Octava práctica de laboratorio de Sistemas de Información para la Web

Motivación

“Piano piano, si arriva lontano”

La mejor forma de valorar las bondades de las tecnologías semánticas es viendo su aplicación práctica en casos reales. Para ello iremos viendo pequeños ejemplos así como herramientas que nos servirán para introducir conceptos como los microdatos en HTML5 o JSON-LD.

En consecuencia esta sesión práctica estará muy guiada y tan solo al final se plantearán algunos ejercicios sencillos para su entrega.

Se recomienda seguir los pasos en orden y al ritmo que marque el profesor:

Primeros ejemplos

Visita el siguiente documento. No trates de traducirlo ni de decodificarlo, esta es la “visión” que tiene una máquina de un **texto plano**; es posible que haya un sentido subyacente pero no puede extraerse de manera inmediata. Con acceso a más texto plano podrían extraerse cadenas significativas pero seguiría sin haber una semántica accesible para la máquina.

Visita ahora el siguiente hipertexto. No visualices el código HTML, céntrate tan sólo en los aspectos más obvios: los enlaces y las cadenas con énfasis. Esa sería la “visión” de una máquina de un **hipertexto básico**. Sigue sin disponer del significado de ese texto pero sí “sabe” que las siguientes cadenas son importantes y van, además, asociadas a una URL:

- *bsfb pg jñufsftu (http://danigayo.info/research)*
- *tpdjbm nfejb sftfbsdi (http://danigayo.info/research/#socialmediaresearch)*
- *J ibwf qvcmjtife (http://danigayo.info/publications)*
- *JFFF Jñufsñfu Dpnqvujñh (http://danigayo.info/publications/detail.php?id=...)*
- *JFFF Nvmujnfejb (http://danigayo.info/publications/detail.php?id=...)*
- *tqfdjbm jttvf pg Jñufsñfu Sftfbsdi pñ uif qsfejdujwf qpxfs pg tpdjbm nfejb (http://www.emeraldinsight.com/toc/intr/23/5)*
- *dibqufs pñ Qpmjujdbm Pqjñjpñ (http://danigayo.info/publications/detail.php?id=...)*

Con independencia de lo que signifique ese texto o las URLs un buscador “sabría” que debería asociarlo como metadatos a las URLs indicadas ya que tal vez pueda ser útil para resolver

consultas; después de todo, ya hay al menos un sitio donde se hace referencia a esas URLs con esos términos.

Visita ahora el siguiente documento; se trata en apariencia del mismo hipertexto anterior pero hay diferencias. Antes de profundizar en el código fuente visita la siguiente herramienta: *Google Structured Data Testing Tool*.

Carga en la misma la siguiente URL:

```
http://danigayo.info/teaching/SIW/ejemplos/02-microdata.html
```

La herramienta te mostrará lo que una máquina puede "saber" gracias a los microdatos incrustados en el HTML:

- Ha descubierto que en ese texto se mencionan 4 publicaciones periódicas:
  - *Dpnnvñjdbujpñt pg uif BDN*
  - *JFFF Jñufsñfu Dpnqvujñh*
  - *JFFF Nvmujnfejb*
  - *Jñufsñfu Sftfbsdi*
- Ha descubierto un capítulo de libro:
  - titulado: *"Qpmjujdbm Pqjñjpñ"*
  - en el libro: *"Uxjuufs: B Ejhjubm Tpdjptdpqf"*
  - con ISBN: 9781107500075
  - publicado por la editorial: *Dbncsjehf Vñjwfstjuz Qsftt*
- Ha descubierto además una persona:
  - *Ebñjfm Hbzp-Bwfmmp*
  - que trabaja de *bttpdjbuf qspgfttps*
  - en un college o universidad llamada *Vñjwfstjuz pg Pwjfep*
  - dentro del departamento *Efqbsunfñu pg Dpnqvufs Tdjfñdf*

Si observas el código fuente podrás ver cómo toda esa información se incrustó gracias a unas propiedades HTML que tal vez no conozcas:

- `itemscope`
- `itemid`
- `itemtype`
- `itemprop`

Toda la información necesaria para trabajar con esta extensión de HTML la tienes en el documento *HTML Microdata*. Sin embargo, para comenzar te basta saber lo siguiente:

- La primera propiedad, **`itemscope`**, indica que todo el código HTML contenido dentro del elemento (habitualmente un *div* o un *span*) que la usa hace referencia a un único ítem, esto es, un objeto físico o lógico (p.ej. Una persona, una película, una organización, una canción, …)

- La propiedad **`itemid`** es opcional pero, de aparecer, siempre va asociada a un itemscope. Dicha propiedad recibe una URL válida para identificar al ítem en cuestión. En el ejemplo que estamos usando aparecen una URL de una página personal y dos ISBNs usando URNs[1].
- La propiedad **`itemprop`** va asociada a un elemento HTML (habitualmente un *div* o un *span*) anidado dentro de un elemento con `itemscope` e indica que el código HTML que encierra se refiere a una propiedad del `itemscope` que lo contiene. Obsérvese en el ejemplo que se puede usar en combinación con itemscope para definir un ítem dentro de otro ítem.
- Por último, **`itemtype`** permite indicar (con una URL absoluta válida) el tipo (o clase si se quiere) al que pertenece un `itemscope` dado.

Surge entonces una cuestión no trivial: ¿de dónde salen los tipos para `itemtype` y las propiedades para `itemprop`?

El ejemplo vuelve a darnos pistas importantes. Como se puede apreciar todos los tipos usados en el ejemplo (*Book, Chapter, Periodical, Person, CollegeOrUniversity*, y *Organization*) han sido definidos en `http://schema.org`[2].

Si visitamos la URL de alguno de esos tipos, p.ej. `http://schema.org/Person` veremos que aparecen las propiedades de dicho tipo incluyendo *affiliation* y *jobTitle*. A su vez, al ser *Person* un subtipo/subclase de *Thing* hereda sus propiedades, como p.ej. *Name*.

Primer ejercicio entregable

¡Atención! No comiences aún este ejercicio; pasa al siguiente conjunto de ejemplos.

Elige una noticia de un periódico online nacional o extranjero; el idioma puede ser inglés o castellano. Seleccionar 2 o 3 párrafos (alrededor de 300 palabras).

Utiliza la demo de *Open Calais* o la de *Dandelion* para obtener una primera lista de entidades candidatas junto con sus posibles tipos.

Completa dicha lista con tu propio conocimiento experto. El objetivo sería localizar el mayor número posible de ítems, junto con propiedades y valores para los mismos.

Visita la lista de esquemas disponibles en *Schema.org*, selecciona los tipos más apropiados para tus ítems junto con sus propiedades. Asigna valores para las propiedades que estén presentes en el texto que estás etiquetando.

---

[1] Puedes encontrar aquí la lista de espacios de nombres en URNs.

[2] Proyecto colaborativo con apoyo de empresas de búsqueda en la Web para crear y mantener vocabularios semánticos para la estructuración de información.

Plantéate las siguientes cuestiones: ¿deberían tener valor para `itemid` todos los ítems? ¿Qué valor debería asignarse? ¿Si hubiera varios candidatos cuál escogerías? ¿Por qué? ¿Cuál crees que es la solución de compromiso que podría resultar menos polémica en la mayoría de casos? ¿Qué inconvenientes concretos te ha supuesto la obligación de incrustar los metadatos allí donde te "forzaba" la estructura del texto?

Se entregará en el campus virtual un archivo comprimido que contendrá:

- El enlace permanente a la noticia que se haya usado.
- El texto plano seleccionado para su etiquetado.
- El texto plano con el correspondiente etiquetado HTML usando microdatos.
- Un documento que responda a las preguntas planteadas antes así como capturas de pantalla de los resultados de evaluación del HTML etiquetado con *Google Structured Data Testing Tool*.

Más ejemplos

Visita el siguiente documento, se trata de la versión "descifrada" de los ejemplos que has estado usando hasta ahora. Para una máquina el texto (más allá de los metadatos) sigue siendo igual de indescifrable pero a nosotros nos resulta más manejable.

Al igual que el último ejemplo en este caso se han usado microdatos HTML5 para etiquetar; eso supone una ventaja (no se usa ninguna tecnología externa) pero también un inconveniente. Por ejemplo, por la forma en que se ha redactado el texto ha sido posible dar el título del capítulo y del libro pero no es viable indicar el título de los artículos publicados en revistas.

Obviamente podría añadirse el título de las publicaciones pero si nos ceñimos a microdatos tendría que hacerse en el propio texto y podría resultar muy verboso. ¿Habría alguna opción para añadir metadatos visibles para la máquina pero invisibles para los lectores?

Visita la siguiente herramienta: *RDF Translator* e introduce esta URL:

```
http://danigayo.info/teaching/SIW/ejemplos/microdata.html
```

En el desplegable para el *input* déjalo en automático y en el *output* indica JSON-LD. Pulsa *Submit.*

Debería aparecer algo parecido a este fichero.

Antes de profundizar en el mismo y ver cómo manipularlo para incluir información adicional veamos si valida.

Visita *Google Structured Data Testing Tool* e introduce la siguiente URL:

```
http://danigayo.info/teaching/SIW/ejemplos/03-json-ld.json
```

Si tratas de **validarla** se producirán lamentablemente varios errores (en principio 5) pero los 3 más llamativos son los siguientes:

- *https://schema.org/CollegeOrUniversity is not a known valid target type for the affiliation property.*
- *https://schema.org/CollegeOrUniversity (The type https://schema.org/CollegeOrUniversity is not a type known to Google.)*
- *https://schema.org/Organization (The type https://schema.org/Organization is not a type known to Google.)*

Aparentemente la herramienta de Google no es capaz de reconocer en JSON-LD dos tipos que sí reconoce sin problemas con microdatos. El problema parece deberse a que el "traductor" de microdatos a JSON-LD introdujo HTTPS en las URLs de *Schema.org* y el validador de Google sólo las reconoce con el protocolo HTTP.

Si ahora probases con esta URL…

```
http://danigayo.info/teaching/SIW/ejemplos/03-json-ld-v2.json
```

Parte de los problemas **deberían resolverse** pero sigue habiendo uno con el tipo del siguiente identificador:

```
http://danigayo.info/teaching/SIW/ejemplos/microdata.html
```

Se trata del documento original con microdatos que fue traducido así que, realmente, no lo necesitamos. Así pues, si se elimina la información relativa a esa URL del archivo tendríamos el siguiente archivo que **sí valida** y proporciona obviamente la misma información que los microdatos originales.

Segundo ejercicio entregable

¡Atención! No comiences aún con este ejercicio; pasa al siguiente conjunto de ejemplos.

Modificar la tercera versión del archivo JSON teniendo en cuenta lo siguiente:

- Los identificadores de las publicaciones periódicas pueden ser URLs válidas no ficticias. Recuérdese que se trata de revistas que tendrán una página web principal.
- El capítulo de libro puede tener como identificador un DOI. Busca en la página oficial del libro…
- Cambiar el tipo asignado a *Cambridge University Press*, aparece como *Thing* pero eso es manifiestamente mejorable.
- Cambiar el identificador asociado a la persona *Daniel Gayo-Avello*, recordar que en la Web se les suele asociar su página web personal.
- Cambiar los identificadores para *University of Oviedo* y *Department of Computer Science*. Aquí hay varias posibilidades válidas.

- **Muy importante:** añadir los artículos científicos publicados en las publicaciones periódicas, recuerda que era información importante pero que no era fácil de añadir con microdatos. Se recomienda usar el tipo *ScholarlyArticle*. Se valorará especialmente el uso adecuado de propiedades para incluir metadatos relevantes para cada artículo (disponibles en la web del profesor).

Hecho eso editar un archivo HTML que contenga el contenido del archivo microdata **sin los microdatos** e incrustar el correspondiente contenido JSON-LD (véase *"Introduction to Structured Data"*). El archivo HTML final con metadatos incrustados tiene que validar en la herramienta de Google usada hasta ahora.

Aún más ejemplos

Visita los siguientes documentos y trata de validar los metadatos incrustados usando la herramienta de Google. En caso de que no validasen trata de "repararlos" (este ejercicio no forma parte del entregable de la sesión).

- Los expertos de la ONU urgen a tomar medidas drásticas contra el cambio climático
- Two ships collide in the Mediterranean, causing fuel spill
- Climate report: Scientists urge deep rapid change to limit warming

Referencias bibliográficas

- W3C (2018), HTML Microdata, W3C Working Draft 26 April 2018
- JSON for Linking Data