



9-5-2013

PRÁCTICA 5

**Marcado semántico y
Recuperación de Información
en Videojuegos.**

**Álvaro Pérez Liaño
Xavier Gallofré Nieva
Raúl Vicente Bueno Sevilla**
ISBC 2012/2013

Índice

Direcciones de Correo.	2
Diseño y formalización en OWL de una ontología para representar conocimiento sobre fotografías de videojuegos.	2
Idea original.....	2
Evolución de la ontología	3
Etiquetación en la ontología.	7
Consultas sobre la ontología.....	7
Anotación y recuperación semántica de imágenes de videojuegos.....	8
Descripción del conjunto de imágenes.....	8
Descripción del proceso de anotación	8
La herramienta y el proceso de anotación	10
Herramienta de recuperación.....	13
Panel de Archivo	16
Búsqueda semántica y búsqueda sintáctica	16
Trabajo de consulta y documentación. Relación con otras herramientas.....	17
AKTiveMedia: http://www.dcs.shef.ac.uk/~ajay/html/cresearch.html	17
FlickrR: www.flickr.com	17
Facebook (solo la gestión de fotos)	
http://www.facebook.com/sitetour/photos_notes.php	17
iPhoto http://www.apple.com/ilife/iphoto/what-is-iphoto.html	18
ESP Game. Labeling the web. http://www.espgame.org/	18
SMORE http://owl.mindswap.org/downloads/	18
PhotoStuff - http://www.mindswap.org/2003/PhotoStuff/	19
FotoTagger http://www.fototagger.com/	19
Caliph & Emir http://www.semanticmetadata.net/features/	19
Grupo del trabajo	20
Conclusiones	21
Explica si crees que se han cumplido los objetivos docentes planteados (puedes valorarlo del 1 al 5):.....	22
Opinión de la práctica	23

Direcciones de Correo.

Álvaro Pérez Liaño: alvaro.perez@ucm.es
Raúl Vicente Bueno Sevilla: rvbueno@ucm.es
Xavier Gallofé Nieva: xavier.gallofre@gmail.com

Diseño y formalización en OWL de una ontología para representar conocimiento sobre fotografías de videojuegos.

Idea original

La adquisición del conocimiento de la práctica no fue en absoluto complicada, ya que los videojuegos son un tema que nos gusta y que conocemos muy bien.

La idea de trabajar con fotos de videojuegos fue para hacer la práctica más interesante y poder intentar buscar propiedades que en fotos turísticas no podríamos encontrar, como rivalidades entre personajes o escenarios más variopintos.

La estructura que decidimos dar a la ontología intenta ser muy simple, sobretudo porque lo que queremos es dar al usuario la información necesaria, sin agobiarle con demasiadas clases o relaciones muy complejas que pudieran llevar a equivocación. También nos ha parecido que era muy útil que el usuario sólo pudiera etiquetar individuos de clases muy generales, para que la ontología fuera la encargada de decidir en qué subclases inferirlo.

Inicialmente la ontología tenía la siguiente cantidad de datos:

Nº de conceptos primitivos:.....	19
Nº de conceptos definidos:	29
Nº de individuos:	27
Nº de relaciones y atributos:	11

A medida que hemos ido evolucionándola, hemos llegado a muchos más conceptos definidos y menos primitivos, muchas clases primitivas que teníamos no eran útiles para inferir información así que las eliminamos.

Nº de conceptos primitivos:.....	11
Nº de conceptos definidos:	34
Nº de individuos:	114
Nº de relaciones y atributos:	14

Evolución de la ontología

La ontología inicialmente se dividía en **Fotos** y **Contenido**, pero no tardamos en darnos cuenta que nos era muy útil añadir una tercera clase: **Atributos**. Esta última clase nos servía a modo de clase interna, de este modo podíamos asignar instancias concretas los objetos, lugares, personajes.

La estructura “raíz” por tanto, es la siguiente:



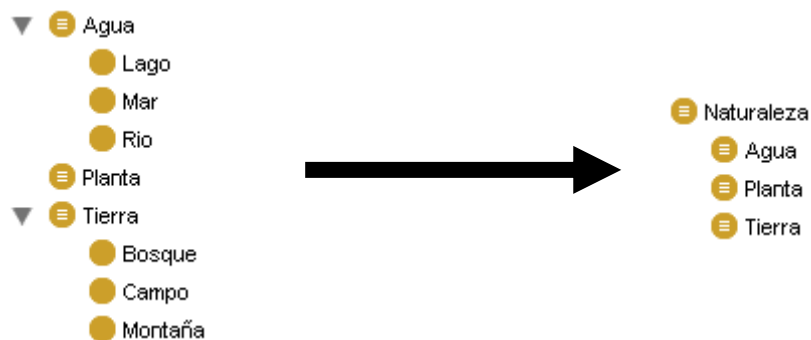
El contenido se divide en las categorías que hemos considerado más significativas en una foto de un videojuego, que son las **estructuras** que puedan aparecer, la **naturaleza** involucrada, los **objetos** que los personajes lleven y los propios **personajes**. Estas cuatro categorías son aquellas que mostramos al usuario para poder crear individuos.



Las **Estructuras** se dividen en **Edificios** y **Monumentos**, pero como se puede ver más abajo, hemos ido retocando mucho las subclases que estaban en cada una de ellas. Al principio nos costó darnos cuenta que si no creábamos las clases como definidas, no se nos iban a etiquetar los individuos de las estructuras, pero finalmente arreglamos el fallo.



La clase **Naturaleza** ha sufrido muy pocos cambios, únicamente la usamos para indicar entornos acuáticos, terrestres o la vegetación. Al igual que con las estructuras tuvimos que definir las subclases para que se infirieran bien los individuos.



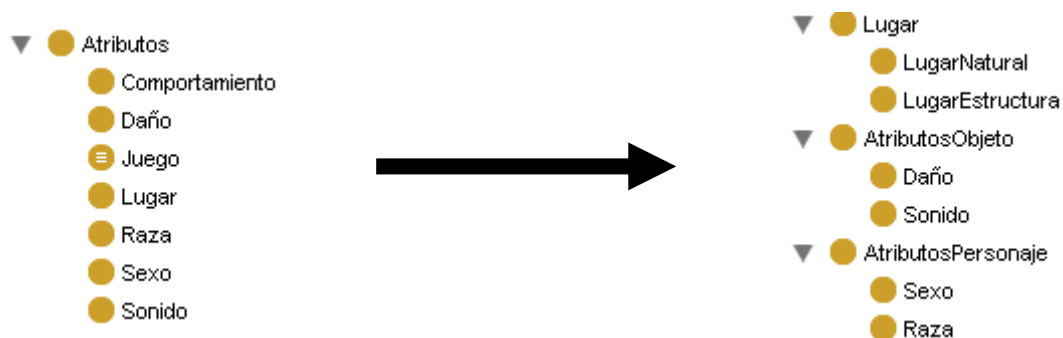
Los **Personajes** y los **Objetos** están íntimamente ligados, dado que tienen una relación inversa (*un personaje puede usar un objeto y un objeto sólo es objeto si es usado por un personaje*). En estos dos grupos de clases fue donde nos dimos cuenta de que teníamos que definir bien las subclases para poder tener una inferencia que especializara los individuos.



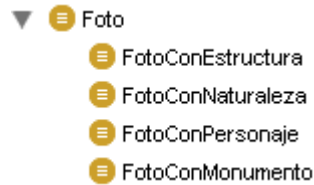
La clase de los **Atributos** únicamente refleja propiedades que tienen el resto de clases, de este modo conseguimos mucha más especialización en las propiedades, creando propiedades que actúen sobre individuos concretos.

En esta clase no teníamos que preocuparnos de crear subclases definidas. Como se puede apreciar en la imagen de debajo, la clase **Juego** era una clase definida, porque tenía una relación inversa de las fotos en las que aparecía dicho juego. Finalmente decidimos quitar esta propiedad por no aportar ningún tipo de información útil.

Esta clase ha sufrido tres especializaciones en sus subclases para permitir hacer propiedades sobre ellas más concretas.

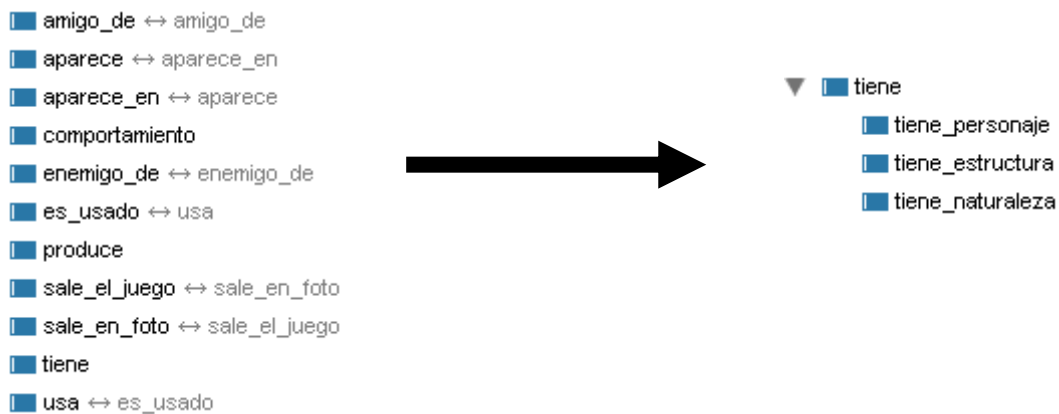


La clase **Foto** es la más simple de todas. No necesitábamos crear algo realmente complejo, lo importante era conseguir que en la aplicación el usuario decidiera usar las herramientas de recuperación en vez de buscar con el árbol las fotos.

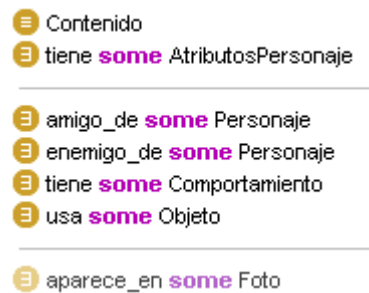


Las propiedades que hemos decidido poner no son demasiadas, no nos parecía que para una herramienta de etiquetación de fotos fueran necesarias más propiedades o más complejas. Con las que tenemos, podemos sacar bastante juego en las consultas, sin llegar a confundir al usuario.

La única variación que hicimos en las propiedades, fue especializar la propiedad **tiene**, para que quedara más intuitiva.



El concepto más representativo de la Ontología es la clase Personaje, que en el fondo es el tema central de cada juego. Su definición sería la siguiente:



Aquí podemos ver que un personaje tiene obligatoriamente algún atributo de Personaje, puede tener relaciones con otros personajes, puede usar objetos y además puede tener un comportamiento.

Etiquetación en la ontología.

Las etiquetas siempre van a ser sobre los cuatro conceptos principales de Contenido: **Estructura**, **Naturaleza**, **Objeto** y **Personaje**, así que las consultas siempre van a sacar partido del razonador, que en función de cuanto haya rellenado el usuario cada instancia lo ubicará en una categoría u otra.

Consultas sobre la ontología

Las consultas desde Protegé usando únicamente la ontología, se centran en especializar individuos con la inferencia, saber que **Contenido** aparece en una fotografía o ver si alguna fotografía cumple algunas propiedades concretas.

Como hemos dicho anteriormente, este tipo de consultas no nos parecían muy relevantes, ya que habiendo hecho los dos ejercicios anteriores de Protegé no íbamos a sacar nada que fuera muy diferente. Las consultas realmente interesantes son las sintácticas y semánticas que explicaremos más adelante.

Anotación y recuperación semántica de imágenes de videojuegos

Descripción del conjunto de imágenes.

La aplicación viene con cuarenta imágenes, no todas ellas completamente etiquetadas, pero si cargadas en la ontología.

Hemos cogido fotos de cuatro juegos distintos, que nos han parecido los suficientemente diferentes como para cubrir un gran margen de posibilidades de etiquetación y consulta.

Descripción del proceso de anotación

Para poder realizar una anotación correcta, sería ideal tener un mínimo conocimiento sobre los juegos, pero esto solo afectaría a los nombres que se les asignen a los personajes, por lo que el impacto en la aplicación es mínimo.

Cada vez que se etiqueta un individuo, se puede elegir entre cuatro clases para ubicarlo: **Personaje**, **Estructura**, **Objeto** y **Naturaleza**. Dependiendo donde lo posicionemos se mostraran unas propiedades u otras para rellenar.

Internamente la ontología se encarga de especializar lo máximo posible a los individuos a medida que va teniendo más información sobre ellos, así podemos ejecutar consultas cada vez más complejas. Esto es un compromiso entre el usuario y la ontología, mientras más se “individualice” cada individuo mejores consultas se podrán hacer sobre él en el futuro.

En cuanto a las propiedades de la ontología, las relaciones de *amigo* y *enemigo* son simétricas. Las relaciones de *usa* y *es usado* son inversas y *tiene* posee tres sub relaciones. Este tipo de relaciones más complejas son muy útiles

para no tener al usuario etiquetando constantemente cosas que se pueden deducir solas.

Hay ciertas propiedades, como son *comportamiento* o *tiene* (con sus sub propiedades), que actúan sobre un individuo de la forma: “*Link tiene sexo masculino*”, que en OWL se traduciría a: “*Link tiene_personaje A_Masculino*”. Este tipo de propiedades son complejas de sacar en una consulta, por lo que para ello hemos creado clases definidas como “Elfo”, “Humano”, etc...

No hemos creado todas las clases que infieran este tipo de propiedades para hacer hincapié en lo importante que es que la ontología esté bien definida, ya que si no se define bien la ontología el usuario no podrá ejecutar consultas tan simples como: “Personajes Femeninos”.

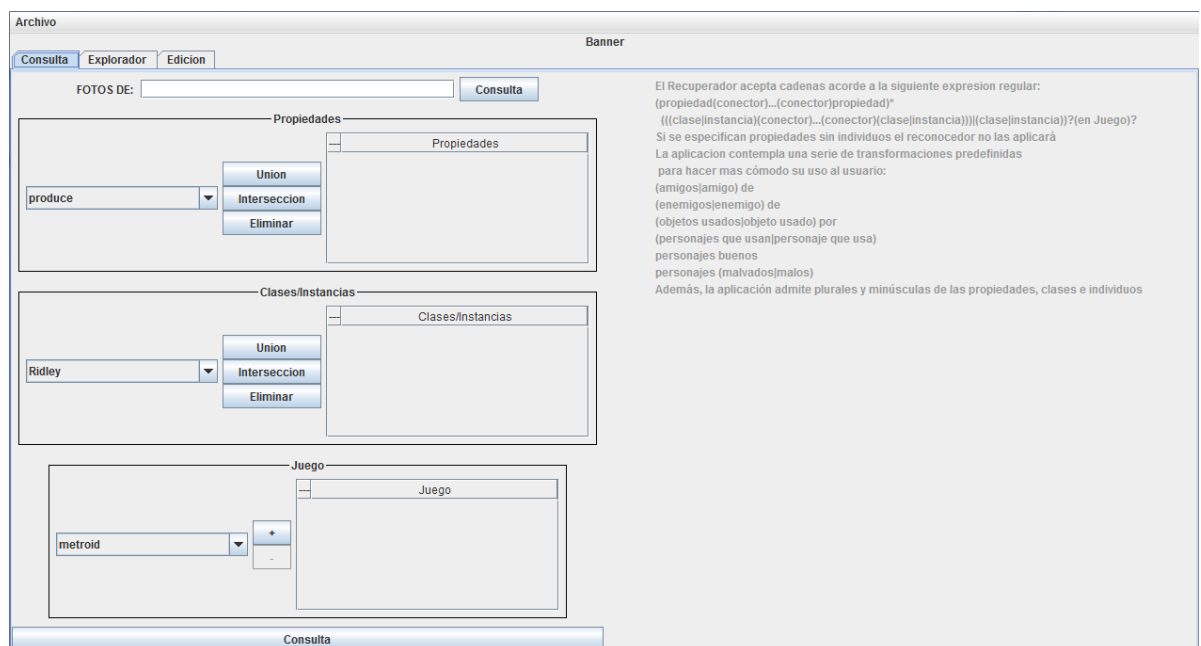
Cada individuo de foto se representa en la ontología con un **datatype** con su url local, el **Contenido** que tiene y que **Juego** o Juegos aparecen.

La herramienta y el proceso de anotación

En el arranque de la aplicación todas las fotografías que estén en el directorio “*fotos*” se etiquetarán automáticamente (si no lo están ya) y se les asignará en el datatype *urlfoto* su dirección global dentro de la máquina. Si no existe este directorio se crea.

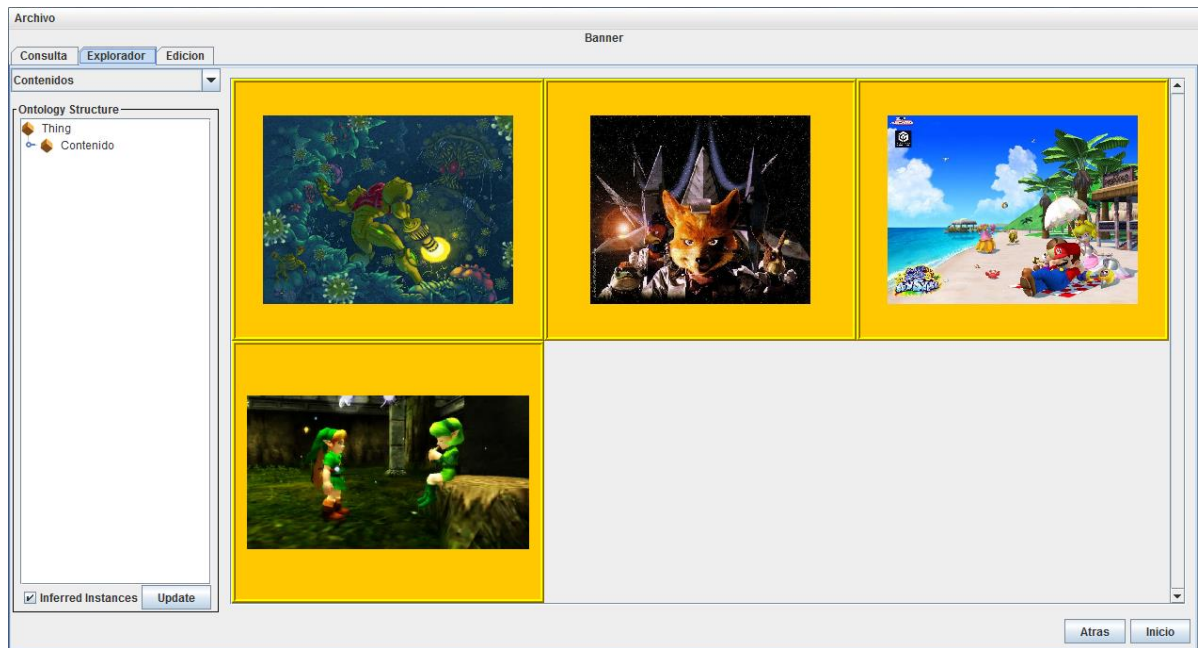
La interfaz de usuario tiene 3 paneles principales:

1. Panel de Consulta:



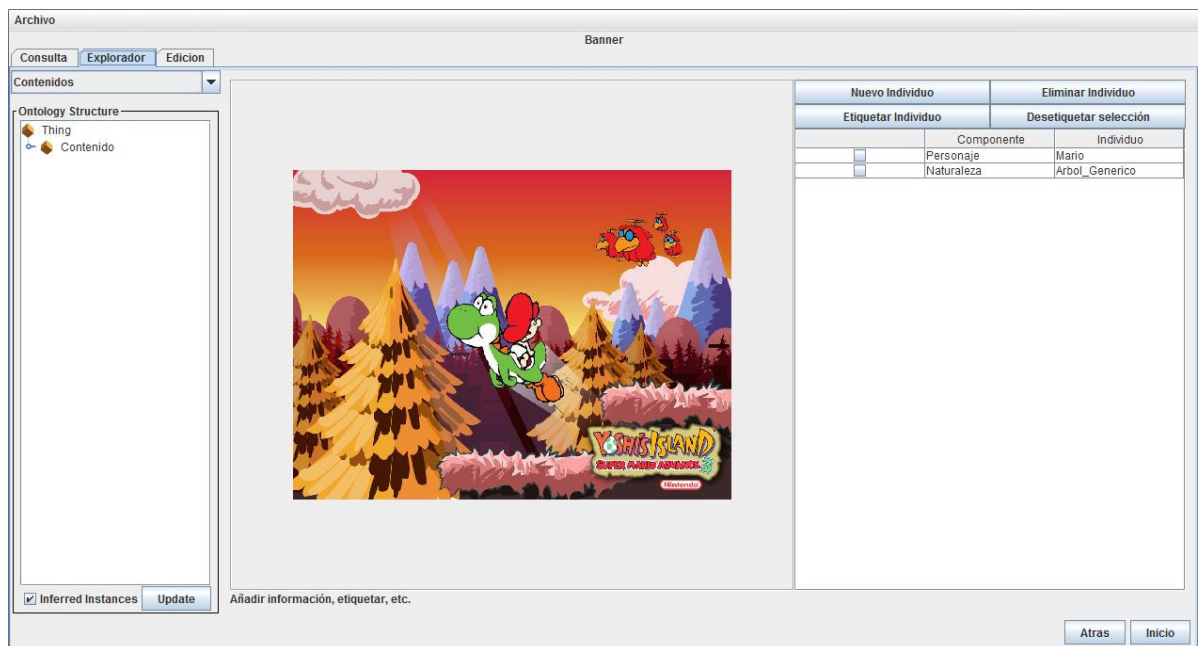
En este panel tenemos a la izquierda los dos modos de búsqueda que proponemos al usuario, la búsqueda por **lenguaje natural** (arriba) y la búsqueda **mediante paneles** (debajo). A la derecha se muestra información sobre la expresión regular que reconoce la aplicación y sobre expresiones que admitimos para suavizar el lenguaje que el usuario puede usar.

2. Panel de Explorador:



Este panel permite al usuario una búsqueda sin ningún tipo de inteligencia. Esta búsqueda se limita a permitirle navegar por las carpetas que tenga la aplicación (mostrándose estas en amarillo) y acceder a las fotos (mostradas en azul).

Una vez se clicla en una foto, se muestra la información relacionada con esta, tal y como se ve debajo:



Arriba a la derecha permitimos también añadir o quitar individuos de la ontología, etiquetar individuos (seleccionándolos en el árbol primero) o desetiquetar individuos (seleccionándolos en el panel de la derecha).


El proceso de etiquetado sería el siguiente:

- Buscamos en el árbol el individuo que deseemos.
- Con el individuo seleccionado le damos a etiquetar individuo.
- Después de unos breves segundos el individuo se etiqueta en la foto.

La aplicación también permite cambiar la raíz del árbol, pudiendo cambiar al árbol de fotografías para poder ver de manera visual la jerarquía.

El árbol de fotografías no permite interacción para etiquetar.

3. El último panel es el de crear nuevos individuos:



The screenshot shows a web application interface with a title bar 'Archivo' and a sub-header 'Banner'. Below the header are three tabs: 'Consulta', 'Explorador', and 'Edición'. The 'Edición' tab is active. The main content area contains four light blue rectangular input fields arranged in a 2x2 grid. The fields are labeled 'Objeto', 'Naturaleza', 'Personaje', and 'Estructura'.

Este panel permite crear individuos en la ontología rellenando sus propiedades como podemos ver debajo.

El único panel que permite la escritura es el nombre del individuo, que quita los espacios en blanco y comprueba que el nombre no exista en la ontología. El resto de campos se rellenan en base a las propiedades que pueda tener el individuo y además se tienen en cuenta los individuos que ya están en la ontología para las relaciones entre ellos.

No hace falta usar Protegé para etiquetar nada de la ontología salvo la clase **Atributos**, que al ser algo privado para los administradores, hemos preferido mantenerlo fuera del alcance de los usuarios.

Herramienta de recuperación

La recuperación semántica la ofrecemos de dos formas:

1. **Mediante paneles:** Damos al usuario tres paneles para que pueda elegir propiedades, individuos o clases y un juego.
2. **Mediante lenguaje natural:** Esta parte permite al usuario introducir consultas respetando los nombres de los individuos y propiedades de la ontología o, usar ciertos patrones para poder hacer más cómoda la consulta.

Ambas maneras responden siempre a la siguiente expresión regular:

```
((propiedad(conector)...(conector)propiedad)*  
(((clase|instancia)(conector)...(conector)(clase|instancia)))|(clase|instancia)  
)?(en Juego)?
```

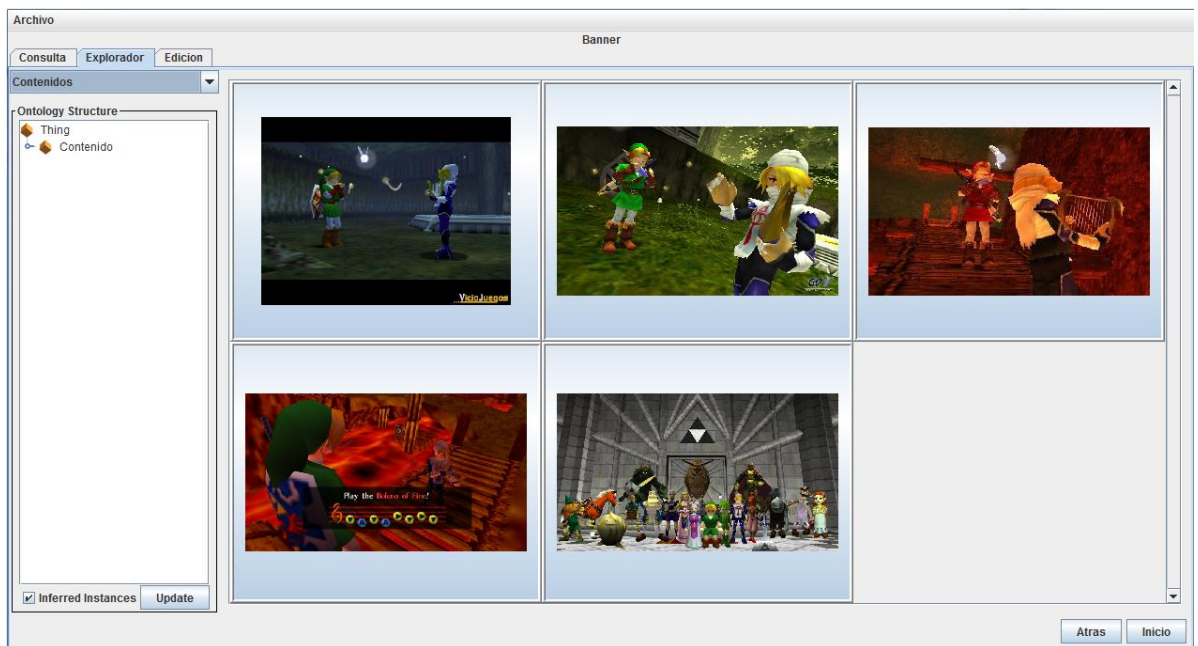
Los conectores, son uniones o intersecciones que asocian siempre a izquierdas. Por ejemplo, si se pusiera “fotos de Link y Mario o Peach”, se devolverían las fotos en las que aparecen Link y Mario y además puede aparecer Peach.

Nos hemos preocupado bastante de dar al usuario una herramienta de consulta lo más potente posible para que tenga la mejor sensación al ejecutar búsquedas. *Creemos que es importante que el usuario pueda percibir todo el potencial que subyace en la ontología y la única manera de conseguirlo, es haciendo que pueda explotarla de la manera más sencilla y cómoda posible.*

El proceso que realiza la herramienta internamente tiene cierta complejidad y se divide en varios pasos.

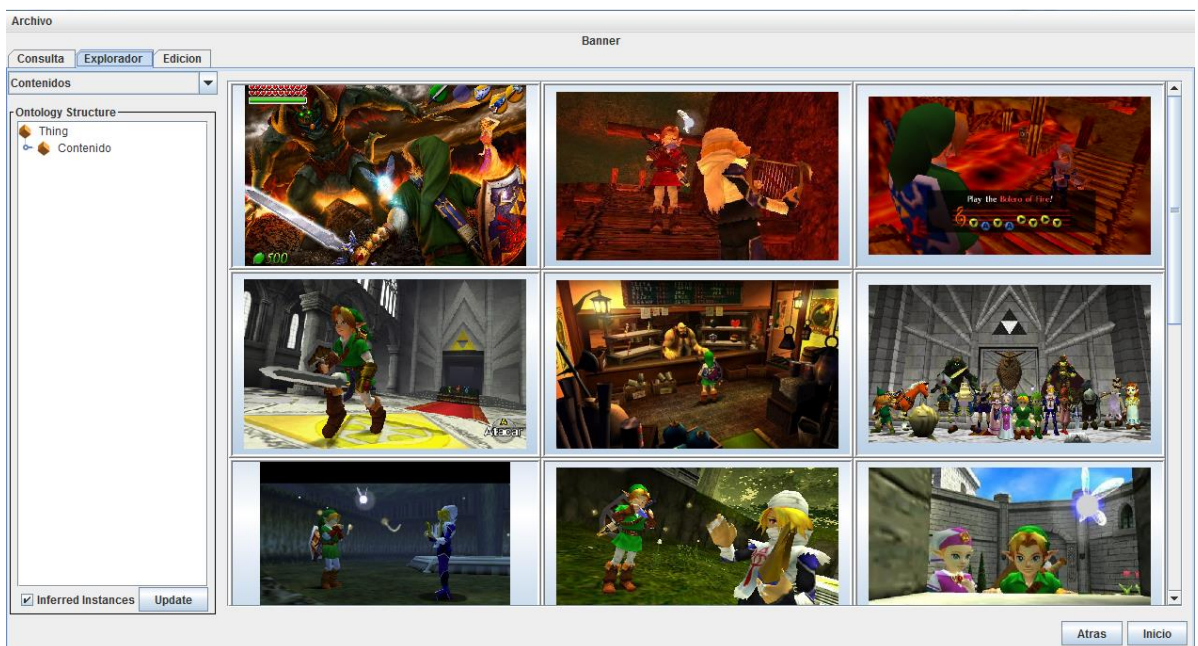
1. Se trata la cadena que el usuario nos pasa para eliminar posibles plurales, mayúsculas y reconocer ciertos patrones de lenguaje.
2. Una vez tenemos la cadena en el lenguaje de nuestra ontología pasamos a ir reconociendo los distintos tokens y a asociarles su categoría, usándose para esto tres fases: fase de propiedades, fase de instancias o individuos y la última fase que comprueba el nombre del juego (en caso de haberlo).
3. El último paso es tratar toda la información sacando los individuos que cumplen las propiedades en base a sus conectores, tratar los conectores de los propios individuos para generar el conjunto buscado y por último sacar las fotos en las que aparezcan dichos individuos.

Una recuperación semántica podría ser escribir “amigos de Link” (la parte de “fotos de” ya va implícita), obteniéndose un resultado similar el siguiente:



Podemos ver que la propia ontología nos ha filtrado la información en base a lo que el usuario ha ido añadiendo al etiquetar los individuos en las fotos.

En cambio una búsqueda sintáctica sería entrar en las fotos del juego y mirar las que hay, sin usar la ontología para nada.



En el ámbito de esta práctica podemos usar la mayor parte de las propiedades, relacionándolas con individuos y entre ellas mismas con conectores lógicos. Una ampliación posible de cara a un proyecto más a largo plazo y con mucho más tiempo, sería el de hacer un reconocedor más complejo y elástico en cuanto al lenguaje que acepte. Esta parte ha sido realmente interesante, sobre todo por su relación con la asignatura PLG en cuanto al tratamiento de lenguajes y su procesamiento.

Panel de Archivo

Este panel incluye la creación de una **nueva ontología**, **salvar la ontología** y **actualizar ontología**, este último sirve para hacer una búsqueda en el fichero de fotos y etiquetar las fotos que no estén ya en la ontología.

Búsqueda semántica y búsqueda sintáctica

Las ventajas de la búsqueda semántica radican en que permite que el usuario vea reconocida en cierta medida el ligero esfuerzo invertido en la etiquetación. Este tipo de búsquedas se valen de propiedades sencillas, que agilizan mucho la búsqueda de información en base a una estructura.

Por el contrario, la búsqueda sintáctica es la obtención de la información sin tratamiento alguno, es mucho menos eficiente, pero la responsabilidad recae totalmente en el usuario, que no puede culpar a la máquina de una mala devolución de la información. Este tipo de búsquedas son más útiles si el usuario no tiene información del ámbito de búsqueda, ya que una mala etiquetación tiene efectos negativos sobre la recopilación de información.

Trabajo de consulta y documentación. Relación con otras herramientas.

AKTiveMedia: <http://www.dcs.shef.ac.uk/~ajay/html/cresearch.html>

No hemos podido acceder a la aplicación desde esa dirección, sin embargo hemos descargado la aplicación desde otro sitio. Permite etiquetar documentos HTML fácilmente y con distinción de colores, lo que facilita el etiquetado y la posterior supervisión. También permite el etiquetado de fotografías, lo que la hace una aplicación con más funcionalidad que la nuestra. Además permite obtener estadísticas interesantes que se muestran en forma de gráficas, lo cual hace que el entorno sea muy visual.

FlickrR: www.flickr.com

FlickrR permite añadir etiquetas adicionales a personas, además de poder añadir comentarios y sacar una gran cantidad de metadatos de la imagen. Además tiene conexión con las redes sociales, cosa que nuestra aplicación no tiene. Se puede también crear grupos y debates, lo que nos ha parecido bastante curioso. En cuanto a las consultas, permite buscar fotos, grupos o personas, pero no una búsqueda más exhaustiva como la que permitimos nosotros en nuestra aplicación.

Facebook (solo la gestión de fotos) http://www.facebook.com/sitetour/photos_notes.php

La gestión de fotos de Facebook es bastante más pobre que nuestra aplicación. Permite etiquetar individuos, pero no tiene información adicional, y las búsquedas son muy limitadas. Se puede ver a través de la biografía cuando se etiquetaron las fotos.

iPhoto <http://www.apple.com/ilife/iphoto/what-is-iphoto.html>

Tiene conexión con redes sociales, lo que parece ser un factor común en muchas de estas aplicaciones. Sin embargo parece un programa mucho más simple, para usuarios poco especializados y no parece tener muchas opciones semánticas, más bien es una herramienta de compartir fotos sin especial significado en información semántica.

ESP Game. Labeling the web. <http://www.espgame.org/>

Una aplicación de etiquetado de fotos, en donde la ontología queda oculta a los usuarios, que son jugadores. Es una forma de etiquetado encubierta. Al ser un juego no tiene muchas opciones de incluir nuestras fotos o de modificar la ontología. Aunque el juego es bastante divertido, en nuestra prueba nos tocó un señor bastante borde que se enfadó con nosotros por no saber jugar (que sabrá el de ontologías).

SMORE <http://owl.mindswap.org/downloads/>

Se pueden manejar muy cómodamente las ontologías, así como crear nuevos individuos, ver en una lista todas las clases, propiedades e individuos, etc. Permite importar varias ontologías. Lo que difiere de nuestra aplicación es que mientras SMORE sirve para marcar HTML, nuestra aplicación es para marcar fotos, con lo cual las aplicaciones de ambas son en distintos campos, aunque hemos de decir que la forma de poder modificar la ontología es mucho más abierta que en nuestra aplicación.

PhotoStuff - <http://www.mindswap.org/2003/PhotoStuff/>

Este programa si se parece más al nuestro y tiene algunas opciones interesantes, como la de favoritos, que permite guardar direcciones de fotos que nos hayan gustado o de ontologías para volverlas a cargar cuando queramos. La forma de navegar es un poquito más simple que la nuestra, pero también con un árbol a la vista y permite crear individuos mientras visualizamos la foto para etiquetar. Tiene una interfaz menos recargada que la versión desarrollada por nosotros y con un poco de funcionalidad adicional, se trata de una herramienta muy parecida.

FotoTagger <http://www.fototagger.com/>

Otra aplicación similar a la nuestra, en donde se da la importancia a la foto y con un toque original: los etiquetados quedan representados por bocadillos. Sin embargo es un etiquetado más visual, con lo que la información de ontologías queda más descuidado, así como la parte de recuperación de información.

Caliph & Emir <http://www.semanticmetadata.net/features/>

Separado en dos programas, Caliph & Emir, para poder etiquetar y consultar. El etiquetado es muy simple, pero saca una gran cantidad de meta información de la fotografía y permite introducir mucha información adicional (tiene un campo de texto para otros comentarios, indicar la calidad de la foto, quién puso los metadatos, quién tomó la foto, etc). La parte de la ontología se muestra muy simple y permite añadir relaciones y propiedades de forma muy intuitiva y visual gracias a grafos.

Grupo del trabajo

Raúl:

- Búsqueda de fotos.
- Ayuda en la realización de la ontología.
- Ayuda en cosas puntuales de la interfaz.
- Realización del controlador.

Xavier:

- Realización de la Interfaz.

Alvaro:

- Ontología.
- Recuperador Semántico.
- Ayuda en el controlador.
- Etiquetado de los individuos.

Conclusiones

Con el diseño del sistema hemos aprendido a manejar ontologías desde Protegé y a poder usarlas luego de manera externa para búsquedas semánticas. También se nos han planteado las ventajas y desventajas de usar este tipo de búsquedas, frente a las sintácticas, que aun siendo más simples pueden ser más útiles si el usuario no hace buen uso de las herramientas de las que dispone.

La importancia de crear clases que sean de verdad necesarias y útiles y que permitan una mejor clasificación de la información nos ha parecido un aspecto fundamental del proceso, ya que alguna vez hemos creado clases superfluas o clases que eran demasiado específicas (casi como si actuaran a modo de individuo).

Todo este proceso por desgracia no sirve de nada si el usuario no encuentra atractivo su uso, por lo que hemos intentado hacer las búsquedas semánticas lo más generales posibles, permitiendo todo tipo de variaciones en los parámetros de búsqueda. También hemos creado una interfaz cómoda de usar y que no agobie con una cantidad de información demasiado extensa por cada individuo.

Nuestra práctica destaca en que el uso de Protegé únicamente ha sido necesario para construir un esqueleto sobre el que trabajar, ya que toda la interacción se realiza sin necesidad de usar esta herramienta. Nuestra interfaz es realmente completa, permitiendo al usuario expresar al máximo todas las propiedades de la ontología, pero restringiendo lo máximo posible para que no haya ambigüedades. La ontología nos parece bastante sólida y muy acertada para el uso que le hemos dado.

El proceso de anotación podría automatizarse si consiguiésemos reconocer algún tipo de patrón en las imágenes, lo que daría mucha agilidad al proceso.

Una anotación mucho más simple, que si realizamos nosotros, es etiquetar el nombre del juego según la carpeta donde esté (proceso que realizamos de forma automática).

El programa se ha probado en Windows y no ha dado fallos, por desgracia no podemos asegurar su funcionamiento fuera de este sistema operativo.

Las dificultades de esta práctica se nos han presentado en cuanto a estética, ya que todo el trabajo que hemos hecho por debajo teníamos que presentarlo de forma acorde para que el usuario percibiera que la aplicación tiene la funcionalidad que le estamos dando. También nos hemos peleado mucho con la recuperación, que nos ha traído de cabeza para intentar generalizarla.

Explica si crees que se han cumplido los objetivos docentes planteados (puedes valorarlo del 1 al 5):

- Aprender a editar ontologías con la herramienta Protege utilizando el lenguaje OWL-DL: **5**
- Utilizar las ontologías como mecanismo de anotación semántica y recuperación de información: **5**
- Cargar y manejar ontologías desde Java usando el puente OntoBridge¹ (JENA)/ DIG. Uso de razonadores de DLs: **5**
- Entender las dificultades y la utilidad de las ontologías como mecanismo de anotación semántica y recuperación de información (imágenes): **5**
- Entender las diferencias entre **recuperación sintáctica/semántica**: **5**

Extensiones que se podrían realizar de forma sencilla. Explicar cómo lo harías.
¿Qué extensiones te parecerían adecuadas/útiles/interesantes para incluir en tu herramienta?

¹ Desarrollado por el grupo GAIA (gaia.fdi.ucm.es) y accesible a través de la página WEB <http://gaia.fdi.ucm.es/grupo/projects/ontobridge/>

Opinión de la práctica

- Nivel de dificultad: 4.
- Interés: 5.
- Motivación: 4.