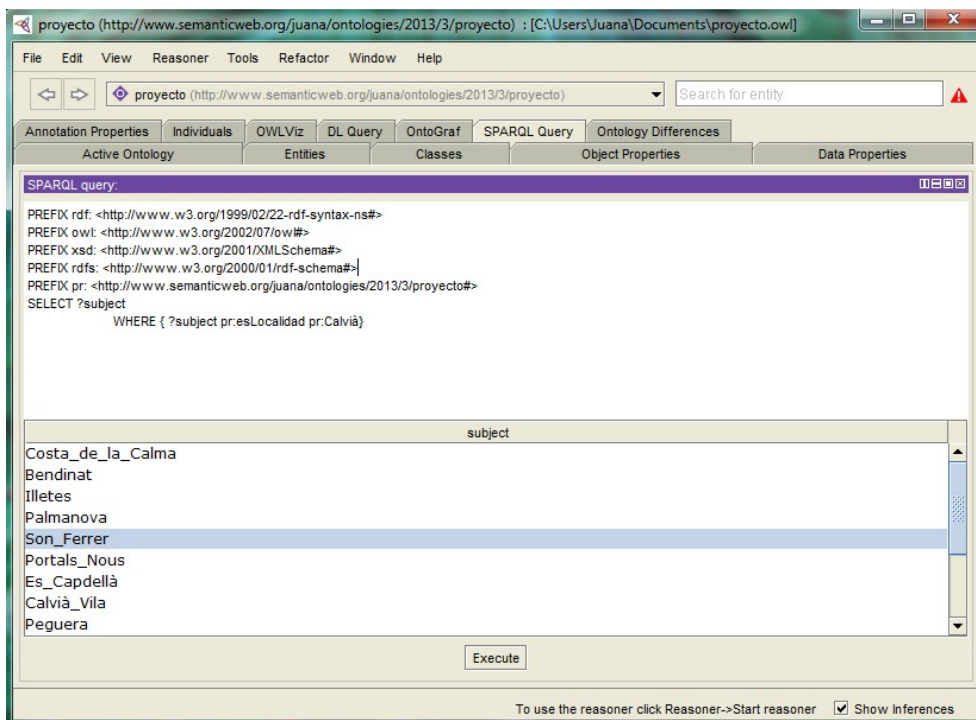


Figura 27 Consulta SPARQL Localidades

Para terminar se muestra parte de código obtenido a través del Protégé. Donde se puede observar como se declara las clases y sus propiedades. La totalidad del código se incluye en el Anexo I.

```

<?xml version="1.0"?>
<!DOCTYPE rdf:RDF [
  <!ENTITY owl "http://www.w3.org/2002/07/owl#" >
  <!ENTITY xsd "http://www.w3.org/2001/XMLSchema#" >
  <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
  <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
]
>
<rdf:RDF xmlns="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#"
  xml:base="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  <owl:Ontology rdf:about="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto/">
<!--
  //
  // Object Properties
  //
  //
  //
  <!-- http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#esHotelL -->

  <owl:ObjectProperty
    rdf:about="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#esHotelL">
    <rdfs:domain>
<owl:Restriction>
  <owl:onProperty
    rdf:resource="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#esHotelL"/>
    <owl:someValuesFrom
    rdf:resource="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#Hotel"/>
  </owl:Restriction>
  </rdfs:domain>
  <rdfs:range>
    <owl:Restriction>
      <owl:onProperty
        rdf:resource="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#esHotelL"/>
        <owl:someValuesFrom
        rdf:resource="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#Localidad"/>
      </owl:Restriction>
    </rdfs:range>
  </owl:ObjectProperty>
  ...
<!--
  //
  // Classes
  //
  //
  //
  <!-- http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#Artefacto -->

  <owl:Class
    rdf:about="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#Artefacto">
    <rdfs:subClassOf
    rdf:resource="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#Unidad"/>
  </owl:Class>

  <!-- http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#Construcción -->

  <owl:Class
    rdf:about="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#Construcción">
    <rdfs:subClassOf
    rdf:resource="http://www.semanticweb.org/juana/ontologies/2013/3/proyecto#Artefacto"/>
  </owl:Class>

```

Figura 28 Ejemplo Código RDF/OWL

7 Wiki Semántica

7.1 Introducción

Una wiki semántica [25] es una wiki que posee un modelo de conocimiento subyacente descrito en sus páginas. Las wikis semánticas ofrecen la posibilidad de capturar o identificar información acerca de los datos dentro de sus páginas y las relaciones entre estas, de modo que pueden ser consultados o exportados como una base de datos. En resumen, es un sitio web formado por páginas que permiten a sus usuarios añadir contenido pudiendo compartirlos y reutilizados.

Por ejemplo, si se crea una página en la wiki con los datos de nuestra ontología con las propiedades definidas como población, número de estrellas... En paralelo se puede consultar por los valores que tienen esas propiedades y crear nuevas páginas de la wiki, de una forma automática.

Sus características más significativas son:

- Organización del contenido.
- El usuario puede modificar su contenido, aunque se puede restringir la edición y acceso a este.
- Se pueden incluir elementos multimedia y páginas web. Estos elementos pueden estar o no dentro del mismo servidor que la wiki.
- Publicación ágil ya que se utiliza un navegador web.
- Facilidad en corrección de errores y control de versiones.
- Utilización de contenidos por agentes y servicios web tanto propios como externos.
- Uso de tecnologías de la Web Semántica, que posibilitan el uso u creación de ontologías.
- Edición de metadatos.
- Generación de contenidos en modo colaborativo.
- Utilización de lenguajes de marcado y metadatos estándares en la web.
- Dispone de un editor que permite notación semántica de los contenidos.
- Generación de estructuras de navegación de contenidos en base a las anotaciones semánticas disponibles.
- Personalización por parte del usuario de las opciones de navegación.
- Integración de motores de razonamiento y de búsqueda semánticos.
- Da la posibilidad de interactuar con otros recursos de Web Semántica para el descubrimiento de información asociada.

7.2 Motores de Wiki semánticas

En este apartado se describirán los dos motores sugeridos para realizar el proyecto.

- **Semantic MediaWiki [26]:** Es uno de los motores más utilizados y es una extensión de la herramienta MediaWiki que le proporciona la capacidad de añadir contenidos semánticos a cualquier wiki que la utilice.

Esta herramienta es sobre la que esta construida Wikipedia, que es una de las plataformas más utilizadas para el desarrollo de wikis. Para crear asociaciones semánticas entre artículos se utiliza una sintaxis similar a los typed links (enlace de hipertexto) que se utilizan para enlazar los artículos en una wiki. Esta herramienta transforma los typed links en los atributos RDF, usando XSD para que su formato sea un documento válido XML.

En sus últimas versiones permite importar las ontologías creadas en cualquier lenguaje RDF, no solo OWL y tiene extensiones para realizar consultas en SPARQL que anteriormente no permitía. Aunque no permite la importación de estas consultas desde un sistema externo.

- **OntoWiki [27]**: Es una herramienta wiki que permite la generación de ontologías de forma colaborativa, facilita la presentación visual de una base de conocimientos como la información, con diferentes puntos de vista sobre los datos . Permite creación intuitiva de contenido semántico, posee un modo de edición en línea de contenido RDF. Se fomenta la colaboración de los usuarios sobre los contenidos permitiendo hacer el seguimiento de los cambios y de los comentarios, lo que permite para valorar y medir la popularidad de los contenidos. OntoWiki mejora la navegación entre contenidos y recuperación de datos, ofreciendo una búsqueda semántica mejorada.

Es destacable la agilidad para presentar las bases de conocimiento y lo intuitivo que es su editor de contenido RDF.

7.3 Instalación OntoWiki

Para la realización de este proyecto se ha elegido el motor semántico OntoWiki. Se ha elegido este motor semántico ya que:

- Se ha considerado un software más ágil a la hora de importar nuestra ontología. Ya que una vez instalada la aplicación la importación se realiza a través de esta fácilmente en cambio por la documentación leída de Semantic MediaWiki la importación de ontologías se realiza mediante comandos.
- Por la documentación consultada de ambos motores semánticos, la instalación de OntoWiki es más simple, necesita menos configuraciones y genera menos errores que Semantic MediaWiki.
- Se ha tenido en cuenta la usabilidad del software que permite al usuario crear o modificar su ontología con facilidad.

No ha influido en la decisión, ya que ninguno de los dos softwares lo permite,

- la importación de consultas en SPARQL, aunque si la creación y publicación de ellas desde la aplicación,
- el poder trabajar en otra codificación que no sea UTF-8 (Semantic MediaWiki dispone de una extensión para trabajar en Unicode más pesada que la recomendada para UTF-8).

Para efectuar la instalación de este software se ha instalado los siguientes herramientas:

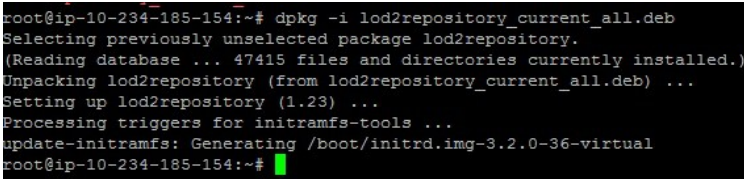
- Sistema operativo: Ubuntu.
- Servidor Web: Apache 2 con las librerías PHP 5.
- Base de datos: MySQL

Como servidor para estas herramientas se ha elegido una micro instancia gratuita de Amazon, servidor virtual, que permitirá que nuestro proyecto sea accesible a través de la red.

La instalación se ha ejecutado con el usuario “root” del sistema siguiendo los pasos que se van a continuación:

- Primero se ha de instalar el repositorio LOD2

```
dpkg -i lod2repository_current_all.deb
root@ip-10-234-185-154:~# dpkg -i lod2repository_current_all.deb
Selecting previously unselected package lod2repository.
(Reading database ... 47415 files and directories currently installed.)
Unpacking lod2repository (from lod2repository_current_all.deb) ...
Setting up lod2repository (1.23) ...
Processing triggers for initramfs-tools ...
update-initramfs: Generating /boot/initrd.img-3.2.0-36-virtual
root@ip-10-234-185-154:~#
```



```
root@ip-10-234-185-154:~# dpkg -i lod2repository_current_all.deb
Selecting previously unselected package lod2repository.
(Reading database ... 47415 files and directories currently installed.)
Unpacking lod2repository (from lod2repository_current_all.deb) ...
Setting up lod2repository (1.23) ...
Processing triggers for initramfs-tools ...
update-initramfs: Generating /boot/initrd.img-3.2.0-36-virtual
root@ip-10-234-185-154:~#
```

Figura 29 Instalación LOD2

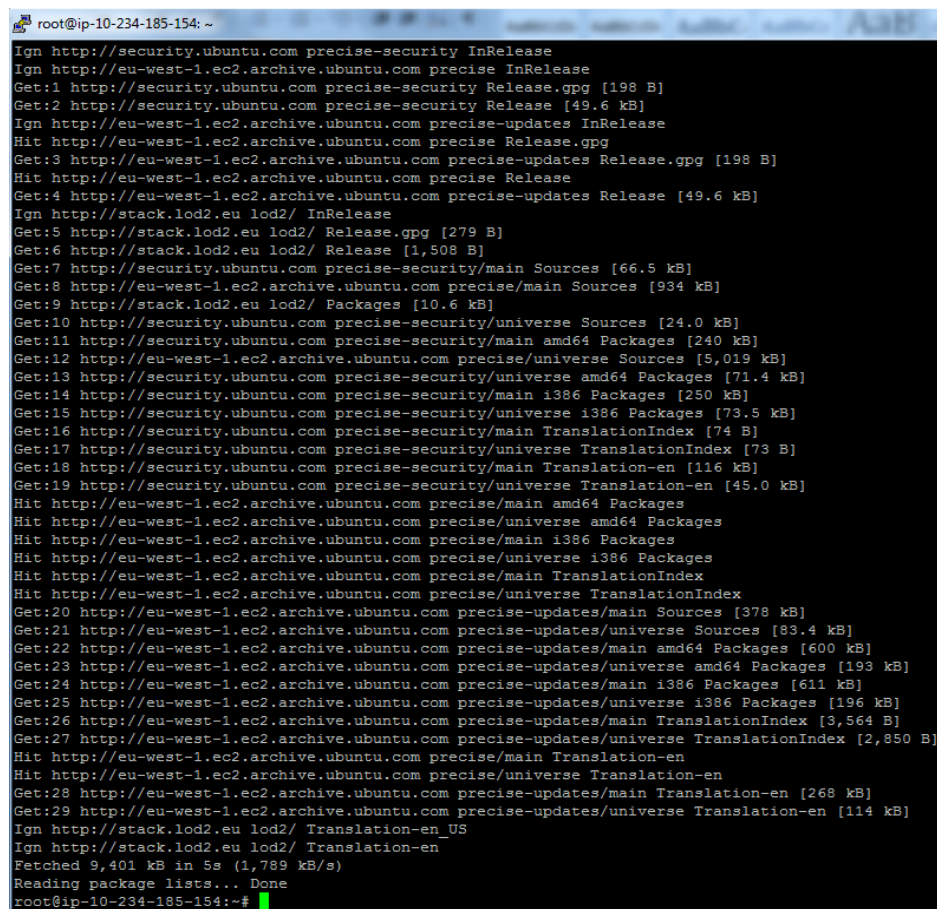
- Se actualiza el repositorio de fuentes de software de Ubuntu

```
apt-get update
Ign http://security.ubuntu.com precise-security InRelease
Ign http://eu-west-1.ec2.archive.ubuntu.com precise InRelease
Get:1 http://security.ubuntu.com precise-security Release.gpg [198 B]
Get:2 http://security.ubuntu.com precise-security Release [49.6 kB]
Ign http://eu-west-1.ec2.archive.ubuntu.com precise-updates InRelease
Hit http://eu-west-1.ec2.archive.ubuntu.com precise Release.gpg
Get:3 http://eu-west-1.ec2.archive.ubuntu.com precise-updates Release.gpg [198 B]
Hit http://eu-west-1.ec2.archive.ubuntu.com precise Release
Get:4 http://eu-west-1.ec2.archive.ubuntu.com precise-updates Release [49.6 kB]
Ign http://stack.lod2.eu lod2/ InRelease
Get:5 http://stack.lod2.eu lod2/ Release.gpg [279 B]
Get:6 http://stack.lod2.eu lod2/ Release [1,508 B]
Get:7 http://security.ubuntu.com precise-security/main Sources [66.5 kB]
Get:8 http://eu-west-1.ec2.archive.ubuntu.com precise/main Sources [934 kB]
Get:9 http://stack.lod2.eu lod2/ Packages [10.6 kB]
Get:10 http://security.ubuntu.com precise-security/universe Sources [24.0 kB]
Get:11 http://security.ubuntu.com precise-security/main amd64 Packages [240 kB]
Get:12 http://eu-west-1.ec2.archive.ubuntu.com precise/universe Sources [5,019 kB]
Get:13 http://security.ubuntu.com precise-security/universe amd64 Packages [71.4 kB]
Get:14 http://security.ubuntu.com precise-security/main i386 Packages [250 kB]
Get:15 http://security.ubuntu.com precise-security/universe i386 Packages [73.5 kB]
Get:16 http://security.ubuntu.com precise-security/main TranslationIndex [74 B]
Get:17 http://security.ubuntu.com precise-security/universe TranslationIndex [73 B]
Get:18 http://security.ubuntu.com precise-security/main Translation-en [116 kB]
Get:19 http://security.ubuntu.com precise-security/universe Translation-en [45.0 kB]
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/main amd64 Packages
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/universe amd64 Packages
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/main i386 Packages
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/universe i386 Packages
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/main TranslationIndex
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/universe TranslationIndex
Get:20 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/main Sources [378 kB]
Get:21 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/universe Sources [83.4 kB]
Get:22 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/main amd64 Packages [600 kB]
Get:23 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/universe amd64 Packages [193 kB]
Get:24 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/main i386 Packages [611 kB]
Get:25 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/universe i386 Packages [196
```

```

kB]
Get:26 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/main TranslationIndex [3,564
B]
Get:27 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/universe TranslationIndex
[2,850 B]
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/main Translation-en
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/universe Translation-en
Get:28 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/main Translation-en [268 kB]
Get:29 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/universe Translation-en [114
kB]
Ign http://stack.lod2.eu lod2/ Translation-en_US
Ign http://stack.lod2.eu lod2/ Translation-en
Fetched 9,401 kB in 5s (1,789 kB/s)
Reading package lists... Done
root@ip-10-234-185-154:~#

```



```

root@ip-10-234-185-154:~#
Ign http://security.ubuntu.com precise-security InRelease
Ign http://eu-west-1.ec2.archive.ubuntu.com precise InRelease
Get:1 http://security.ubuntu.com precise-security Release.gpg [198 B]
Get:2 http://security.ubuntu.com precise-security Release [49.6 kB]
Ign http://eu-west-1.ec2.archive.ubuntu.com precise-updates InRelease
Hit http://eu-west-1.ec2.archive.ubuntu.com precise Release.gpg
Get:3 http://eu-west-1.ec2.archive.ubuntu.com precise-updates Release.gpg [198 B]
Hit http://eu-west-1.ec2.archive.ubuntu.com precise Release
Get:4 http://eu-west-1.ec2.archive.ubuntu.com precise-updates Release [49.6 kB]
Ign http://stack.lod2.eu lod2/ InRelease
Get:5 http://stack.lod2.eu lod2/ Release.gpg [279 B]
Get:6 http://stack.lod2.eu lod2/ Release [1,508 B]
Get:7 http://security.ubuntu.com precise-security/main Sources [66.5 kB]
Get:8 http://eu-west-1.ec2.archive.ubuntu.com precise/main Sources [934 kB]
Get:9 http://stack.lod2.eu lod2/ Packages [10.6 kB]
Get:10 http://security.ubuntu.com precise-security/universe Sources [24.0 kB]
Get:11 http://security.ubuntu.com precise-security/main amd64 Packages [240 kB]
Get:12 http://eu-west-1.ec2.archive.ubuntu.com precise/universe Sources [5,019 kB]
Get:13 http://security.ubuntu.com precise-security/universe amd64 Packages [71.4 kB]
Get:14 http://security.ubuntu.com precise-security/main i386 Packages [250 kB]
Get:15 http://security.ubuntu.com precise-security/universe i386 Packages [73.5 kB]
Get:16 http://security.ubuntu.com precise-security/main TranslationIndex [74 B]
Get:17 http://security.ubuntu.com precise-security/universe TranslationIndex [73 B]
Get:18 http://security.ubuntu.com precise-security/main Translation-en [116 kB]
Get:19 http://security.ubuntu.com precise-security/universe Translation-en [45.0 kB]
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/main amd64 Packages
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/universe amd64 Packages
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/main i386 Packages
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/universe i386 Packages
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/main TranslationIndex
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/universe TranslationIndex
Get:20 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/main Sources [378 kB]
Get:21 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/universe Sources [83.4 kB]
Get:22 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/main amd64 Packages [600 kB]
Get:23 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/universe amd64 Packages [193 kB]
Get:24 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/main i386 Packages [611 kB]
Get:25 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/universe i386 Packages [196 kB]
Get:26 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/main TranslationIndex [3,564 B]
Get:27 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/universe TranslationIndex [2,850 B]
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/main Translation-en
Hit http://eu-west-1.ec2.archive.ubuntu.com precise/universe Translation-en
Get:28 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/main Translation-en [268 kB]
Get:29 http://eu-west-1.ec2.archive.ubuntu.com precise-updates/universe Translation-en [114 kB]
Ign http://stack.lod2.eu lod2/ Translation-en_US
Ign http://stack.lod2.eu lod2/ Translation-en
Fetched 9,401 kB in 5s (1,789 kB/s)
Reading package lists... Done
root@ip-10-234-185-154:~#

```

Figura 30 Instalación LOD2

- Se ha elegido la instalación de OntoWiki con la base de datos MySQL.

```

apt-get install ontowiki-mysql
root@ip-10-234-185-154:~# apt-get install ontowiki-mysql
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  apache2-mpm-prefork apache2-utils apache2.2-bin apache2.2-common libapache2-mod-php5
  libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libcap2 libdbd-mysql-perl libdbi-perl liberfurt-
  php libhtml-template-perl
  libjs-rdfaauthor libmarkdown-php libmysqlclient18 libnet-daemon-perl libphp-pclzip
  liblprpc-perl

```



```

libzend-framework-php mysql-client-5.5 mysql-client-core-5.5 mysql-common mysql-server
mysql-server-5.5
mysql-server-core-5.5 ontowiki-common php5 php5-cli php5-common php5-curl php5-mysql ssl-
cert tinymce zend-framework
zend-framework-bin
Suggested packages:
apache2-doc apache2-suexec apache2-suexec-custom php-pear libipc-sharedcache-perl libjs-
dojo-core libjs-dojo-dijit

libjs-dojo-dojox libterm-readkey-perl tinyca mailx owcli php5-suhosin openssl-blacklist
The following NEW packages will be installed:
apache2-mpm-prefork apache2-utils apache2.2-bin apache2.2-common libapache2-mod-php5
libapr1 libaprutil1
libaprutil1-dbd-sqlite3 libaprutil1-ldap libcap2 libdbd-mysql-perl libdbi-perl liberfurt-
php libhtml-template-perl
libjs-rdfaauthor libmarkdown-php libmysqlclient18 libnet-daemon-perl libphp-pclzip
libplrpc-perl
libzend-framework-php mysql-client-5.5 mysql-client-core-5.5 mysql-common mysql-server
mysql-server-5.5
mysql-server-core-5.5 ontowiki-common ontowiki-mysql php5 php5-cli php5-common php5-curl
php5-mysql ssl-cert tinymce
zend-framework zend-framework-bin
0 upgraded, 38 newly installed, 0 to remove and 65 not upgraded.
Need to get 49.8 MB of archives.
After this operation, 178 MB of additional disk space will be used.
Do you want to continue [Y/n]? Y

```

```

root@ip-10-234-185-154:~# apt-get install ontowiki-mysql
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
apache2-mpm-prefork apache2-utils apache2.2-bin apache2.2-common libapache2-mod-php5 libapr1 libaprutil1
libaprutil1-dbd-sqlite3 libaprutil1-ldap libcap2 libdbd-mysql-perl libdbi-perl liberfurt-php libhtml-template-perl
libjs-rdfaauthor libmarkdown-php libmysqlclient18 libnet-daemon-perl libphp-pclzip libplrpc-perl
libzend-framework-php mysql-client-5.5 mysql-client-core-5.5 mysql-common mysql-server mysql-server-5.5
mysql-server-core-5.5 ontowiki-common php5 php5-cli php5-common php5-curl php5-mysql ssl-cert tinymce zend-framework
zend-framework-bin
Suggested packages:
apache2-doc apache2-suexec apache2-suexec-custom php-pear libipc-sharedcache-perl libjs-dojo-core libjs-dojo-dijit
libjs-dojo-dojox libterm-readkey-perl tinyca mailx owcli php5-suhosin openssl-blacklist
The following NEW packages will be installed:
apache2-mpm-prefork apache2-utils apache2.2-bin apache2.2-common libapache2-mod-php5 libapr1 libaprutil1
libaprutil1-dbd-sqlite3 libaprutil1-ldap libcap2 libdbd-mysql-perl libdbi-perl liberfurt-php libhtml-template-perl
libjs-rdfaauthor libmarkdown-php libmysqlclient18 libnet-daemon-perl libphp-pclzip libplrpc-perl
libzend-framework-php mysql-client-5.5 mysql-client-core-5.5 mysql-common mysql-server mysql-server-5.5
mysql-server-core-5.5 ontowiki-common ontowiki-mysql php5 php5-cli php5-common php5-curl php5-mysql ssl-cert tinymce
zend-framework zend-framework-bin
0 upgraded, 38 newly installed, 0 to remove and 65 not upgraded.
Need to get 49.8 MB of archives.
After this operation, 178 MB of additional disk space will be used.
Do you want to continue [Y/n]? Y

```

Figura 31 Instalación OntoWiki-Mysql

- Una vez instalado se nos solicita contraseña del usuario root de MySQL

```

root@ip-10-234-185-154:~#
Package configuration
=====
##### Configuring mysql-server-5.5 #####
While not mandatory, it is highly recommended that you set a password for the MySQL administrative "root" user.
If this field is left blank, the password will not be changed.
New password for the MySQL "root" user:

```

Figura 32 Contraseña Mysql

La base de datos pide doble confirmación del password.

```
grant privileges to ontowiki user;
flush privileges;
Enabling module rewrite.
To activate the new configuration, you need to run:
  service apache2 restart
Module env already enabled
* Restarting web server apache2
---- ending postinst configure
Processing triggers for libc-bin ...
ldconfig deferred processing now taking place
```

Figura 33 Confirmación contraseña Mysql

En este punto finaliza de la instalación.

Continuamos con la configuración de OntoWiki.

- Para visualizar el usuario y password de SuperAdmin vemos el contenido de /etc/ontowiki/config.ini

```
[private]
;
; Database setup
; In most cases you only need to change username, password
; and database name (dbname).
store.backend = zenddb ; zenddb, virtuoso, arc, comparer
store.zenddb.dbname = "ontowiki_deb"
store.zenddb.username = "ontowiki_deb"
store.zenddb.password = "usepwgenhere"
store.zenddb.dbtype = mysql ; mysql
```

Figura 34 Usuario/Pass SuperAdmin

Usuario: “ontowiki_deb”
 Password: “usepwgenhere”

- Hacemos login con estos datos en nuestro servidor <http://ec2-46-137-55-124.eu-west-1.compute.amazonaws.com/ontowiki/>

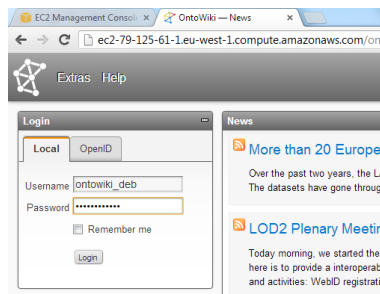


Figura 35 Acceso SuperAdmin

- Entramos en OntoWiki.

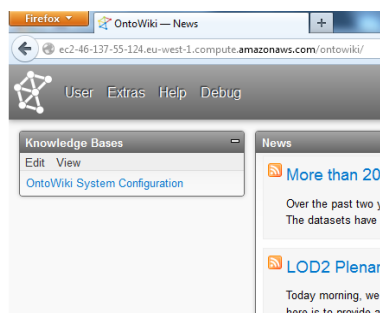


Figura 36 Acceso OntoWiki

- Se accede a “OntoWiki System Configuration”.

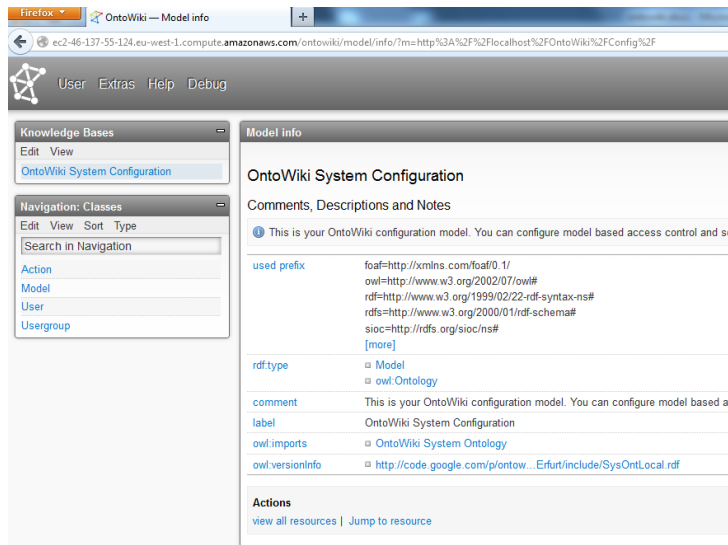


Figura 37 OntoWiki System Configuration

- Se accede al apartado “User”.

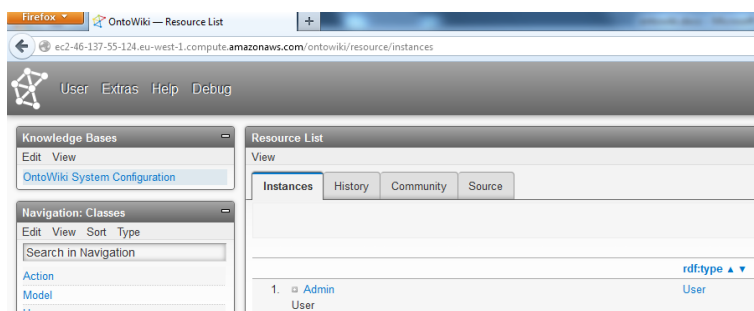


Figura 38 OntoWiki User

- Se selecciona el usuario “Admin” para definirle un password.



Figura 39 OntoWiki configuración password Admin

- Se pulsa sobre “Add Property”.

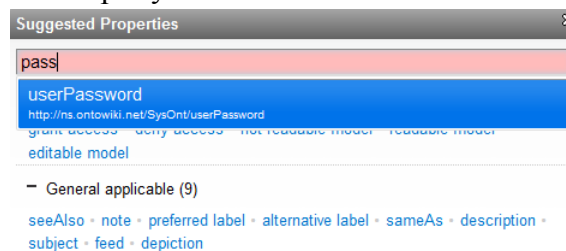


Figura 40 OntoWiki Add Property

- Se añade la propiedad “userPassword”.

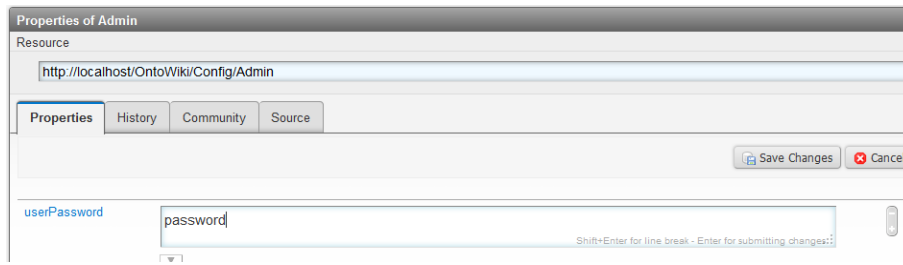


Figura 41 OntoWiki userPassword

- Se guardan los cambios, “Save Changes”.

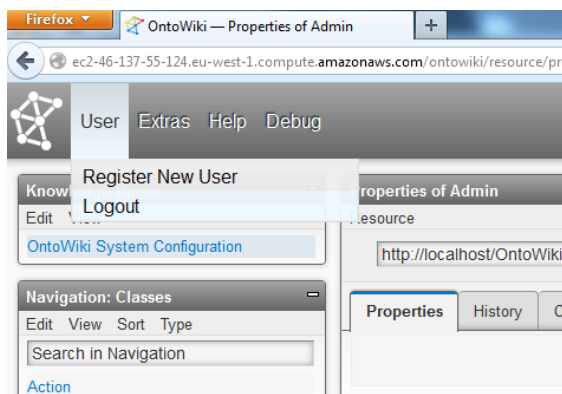


Figura 42 OntoWiki Logout

- Se sale de la aplicación “Logout”. Y se vuelve a Iniciar la sesión con “Admin” y el password anterior.

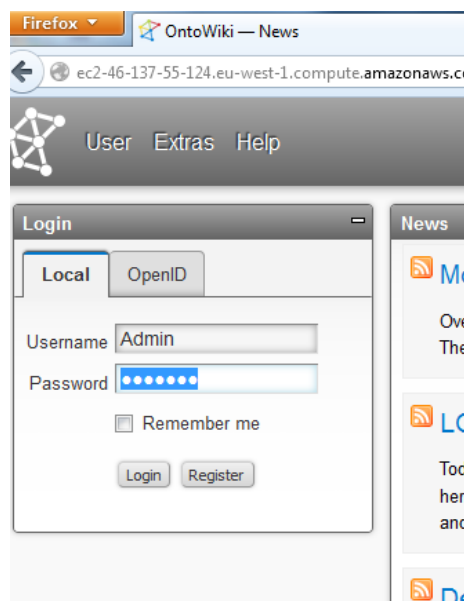


Figura 43 Acceso Admin

- Se accede a “Preferences”.

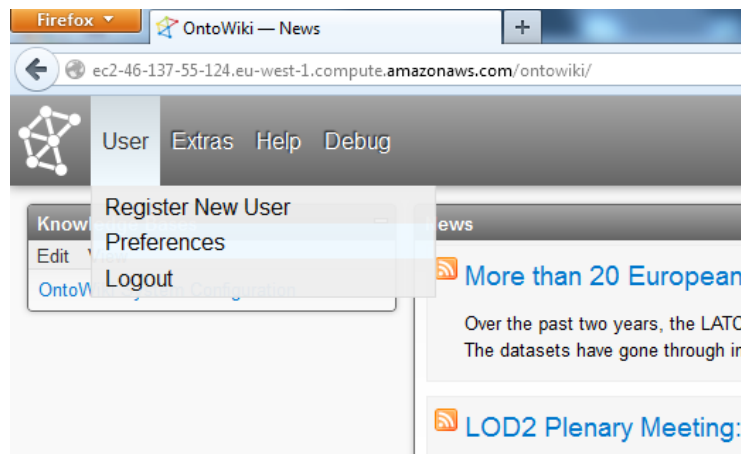


Figura 44 OntoWiki Preferences

- Se accede con el usuario Admin para cambiarle la contraseña y encriptarla. Si no se realiza este paso la contraseña se queda visible.

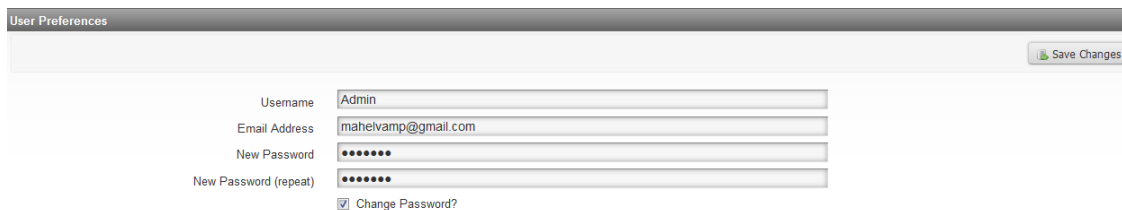


Figura 45 OntoWiki Cambio Password Admin

- Se guardan los cambios “Save Changes” y se encripta el password.

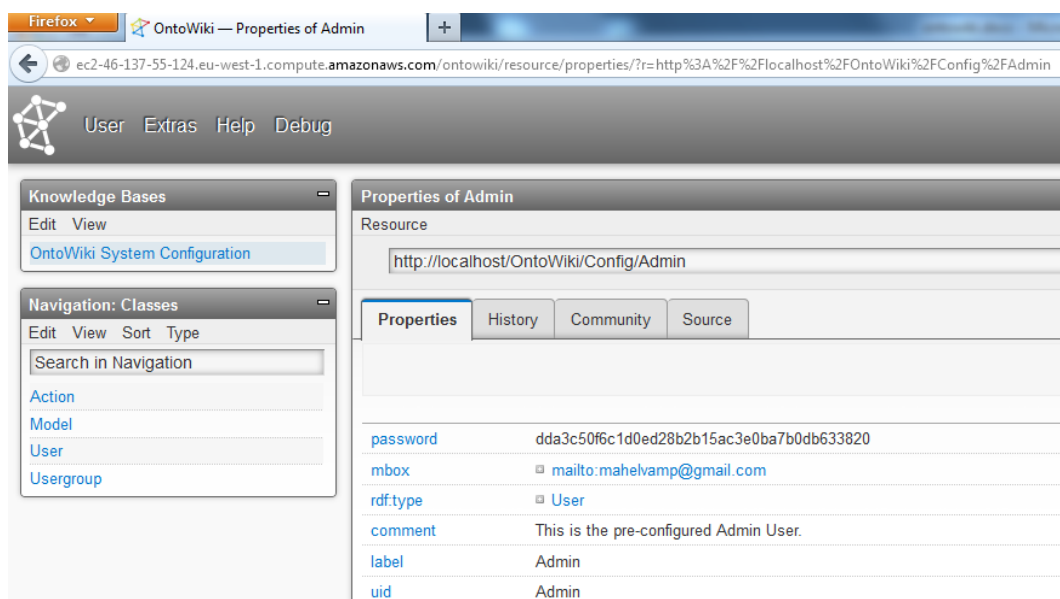


Figura 46 OntoWiki Admin Properties

Con este último paso se deja configurado OntoWiki para proceder a la carga de la ontología diseñada, que se verá en el siguiente apartado.

7.4 Carga de una ontología

Como se ha comentado anteriormente desde la aplicación es fácil importar una ontología ya creada desde Protégé. Las limitaciones son las siguientes:

- La ontología se ha de guardar en formato rdf/xml. Aunque también admite turtle.
- No se pueden incluir acentos o caracteres extraños al nombrar tanto las clases, subclases e individuales.

Veamos el proceso de importación:

- Una vez logeados en OntoWiki con usuario administrador acceder a Edit, tal y como se ve a continuación.

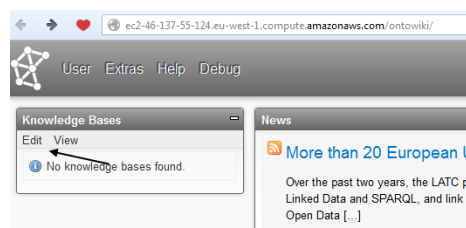


Figura 47 OntoWiki Botón Edit

Al pulsar encima aparece la opción Create Knowledge Base, que al pulsar sobre ella nos redirige a la pantalla donde se realizará la importación de la ontología.

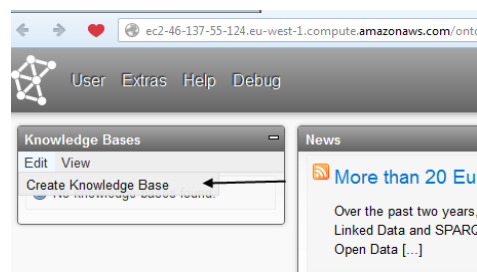


Figura 48 OntoWiki Botón Create Knowledge Base

- En esta página es donde se introducen la datos básicos de la ontología a crear.

 A screenshot of the 'Create New Knowledge Base' dialog box. The title bar says 'Create New Knowledge Base'. There are two buttons in the top right: 'Create Knowledge Base' and 'Cancel'. The main area is titled 'Create Knowledge Base'. It has a text input field for 'Knowledge Base URI'. Below that are four radio button options: 'Create an Empty Knowledge Base', 'Import From the Web' (which is selected), 'Upload a File', and 'Paste Source'. Below these options is a section titled 'Import From the Web' which contains a text input field for 'Location'. A tooltip is visible over the 'Location' field with the text: 'Enter the URL where the model should be downloaded from. If you leave the field empty the model URI will be used.'

Figura 49 OntoWiki Create Knowledge Base

Los campos que en nuestro caso se van a utilizar son:

- URI: En este campo se introduce la URI de la ontología. Siguiendo las indicaciones de las FAQ's de Ontowiki inicialmente nuestra URI será <http://localhost/ontowiki/jperea/>.
- Se selecciona Upload a File.

Una vez realizados estos pasos la página cambia para solicitarnos otros datos adicionales. El tipo de fichero que se va importar, en nuestro caso XML/RDF y su ubicación. Una vez rellenados estos datos se pulsa sobre el botón Create Knowledge Base, ver la siguiente figura.

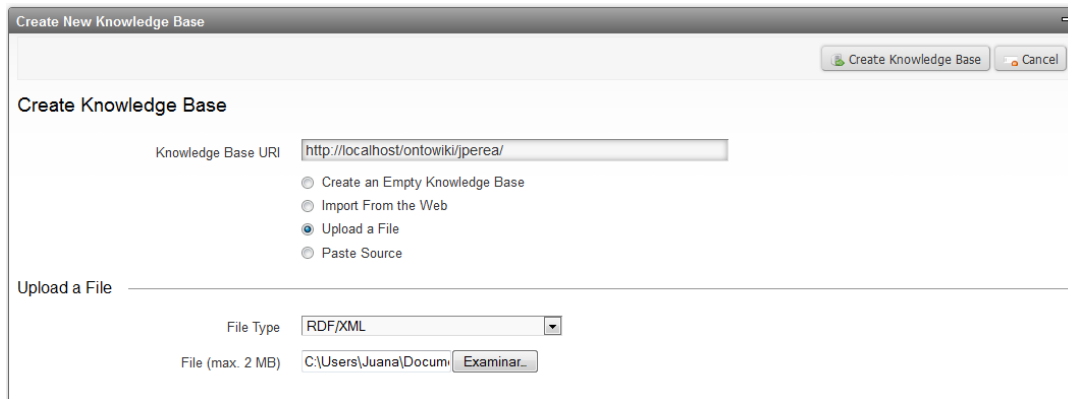


Figura 50 OntoWiki Upload Ontowiki

- En el siguiente paso de la importación la aplicación nos sugiere utilizar su base de conocimiento ya que nuestra ontología no la importa. Se selecciona Use SysBase Model y se pulsa sobre Save Model Configuration.

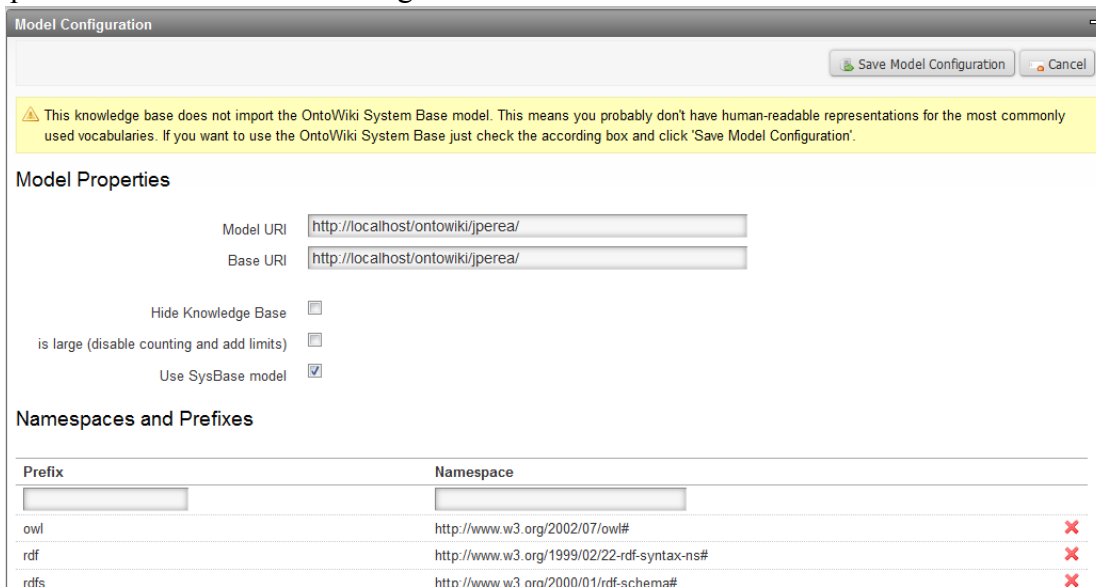


Figura 51 OntoWiki Model Configuration

Una vez realizados todos estos pasos se puede acceder a la configuración y descripción de la ontología a través del enlace que aparece en la pantalla como se puede observar a continuación.

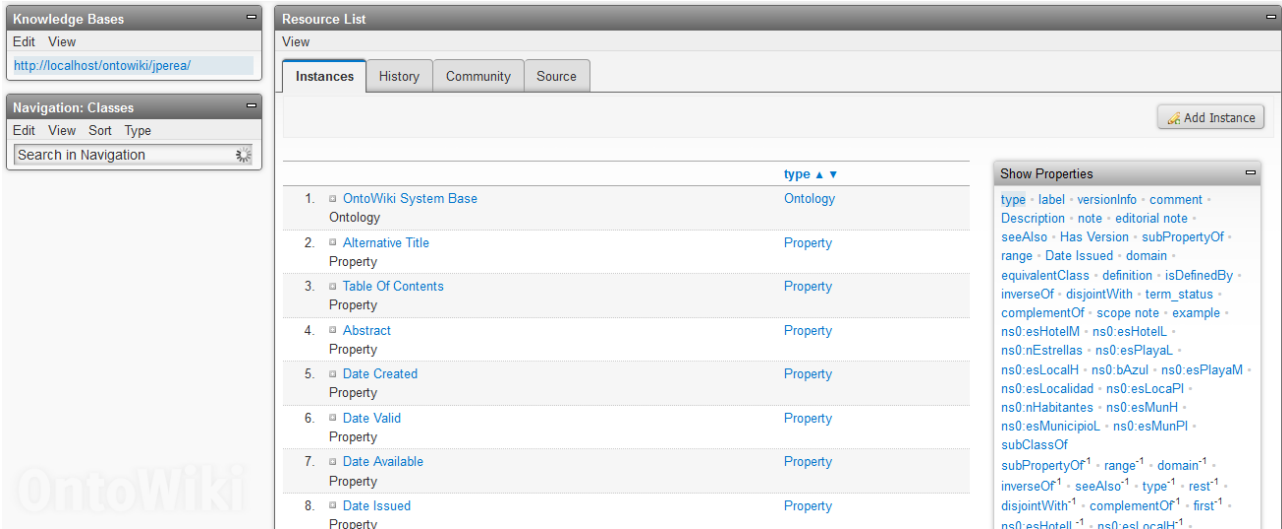


Figura 52 OntoWiki Ontología Cargada

7.5 Uso OntoWiki

Una vez importada la ontología se han realizado diferentes modificaciones sobre ella utilizando las herramientas que ofrece OntoWiki.

7.5.1 Selección de lenguajes

OntoWiki ofrece una serie de extensiones que se pueden configurar una de ellas es Select Lenguaje y viene activada por defecto. Esta extensión permite:

- Seleccionar el lenguaje de las etiquetas de los individuales que se desea visualizar.
- Añadir y/o eliminar los lenguajes creados.

Para esta ontología se han creado dos nuevos lenguajes “es” y “ca”, del siguiente modo:

- Se ha accedido a Extras → Configure Extensions

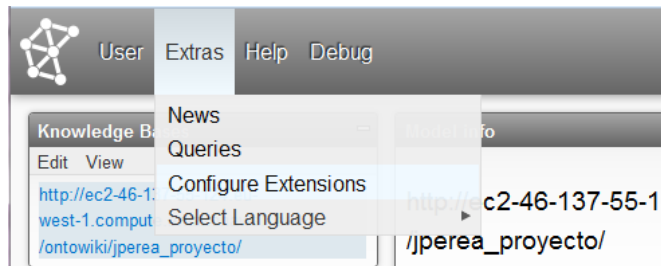


Figura 53 OntoWiki Extras

- Se accede a Select Lenguaje.

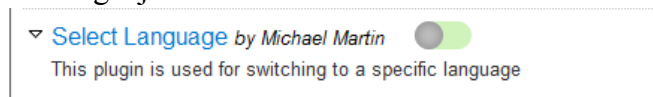


Figura 54 OntoWiki Configurar Select Lenguaje

- En la siguiente página se configuran los lenguajes que se pueden mostrar de la ontología.

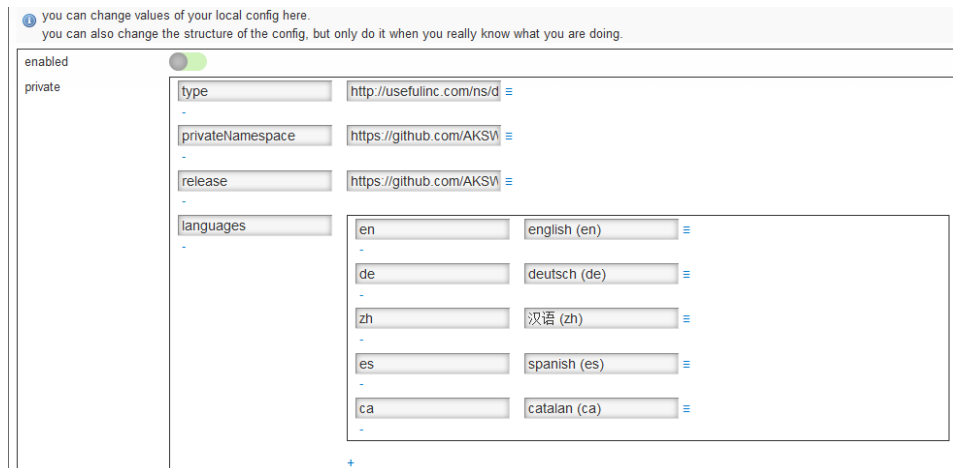


Figura 55 OntoWiki añadir lenguajes en Select Languages

Como se puede observar se han añadido nuestros propios lenguajes, dicha acción se realiza pulsando sobre el “+” y escribiendo en el primer campo el código de la label del lenguaje y en el segundo la descripción del idioma. Para eliminar un lenguaje se ha de pulsar sobre “-”, en este caso se elimina “de” y “zh”, ya que estos idiomas no se incluyen en nuestra ontología.

La configuración definitiva de los lenguajes queda como sigue:

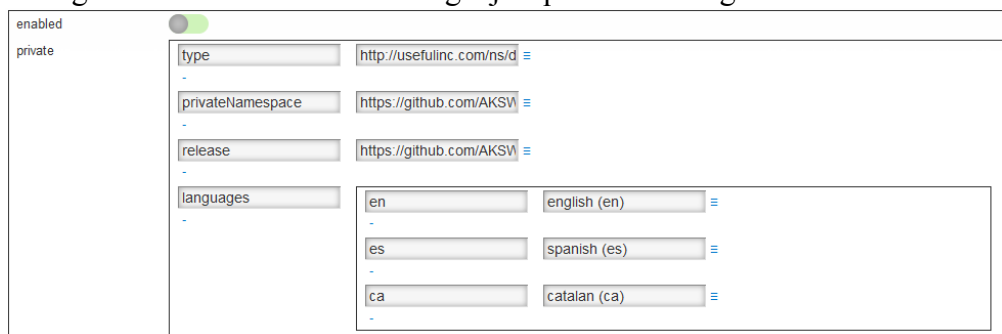


Figura 56 OntoWiki Configuración Select Languages

Nota: ver apartado 7.6

Para que esta configuración funcione correctamente tanto las clases, los individuos, relaciones y propiedades de la ontología han de tener declaradas y rellenas las etiquetas del idioma que corresponda, como se ve en el código incluido a continuación:

```
<!-- http://ec2-46-137-55-124.eu-west-1.compute.amazonaws.com/ontowiki/jperea_proyecto#Construccion
-->
<owl:Class rdf:about="http://ec2-46-137-55-124.eu-west-1.compute.amazonaws.com/ontowiki/jperea_proyecto#Construccion">
  <rdfs:label xml:lang="ca">Construcció</rdfs:label>
  <rdfs:label xml:lang="en">Construction</rdfs:label>
  <rdfs:label xml:lang="es">Construcción</rdfs:label>
  <rdfs:subClassOf rdf:resource="http://ec2-46-137-55-124.eu-west-1.compute.amazonaws.com/ontowiki/jperea_proyecto#Artefacto"/>
</owl:Class>
```

Figura 57 Código RDF clase Construcción

También se han modificado las etiquetas de las propiedades creadas para que un usuario externo

entienda el significado de estas. Las propiedades quedan como sigue:

- “Su localidad/La seva localitat”: esHotelL, esPlayaL.
- “Su municipio/El seu municipio”: esHotelM, esLocalidad, esPlayaM.
- “Sus hoteles/Els seus hotels”: esLocalH, esMunH.
- “Sus localidades/Les seves localitats”: esMunicipioL
- “Sus Playas/Les seves platjes”: esLocaPl, esMunPl.

7.5.2 Creación de consultas

La extensión Queries permite crear consultas en SPARQL, comprobar su buen funcionamiento y guardarlas para que las ejecuten usuarios externos.

Para crear una consulta se han de realizar los siguientes pasos:

- Se accede a Extras → Queries

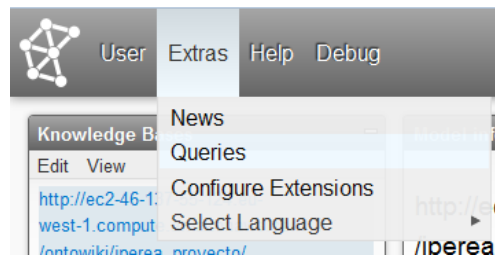


Figura 58 OntoWiki Queries

- En la pantalla siguiente aparece el *Query Editor* que permite escribir las consultas, validarlas, ejecutarlas y guardarlas.



Figura 59 OntoWiki Query Editor

- Se escribe la consulta que se desee crear, en este caso será la que sigue:

SELECT ?playa WHERE { ?subject ns0:bAzul true. ?subject rdfs:label ?playa. FILTER (lang(?playa) = "es") }

Y en el editor se refleja del siguiente modo:



Figura 60 OntoWiki Query Editor Consulta

- Para comprobar su buen funcionamiento basta con pulsar *Submit Query* y se ejecuta la consulta.

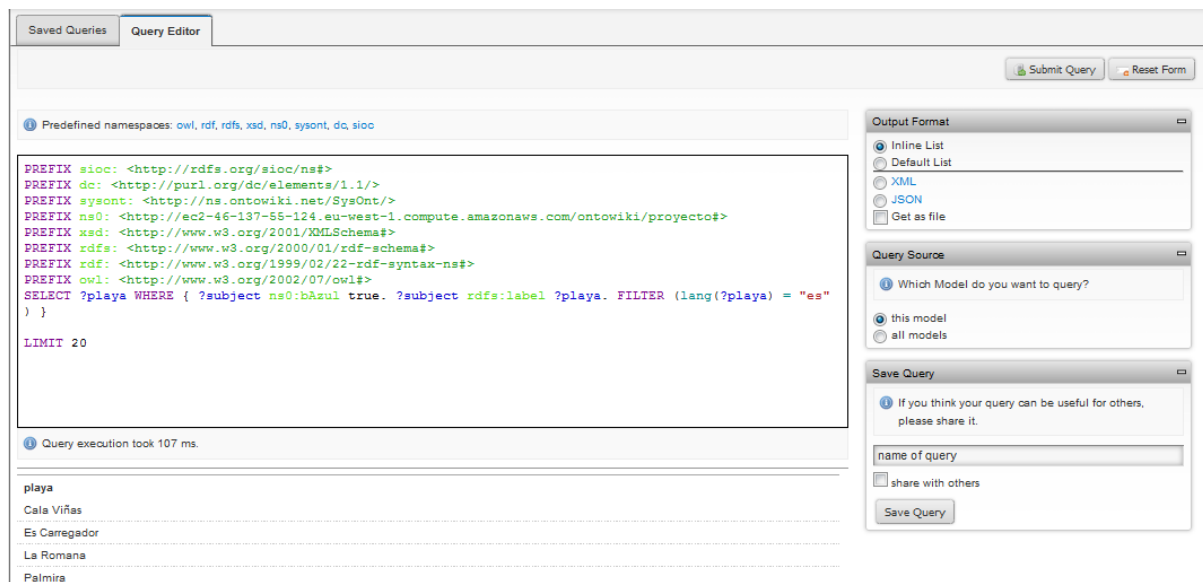


Figura 61 OntoWiki Query Editor Submit Query

Como se ve en la imagen anterior OntoWiki al ejecutar las sentencias añade *LIMIT 20* que limita el número de resultados que se muestra por pantalla, en este caso 20, pero que se puede quitar o aumentar cada vez que se realice la consulta.

- Para grabar las consultas simplemente se rellena el campo *name of query* con el nombre que se desee y se pulsa el botón *Save Query*. Esta acción devuelve un mensaje de error aunque la query si se haya guardado, tal y como se verán el las imágenes siguientes.

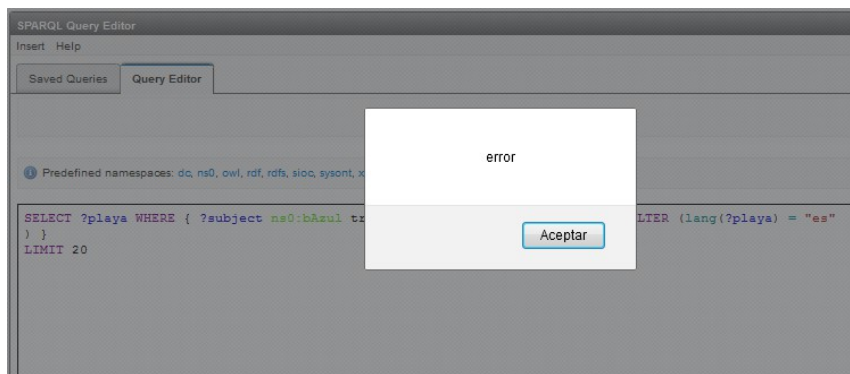


Figura 62 OntoWiki Query Editor Error Save



Figura 63 OntoWiki Query Editor Save Queries

Para visualizar las consultas ya cargadas basta con pulsar sobre el botón *Editor* que nos abrirá la misma pantalla donde se ha grabado la sentencia.

También se pueden visualizar y modificar las cuestiones grabadas accediendo desde el enlace *Query*.

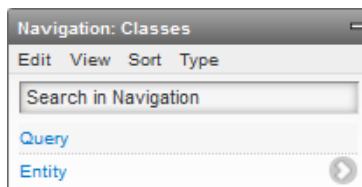


Figura 64 OntoWiki Query

El enlace anterior nos dirige a otra pantalla donde se listan las consultas.

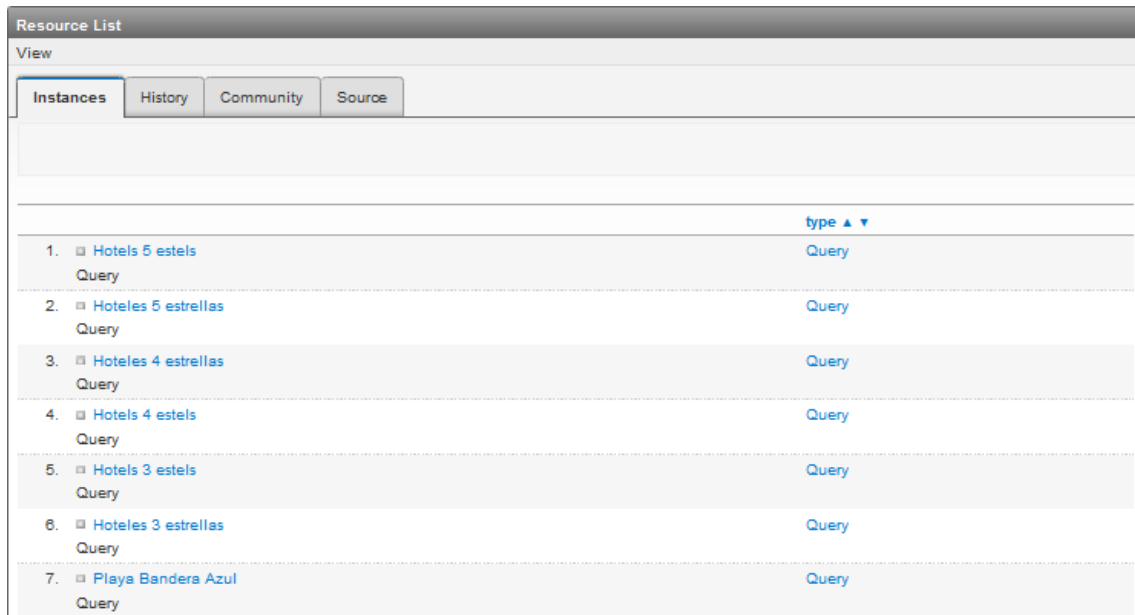


Figura 65 OntoWiki Query Instances

Desde esta ubicación pulsado sobre el nombre de la sentencia deseado se nos muestra la siguiente pantalla desde donde se puede editar, añadir y/o borrar las propiedades de la consulta seleccionada.

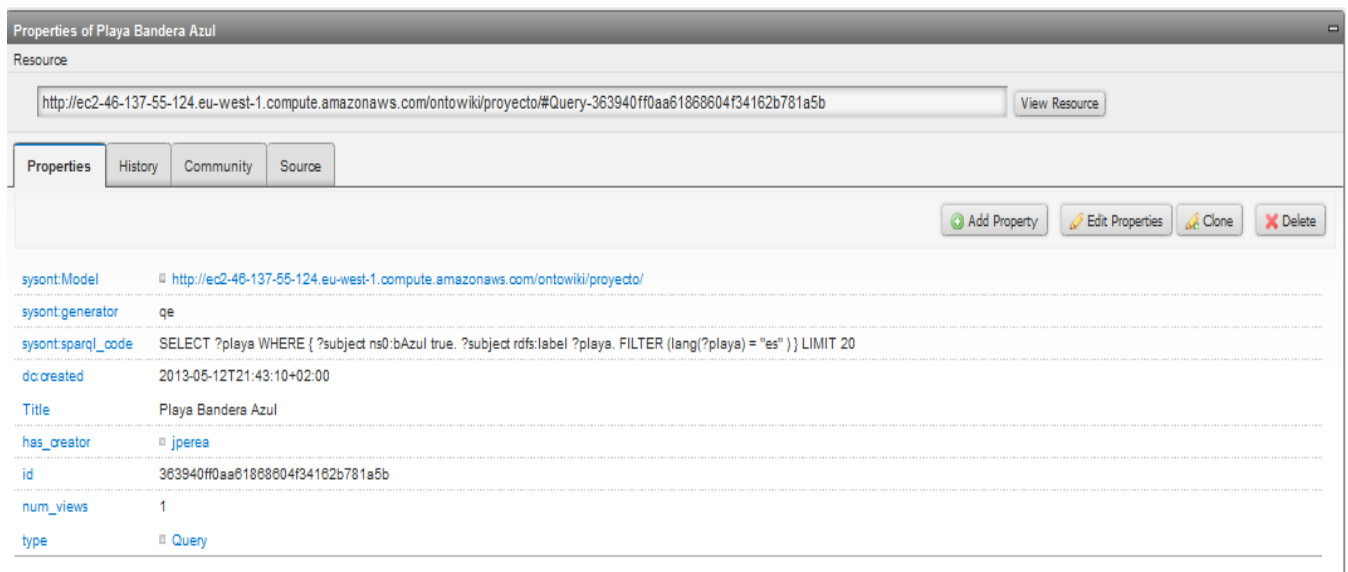


Figura 66 OntoWiki Query Properties

Las sentencias creadas para el proyecto son las siguientes:

- **Hotels 5 estels:** Recoge los hoteles de 5 estrellas y muestra la etiqueta en catalán.

```
SELECT ?Hotel WHERE { ?subject ns0:nEstrellas 5 . ?subject rdfs:label ?Hotel. FILTER (lang(?Hotel) = "ca" ) }
```

- **Hoteles 5 estrellas:** Muestra la etiqueta en castellano de los hoteles de 5 estrellas.

```
SELECT ?Hotel WHERE { ?subject ns0:nEstrellas 5 . ?subject rdfs:label ?Hotel. FILTER (lang(?Hotel) = "es" ) }
```
- **Hotels 4 estels:** Selecciona los hoteles de 4 estrellas y mostrando su etiqueta en catalán.

```
SELECT ?Hotel WHERE { ?subject ns0:nEstrellas 4 . ?subject rdfs:label ?Hotel. FILTER (lang(?Hotel) = "ca" ) }
```
- **Hoteles 4 estrellas:** Realiza la misma acción de la consulta anterior pero mostrando la etiqueta en castellano.

```
SELECT ?Hotel WHERE { ?subject ns0:nEstrellas 4 . ?subject rdfs:label ?Hotel. FILTER (lang(?Hotel) = "es" ) }
```
- **Hotels 3 estels:** Muestra la descripción en catalán los hoteles de 3 estrellas.

```
SELECT ?Hotel WHERE { ?subject ns0:nEstrellas 3 . ?subject rdfs:label ?Hotel. FILTER (lang(?Hotel) = "ca" ) }
```
- **Hoteles 3 estrellas:** Funciona como la cuestión anterior pero mostrando el nombre en castellano.

```
SELECT ?Hotel WHERE { ?subject ns0:nEstrellas 3 . ?subject rdfs:label ?Hotel. FILTER (lang(?Hotel) = "es" ) }
```
- **Platja Bandera Blava:** Selecciona las playas que tiene bandera azul mostrando la etiqueta en catalán.

```
SELECT ?playa WHERE { ?subject ns0:bAzul true. ?subject rdfs:label ?playa. FILTER (lang(?playa) = "ca" ) }
```
- **Playa Bandera Azul:** Como la anterior sentencia devuelve las playas con bandera azul presentando las etiquetas en castellano.

```
SELECT ?playa WHERE { ?subject ns0:bAzul true. ?subject rdfs:label ?playa. FILTER (lang(?playa) = "es" ) }
```

7.5.3 Etiquetas en diferentes idiomas

Como ya se ha explicado con el Protégé se han asignado etiquetas de diferentes idiomas a las clases, individuales y propiedades. Desde OntoWiki también se puede realizar esa acción con sus propias labels. Se mostrará las dos formas de hacerlo:

- Utilizar *Edit Resource* para acceder a la edición del elemento seleccionado tal y como se ve a continuación.

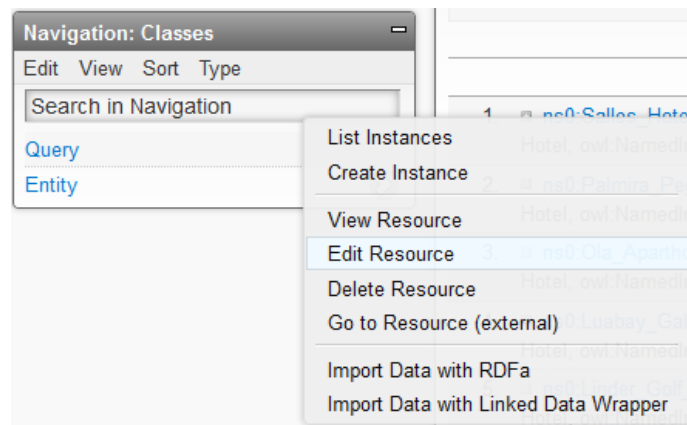


Figura 67 OntoWiki Edit Resource

Simplemente se ha de pulsar en el + y aparecerá un campo nuevo en la propiedad label donde se escribirá la etiqueta elegida, en este caso “Consultas”. Para seleccionar el idioma se ha de pulsar debajo del campo nuevo y aparece un desplegable donde se elige el lenguaje deseado.

Como se puede ver en la imagen siguiente no aparecen las siglas “ca” de catalán, por tanto hay que añadir la etiqueta de otro modo.

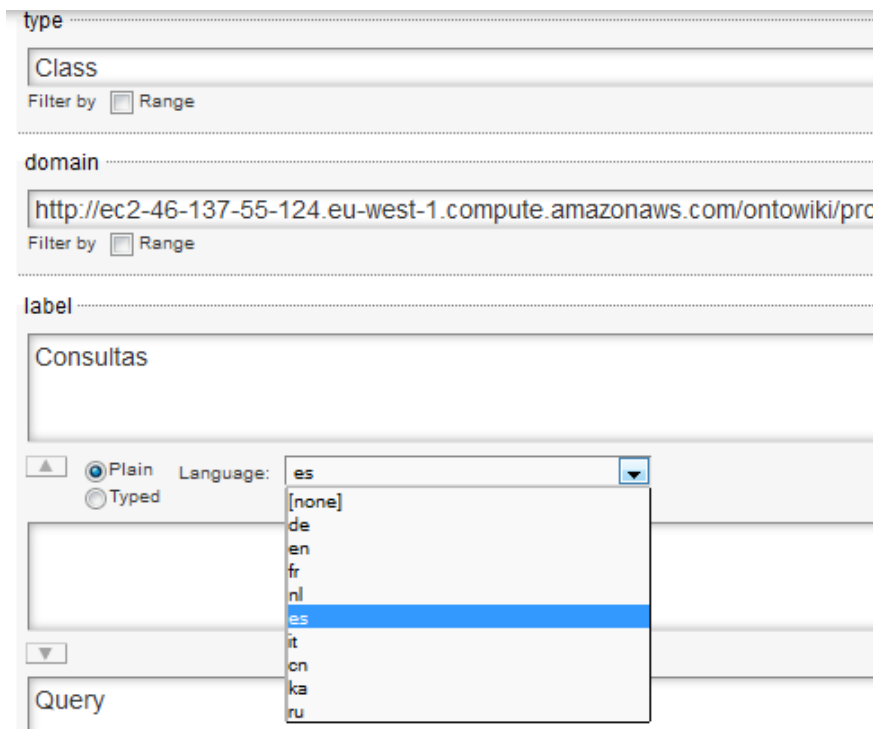


Figura 68 OntoWiki Edit Resource añadir label

- Al otro método para incluir los diferentes lenguajes en las etiquetas se accede desde *View Resource*.

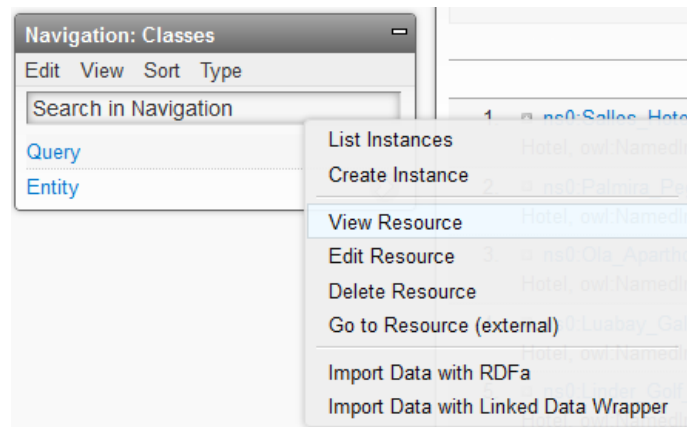


Figura 69 OntoWiki View Resource

Se accede a la pestaña *Source* y se modifica el código añadiendo “Consultes”@ca directamente en el código, tal y como se ve a continuación, y se guardan las modificaciones.

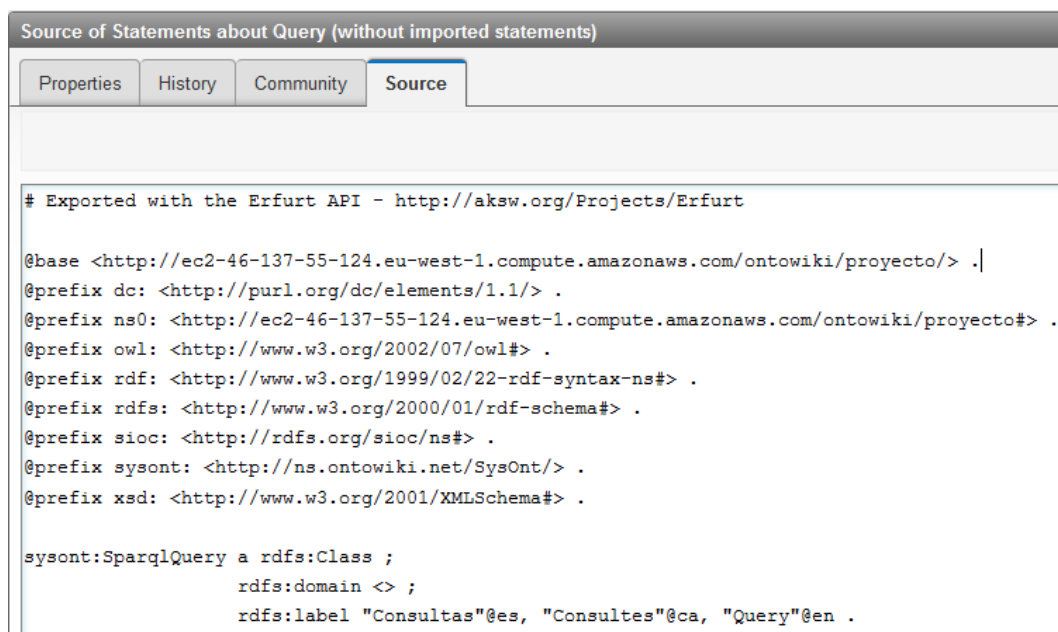


Figura 70 OntoWiki Source

7.6 Problemas Detectados

- El primer problema detectado ha sido originado por la inclusión de acentos y caracteres especiales dentro del nombre de las clases e individuos. Estos caracteres no los detecta correctamente OntoWiki. Tal y como se ve a continuación.

| | | |
|------|----------------------|-------|
| 531. | ▣ ns0:Objeto_F??sico | Class |
| | Class | |
| 532. | ▣ ns0:Pa??s | Class |
| | Class | |
| 533. | ▣ ns0:Playa | Class |
| | Class | |
| 534. | ▣ ns0:Regi??n | Class |
| | Class | |
| 535. | ▣ ns0:Secci??n | Class |
| | Class | |
| 536. | ▣ ns0:Ubicaci??n | Class |
| | Class | |
| 537. | ▣ ns0:Unidad | Class |
| | Class | |

Figura 71 OntoWiki Error caracteres especiales

Como se ve en la imagen anterior OntoWiki traduce los caracteres especiales por ?? y eso origina un error al intentar ver por ejemplo ver la descripción de una clase.

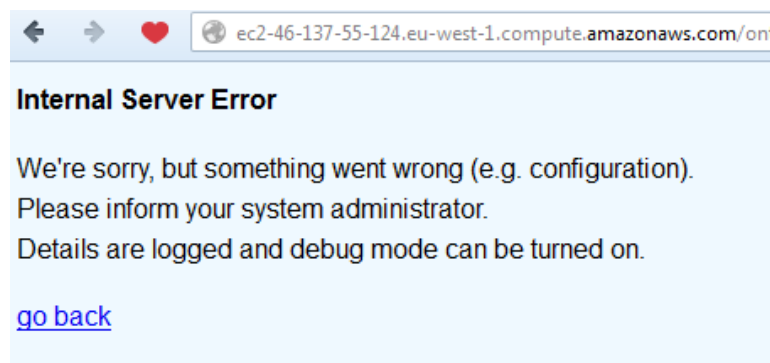


Figura 72 OntoWiki Error

La solución tomada ha sido renombrar todas las clases e incluir el atributo “label” que permite incluir los nombres de los elementos con caracteres especiales y distinguiendo entre idiomas.

- El siguiente problema detectado ha sido la URI que se ha de incluir dentro de la configuración de la ontología, tal y como se ha visto en el apartado 7.3. La que inicialmente se incluyó <http://localhost/ontowiki/jperea/> no permite el acceso por parte de un usuario externo a la ontología por ello se ha cambiado la URI por la siguiente,

<http://ec2-46-137-55-124.eu-west-1.compute.amazonaws.com/ontowiki/proyecto/>

Esto soluciona el acceso a la ontología a los usuarios externos.

- Se generó un error al crear la etiqueta de “cat” para el idioma catalán desde la extensión SelectLenguaje. En el log se nos informaba de los siguiente:

```
013-05-05T21:55:14+02:00 EMERG (0): Zend_Translate_Exception: The given Language (cat) does not exist -> /usr/share/php/libzend-framework-php/Zend/Translate/Adapter.php@433
```

```
2013-05-05T21:55:21+02:00 EMERG (0): Zend_Translate_Exception: The given Language (cat) does not exist -> /usr/share/php/libzend-framework-php/Zend/Translate/Adapter.php@433
2013-05-05T21:59:40+02:00 EMERG (0): OntoWiki_Exception: Entry key must be string. -> /usr/share/ontowiki/application/classes/OntoWiki/Menu.php@40
```

Para solucionar el problema se ha accedido a:
/usr/share/ontowiki/extensions selectlenguaje.ini

se ha borrado la línea

languages.cat.cat = “catalan”

También en nuestra ontología se cambió la etiqueta “cat” por “ca” que es la admitida por Ontowiki.

Tal y como se ha comentado en el apartado anterior desde Extras → Configure Extensions → Select Lenguaje se ha intentado borrar los lenguajes que no se van a usar “de” y “zh” pero no ha sido posible. Ya que aunque el sistema no da ningún error al borrarlos y al grabar la tarea realizada las etiquetas no desaparecen.

Se han eliminado manualmente en el fichero doap.n3 usr/share/ontowiki/selectlenguaje del modo siguiente, se han eliminado las líneas :de “deutsch (de)” ; zh “--(zh)”; y el ; que sigue a :en “english (en)”.

Finalmente la la selección de lenguajes queda como sigue:



Figura 73 OntoWiki Select Lenguaje Correcto

- Se ha intentado cambiar la apariencia de la web original de OntoWiki para que el usuario externo que accediera a la ontología la web fuera más accesible. Para ello se han seguido los tutoriales siguientes:

- <https://github.com/AKSW/site.ontowiki/wiki/Setup-Example-Site>
- <https://github.com/AKSW/site.ontowiki/wiki/Setup-new-Site>

No se ha obtenido ningún resultado satisfactorio. Por ello se no se ha reflejado estas modificaciones en la ontología. Creo que el problema esencial ha sido que para el cambio de imagen en la página web se ha de crear los ficheros necesarios como los css o html, o transformar los ya existentes, pero no se ha realizado al no tener suficiente tiempo y no ser el objetivo final del proyecto.

- El último problema detectado a sido al intentar activar algunas extensiones que la mayoría

fallan y tiran abajo la aplicación. Por ejemplo, se activó la extensión Saved Queries que permite mostrar las consultas guardadas en un módulo y poder ejecutarlas.

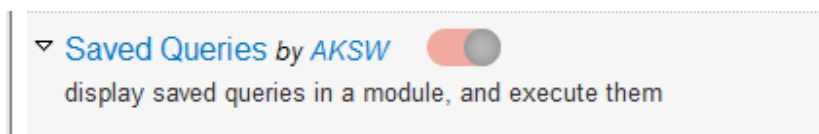


Figura 74 OntoWiki Saved Queries

Al activarlo volvemos a tener el mensaje de error comentado en el primer problema de este apartado y para solucionarlo se ha tenido que acceder al servidor internamente y volverlo a ponerlo en estado Off. Por tanto únicamente se pueden ejecutar las queries desde la sección Queries del menú de OntoWiki.

8. Conclusiones

Durante el desarrollo del proyecto se ha llegado a una serie de conclusiones generales:

- Se dispone del lenguaje de desarrollo necesario para implementar ontologías, ya que desde W3C los han ido proporcionando.
- Existe el software necesario y lo suficientemente ágil para que lo utilice cualquier usuario que permite el diseño de cualquier tipo de ontologías.
- Las wikis semánticas que se han consultado están en proceso de desarrollo o experimental. Entre ellas no existe homogeneidad entre los lenguajes aceptados de una ontología ni en las acciones que se pueden realizar en ellas, se ha hecho en falta un poco de estandarización.

Respecto a la OntoWiki se ha hecho en falta manuales para el usuario, ya que los tutoriales son difíciles de encontrar y no existe un índice de estos. Tampoco se ha encontrado un manual para el usuario externo o que acceda a consultar una ontología y por ello se ha añadido en este documento en el Anexo I. Sí se dispone un foro de discusión que ha sido de gran ayuda para el desarrollo de este proyecto ya que se ha consultado varias veces. Esto da la imagen que todo el entorno de OntoWiki refleja un estado experimental y que por tanto que queda mucho por desarrollar y crear, de hecho existe una gran cantidad de ampliaciones desarrolladas por diferentes usuarios que se pueden instalar en la aplicación, pero hay que tener bastante cuidado al realizarlo ya que algunas no están suficientemente testeadas y pueden tirar el sistema.

Por lo desarrollado para el proyecto, los pasos siguientes a seguir sería:

- Conseguir un entorno más amigable para el usuario externo, ya que el entorno de OntoWiki se considera bastante complejo de utilizar.
- Con el punto anterior se intentaría solucionar la aparición de las descripciones en todos los idiomas a la vez. Ya que si el usuario elige un solo idioma todo debería aparecer en ese único idioma.
- Ampliar la ontología, incluyendo más establecimientos de hostelería.

En conclusión en la implementación de ontologías en la web tanto en wikis semánticas como en motores de búsquedas queda mucho por hacer y habrá que potenciar sus beneficios para que los desarrolladores no experimentales utilicen todo su potencial.

Glosario

Consulta(*Query*): es una forma de pregunta, en una línea de investigación.

Diagrama de Gantt: Diagrama de barras desarrollado por L.Ganntt que muestra la planificación de temporal tareas de un proyecto.

DTD (*document type definition*) : es una descripción de estructura y sintaxis de un documento XML o SGML. Su función básica es la descripción de la estructura de datos, para usar una estructura común y mantener la consistencia entre todos los documentos que utilicen la misma DTD. De esta forma, dichos documentos pueden ser validados, conocen la estructura de los elementos y la descripción de los datos que trae consigo cada documento, y pueden además compartir la misma descripción y forma de validación dentro de un grupo de trabajo que usa el mismo tipo de información.

Espacio de nombres XML: Un espacio de nombres XML es una recomendación W3C para proporcionar elementos y atributos con nombre único en una instancia XML. Una instancia XML puede contener nombres de elementos o atributos procedentes de más de un vocabulario XML. Si a cada uno de estos vocabularios se le da un espacio de nombres, un ámbito semántico propio, referenciado a una URI donde se listen los términos que incluye, se resuelve la ambigüedad existente entre elementos o atributos que se llamen igual, la homonimia. Los nombres de elementos dentro de un espacio de nombres deben ser únicos.

Framework: define, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio, y provee una estructura y una especial metodología de trabajo, la cual extiende o utiliza las aplicaciones del dominio.

HTML (*HyperText Markup Language*): hace referencia al lenguaje de marcado predominante para la elaboración de páginas web que se utiliza para describir y traducir la estructura y la información en forma de texto, así como para complementar el texto con objetos tales como imágenes.

Metadato: son datos que describen otros datos. En general, un grupo de metadatos se refiere a un grupo de datos, llamado recurso. Este concepto es análogo al uso de índices para localizar objetos en vez de datos. Para varios campos de la informática, como la recuperación de información o la web semántica, los metadatos en etiquetas son un enfoque importante para construir un puente sobre el intervalo semántico.

Ontología: Este término en informática hace referencia a la formulación de un exhaustivo y riguroso esquema conceptual dentro de uno o varios dominios dados; con la finalidad de facilitar la comunicación y el intercambio de información entre diferentes sistemas y entidades.

Un uso común tecnológico actual del concepto de ontología, se encuentra en la inteligencia artificial y la representación del conocimiento.

OWL (*Web Ontology Language*): es un lenguaje de marcado para publicar y compartir datos usando ontologías en la WWW. OWL tiene como objetivo facilitar un modelo de marcado

construido sobre RDF y codificado en XML.

RDF (*Resource Description Framework*): es un framework para metadatos en la World Wide Web (WWW), desarrollado por el World Wide Web Consortium (W3C).

Es un lenguaje de objetivo general para representar la información en la web (un metadato data model). Es una descripción conceptual.

RDF-Schema (*RDFS o RDF Schema o Esquema RDF*): es una extensión semántica de RDF. Un lenguaje primitivo de ontologías que proporciona los elementos básicos para la descripción de vocabularios.

SPARQL (*SPARQL Protocol and [RDF Query Language](#)*): se trata de un lenguaje estandarizado para la consulta de grafos RDF, normalizado por el RDF Data Access Working Group (DAWG) del World Wide Web Consortium (W3C). Es una tecnología clave en el desarrollo de la Web Semántica que se constituyó como Recomendación oficial del W3C el 15 de Enero de 2008.

URI (*Uniform Resource Identifier*): es una cadena de caracteres corta que identifica inequívocamente un recurso (servicio, página, documento, dirección de correo electrónico, enciclopedia, etc.). Normalmente estos recursos son accesibles en una red o sistema.

URL (*Uniform Resource Locator*): es una secuencia de caracteres, de acuerdo a un formato modélico y estándar, que se usa para nombrar recursos en Internet para su localización o identificación, como por ejemplo documentos textuales, imágenes, vídeos, presentaciones digitales, etc.

Wiki: Un wiki o una wiki es un sitio web cuyas páginas pueden ser editadas por múltiples voluntarios a través del navegador web. Los usuarios pueden crear, modificar o borrar un mismo texto que comparten.

Wiki semántica: es una wiki que tiene un modelo de conocimiento subyacente descrito en sus páginas. Las wikis regulares, o sintácticas, tienen texto estructurado y hipervínculos sin tipo. Las wikis semánticas, por el contrario, ofrecen la posibilidad de capturar o identificar información acerca de los datos dentro de las páginas y las relaciones entre las páginas, de modo que pueden ser consultados o exportados como una base de datos.

XML (*eXtensible Markup Language*): es un lenguaje de marcas desarrollado por el World Wide Web Consortium (W3C). A diferencia de otros lenguajes, XML da soporte a bases de datos, siendo útil cuando varias aplicaciones se deben comunicar entre sí o integrar información.

XML Schema: es un lenguaje de esquema utilizado para describir la estructura y las restricciones de los contenidos de los documentos XML de una forma muy precisa, más allá de las normas sintácticas impuestas por el propio lenguaje XML.

Bibliografía

- [1] Extensible Markup Language (XML), <http://www.w3.org/XML/> , 13 Marzo 2013
- [2] ISO 8879:1986 Information processing -- Text and office systems -- Standard Generalized Markup Language (SGML), http://www.iso.org/iso/catalogue_detail?csnumber=16387 , 13 Marzo 2013
- [3] W3C Recommendation, Extensible Markup Language (XML) 1.0 (Fifth Edition) <http://www.w3.org/TR/REC-xml/> , 13 Marzo 2013
- [4] W3C Consortium, <http://www.w3.org/> , 13 Marzo 2013
- [5] XML Schema Part 0: Primer Second Edition, <http://www.w3.org/TR/xmlschema-0/> , W3C Recommendation , 15 Marzo 2013
- [6] XML Schema Part 1: Structures Second Edition, <http://www.w3.org/TR/2004/REC-xmlschema-1-20041028/structures.html> , W3C Recommendation , 15 Marzo 2013
- [7] Resource Description Framework (RDF), <http://www.w3.org/RDF> , 17 Marzo 2013
- [8] RDF Primer, <http://www.w3.org/TR/rdf-primer/> , W3C Recommendation, 17 Marzo 2013
- [9] Resource Description Framework (RDF): Concepts and Abstract Syntax, <http://www.w3.org/TR/rdf-concepts/> , W3C Recommendation , 17 Marzo 2013
- [10] Naming and Addressing: URIs, URLs, ..., <http://www.w3.org/Addressing/> , W3C Recommendation , 17 Marzo 2013
- [11] RDF/XML Syntax Specification (Revised), <http://www.w3.org/TR/rdf-syntax-grammar/> , W3C Recommendation , 18 Marzo 2013
- [12] RDF Vocabulary Description Language 1.0: RDF Schema, <http://www.w3.org/TR/rdf-schema/> , W3C Recommendation, 18 Marzo 2013
- [13] SPARQL Lenguaje de consulta para RDF, <http://skos.um.es/TR/rdf-sparql-query/> , Recomendación del W3C, 21 Marzo 2013
- [14] El Consorcio World Wide Web publica las Recomendaciones RDF y OWL, <http://www.w3c-es.org/Prensa/2004/nota040210.html>
- [15] OWL Web Ontology Language Guide, <http://www.w3.org/TR/owl-guide/> , W3C Recommendation, 23 Marzo 2013
- [16] Ontología (informática), http://es.wikipedia.org/wiki/Ontolog%C3%ADa_%28inform%C3%A1tica%29 , 25 Marzo 2013
- [17] Ontology Development 101: A Guide to Creating Your First Ontology, http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html , 25 Marzo 2013
- [18] WordNet, <http://wordnet.princeton.edu/wordnet/> , 27 Marzo 2013
- [19] Euro WordNet, <http://www.illc.uva.nl/EuroWordNet/> , 27 Marzo 2013
- [20] Open Multilingual Wordnet, <http://www.casta-net.jp/~kuribayashi/multi/> , 28 Marzo 2013

- [21] SUMO, http://en.wikipedia.org/wiki/Suggested_Upper_Merged_Ontology, 28 Marzo 2013
- [22] Protégé, <http://protege.stanford.edu/>, 29 Marzo 2013
- [23] Manual Protégé http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/resources/ProtegeOWLTutorialP4_v1_3.pdf, 29 Marzo 2013
- [24] WordNet on line, <http://wordnetweb.princeton.edu/perl/webwn>, 30 Marzo 2013
- [25] Wiki Semántica, http://es.wikipedia.org/wiki/Wiki_sem%C3%A1ntica
- [26] Semantic Mediawiki, <http://semantic-mediawiki.org/>, 16 Abril 2013
- [27] OntoWiki, <http://ontowiki.eu/Welcome>, 17 Abril 2013

Webs Consultadas

- **Proyectos publicados UOC:**
 - <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/385/1/36605tfc.pdf>
 - <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/493/1/27706tfc.pdf>
 - <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/8138/1/jcabezascTFC0611.pdf>
 - <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/10741/9/sfernandezveTFC1211.pdf>
 - <http://openaccess.uoc.edu/webapps/o2/bitstream/10609/955/1/36607tfc.pdf>
- **Ayuntamiento Calvià**, <http://www.calvia.com>
- **Tripadvisor**, <http://www.tripadvisor.es>
- **Una página web**, <http://www.unapaginaweb.org/tag/wiki-semantica/>
- Soren Auer; Sebastian Dietzold; Jens Lehmann; Thomas Riechert, **OntoWiki-A Tool for Social, Semantic Collaboration**, http://www2007.org/workshops/paper_91.pdf, (2006)
- <http://www.youtube.com/watch?v=vP1UDKeZsQk>
- **Grupo discusión OntoWiki**, <https://groups.google.com/forum/?hl=es&fromgroups=#!forum/ontowiki-user>
- **Páginas OntoWiki:**
 - <https://github.com/AKSW/OntoWiki/wiki/GetOntowikiUsers>
 - <https://github.com/AKSW/OntoWiki/wiki/Setup>
 - <https://github.com/AKSW/OntoWiki/wiki/Getting-Started-Users>
 - <https://github.com/AKSW/site.ontowiki/wiki/Setup>
 - <https://github.com/AKSW/site.ontowiki/wiki/Setup-Example-Site>
- **Definiciones del Glosario:**
 - **DTD**, http://es.wikipedia.org/wiki/Definici%C3%B3n_de_tipo_de_documento
 - **HML**, <http://es.wikipedia.org/wiki/HTML>
 - **XML**, http://es.wikipedia.org/wiki/Extensible_Markup_Language
 - **Espacio de Nombres**, http://es.wikipedia.org/wiki/Espacio_de_nombres_XML
 - **Ontología**, http://es.wikipedia.org/wiki/Ontolog%C3%ADa_%28inform%C3%A1tica%29
 - **OWL**, <http://es.wikipedia.org/wiki/OWL>
 - **RDF**, http://es.wikipedia.org/wiki/Resource_Description_Framework
 - **Wiki Semántica**, http://es.wikipedia.org/wiki/Wiki_sem%C3%A1tica
 - **Wiki**, <http://es.wikipedia.org/wiki/Wiki>

- **Consulta**, <http://en.wikipedia.org/wiki/Query>
- **RDF-Schema**, http://es.wikipedia.org/wiki/RDF_Schema
- **XML-Schema**, https://es.wikipedia.org/wiki/XML_Schema
- **SPARQL**, <http://es.wikipedia.org/wiki/SPARQL>
- **URI**, http://es.wikipedia.org/wiki/Uniform_Resource_Identifier
- **URL**, http://es.wikipedia.org/wiki/Localizador_uniforme_de_recursos