NUEVAS FORMAS DE PRODUCCIÓN TEXTUAL: EL DESARROLLO DE LA WEB

Enrique ALONSO
Dept. de Lingüística, Lenguas Modernas, Lógica y Filosofía de la Ciencia Universidad Autónoma de Madrid

ABSTRACT: The current techniques to produce texts and documents have inherited largely the devised models to work with old typewriters. Nevertheless, it has been progressively appraised the possibility of considering new techniques to produce texts impelled by the development of a Wold Wide Web based on documents written in HTML. Tagged texts and Mark-up languages have shown the existence of new strategies to interact with artificial agents far beyond the traditional goals posed by Artificial Intelligence. At the present time the main task consists in designing efficient text processors oriented to compose tagged. The new strategies can produce a true revolution in the textual production techniques.

1. La nueva apertura textual: textos y datos

La consolidación del formato digital como nuevo marco en el que entender la producción de textos ha hecho que lo que hasta hace bien poco sólo era asunto de ingenieros y tecnólogos se haya convertido en un problema que nos afecta a todos.

En la creación de un documento interviene, no solo el ingenio de quien lo produce, sino también una serie de condiciones técnicas que en cierto modo determinan, no sólo lo que se puede contar, sino lo que otros pueden hacer con esos textos. El estudio de la escritura y sus distintas formas es un buen ejemplo de lo que quiero decir. No se pueden contar las mismas historias cuando mi escri-

tura está sometida a grandes variaciones interpretativas, como en los esquemas jeroglíficos o silábicos, que cuando ese problema se minimiza gracias a los alfabetos fonéticos. Y tampoco serán idénticas las reacciones, sentimientos e ideas que pueden provocar en aquellos a quienes están dirigidos esos productos. Ni siquiera cabe hablar de que el público sea el mismo en todos los casos. Sabemos que, de hecho, no lo es.

Para entender el tipo de técnicas vigentes en la fabricación de textos tendremos que mirar necesariamente al mundo de la tecnología y más en concreto a algunas de las tendencias que empiezan a establecerse durante la década de 1980 en el diseño de software. Parece obvia la existencia de dos tipos de aplicaciones que predominan sobre cualesquiera otras durante ese periodo: el procesamiento de textos y el procesamiento de datos. La rápida incorporación de estas herramientas a la vida cotidiana favoreció claramente la progresiva implantación del código propietario sobre otras posibles alternativas. Desarrollar uno mismo un procesador de textos elemental podía suponer un nivel de conocimientos considerable y lo que es peor, comprometía la coherencia y la circulación de textos y archivos. Por otra parte, pedir a las compañías que compartiesen su código era algo que simplemente resultaba y resulta contrario a cualquier lógica empresarial. ¿Quién iba a querer además complicarse aún más la vida intentado cambiar aspectos de sus herramientas ya de por sí complejas?

El éxito de programas como WordStar o las distintas versiones de WordPerfect permitieron resolver el problema de la compatibilidad de los archivos sin abandonar el esquema general de la libre competencia empresarial, algo que a priori podía parecer la cuadratura del círculo. Disponer de máquinas preparadas para trabajar sin grandes cambios como procesadores de texto, con varias versiones, y como procesadores de datos —pienso en la generación de Dbase— hizo que su uso se fijara entorno a estas aplicaciones haciendo de la velocidad y la sencillez para acceder a estos recursos algo primordial. El esquema de línea de comandos —DOS— estaba condenado abriéndose camino la idea de un escritorio icónico desde el que arrancar todas nuestras aplicaciones de forma rápida y sencilla.

Pero lo que me interesa desatacar de esta especie de mirada nostálgica a la prehistoria de la informática es la profunda diferencia que había, y persiste, en el modo de tratar textos y datos. Los programas destinados al procesamiento de datos estuvieron orientados desde un principio a recuperar la información ver-

tida de forma rápida y altamente estructurada. Es decir, existía una considerable diferencia entre el lugar en el que se almacenaban los datos y las presentaciones que de ellos se podía hacer. Lo cual resulta lógico ya que esa es la esencia misma de un proceso de datos. Esta diferencia permitió mantener un esquema relativamente simple para la matriz de datos[1] haciendo de ella un elemento extraordinariamente estable, incluso ante posibles cambios de plataforma.

Los procesadores de texto siguieron, sin embargo, una dirección completamente distinta. Los primeros procesadores, WordStar y las distintas generaciones de WordPerfect —incluido el popular wp5.1— ofrecían en pantalla unos textos en los que era posible reconocer la estructura tipográfica aplicada aunque no se visualizara el resultado final. En la medida en que la estructura aplicada al texto era en su práctica totalidad estructura referida a la presentación final del documento esto podía ser considerado un atraso y no una ventaja. Las etiquetas que acompañaban al texto para aplicar formatos eran en realidad un modo de evitar cargas adicionales para el procesador. WordPerfect 5.1 supuso una notable mejora al incorporar un visor de formato final —preimpresión— en el que, no obstante, no se podía trabajar de forma eficaz.

La revolución impulsada por Windows con la incorporación de Word consiste, precisamente, en ofrecer presentaciones en pantalla que coinciden de forma sustancial con el formato final impreso. La evolución de los procesadores de texto se ha orientado hacia la obtención de la máxima coherencia entre lo que nuestra máquina presenta y lo que nuestros ojos pueden ver. Los textos que elaboramos por medio de los actuales procesadores, en su mayoría Word[2], están pensados para ser leídos por otras personas en soportes tradicionales, esto es, en papel.

Esta es la idea a la que se han ajustado los procesadores de texto desde su extensión masiva a finales de la década de 1980. Sin embargo, las cosas han ido cambiando sustancialmente en este plazo apareciendo posibilidades que com-

---

[1] Por *matriz de datos* entiendo simplemente el lugar en el que se almacenan los registros de una base da datos.

[2] Es cierto que LaTex sigue una dirección completamente distinta pero eso no nos debe hacer olvidar la existencia de procesadores basados en LaTex que intentan forzar su filosofía. Tal es el caso de SW.

prometen la filosofía inicial de estas herramientas. La filosofía del *formato papel* está siendo desafiada a través de las nuevas dinámicas impuestas por la revolución de las telecomunicaciones. La consecuencia quizá más relevante para lo que aquí importa es la existencia de un número creciente de documentos que nunca son finalmente trasladados a papel. Nuestros textos viajan cada vez más de un soporte electrónico a otro sin pasar por papel alguno. La extensión en el uso de procesadores de texto ha generado una biblioteca de documentos .doc —.txt, .rtf, o lo que es peor .pdf— muy considerable. Se ha logrado, además, una cierta coherencia entre los soportes en que se encuentran almacenados con lo que su recuperación no ofrece un excesivo problema —solo el diskett de $3\ ^{1}/_{4}$ se halla en vías de extinción aunque su desaparición está resultando ser lo suficientemente lenta como para garantizar la recuperación de la información relevante. Disponemos pues de inmensas bibliotecas digitales cuyos documentos han sido concebidos, desde el punto de vista de la información que almacenan, como textos preparados para ser impresos y leídos en cualquier momento. Su información no resulta más fácil de tratar y lo que es más importante, de recuperar, que la que puede encontrarse en una hemeroteca o archivo tradicional.

Siempre podemos aducir que disponer de textos digitales nos facilita el acceso a buscadores automáticos como los que habitualmente encontramos en los procesadores de texto, pero lo cierto es que la ventaja es poca. Y ahora la cuestión es ¿cómo debemos enfrentarnos a la tarea de recuperar información .doc que sabemos que se encuentra en formato electrónico?

Todo parece indicar que la única opción es conformarse con la situación heredada de la propia evolución del mercado del software y confiar los datos relevantes a otro tipo de medios. Si hay datos que por su carácter van a ser requeridos en un futuro lo mejor será tratarlos como datos y no como texto. Los documentos escritos parecen quedar irremediablemente fuera de los procesos habituales de recuperación y procesamiento de la información. Esta forma de ver las cosas, fuertemente apoyada en la filosofía del *formato papel*, es incoherente con el inmenso volumen de documentación de tipo corporativo que constantemente es generada por las instituciones. Decisiones de tipo administrativo que con frecuencia tienen consecuencias legales son condenadas de este modo a la penosa tarea de extractar y resumir la información de una y mil maneras hasta presentarla de forma realmente útil para la comunidad a la que sirve.

La situación tiene todo el aspecto de un callejón sin salida desde el cual poco parece que se pueda hacer. Retomaremos el punto más adelante.

3. La Web y el código html

La discusión que he presentado no es del todo original, aunque tampoco pretende serlo. En realidad es una aplicación de las sugerencias hechas por Tim Berners-Lee, James Hendler and Ora Lassila en un artículo publicado en Scientific American en 2001 que lleva por título «The Semantic Web. A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities».

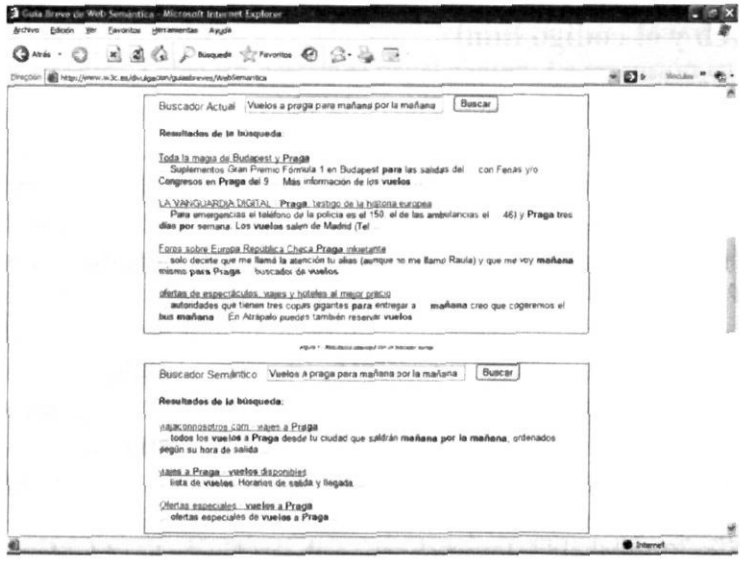
En este artículo se plantea de forma clara la necesidad de reorientar la filosofía que aquí he llamado del formato papel en una dirección completamente distinta. La siguiente cita resume perfectamente el cambio de actitud que se requiere: «Most of the Web's content today is designed for humans to read, not for computer programs to manipulate meaningfully. Computers can adeptly parse Web pages for layout and routine processing—here a header, there a link to another page—but in general, computers have no reliable way to process the semantics...»

Lo cierto es que el diseño de los contenidos de la red parece haberse guiado durante este periodo de crecimiento expansivo por criterios similares a los que en su día influyeron sobre el software de texto. Las páginas Web se orientan hacia la lectura que de ellas puedan hacer los agentes que las consultan dando por supuesto todos aquellos aspectos que, por así decirlo, ya están incorporados en el software correspondiente a la cognición humana.

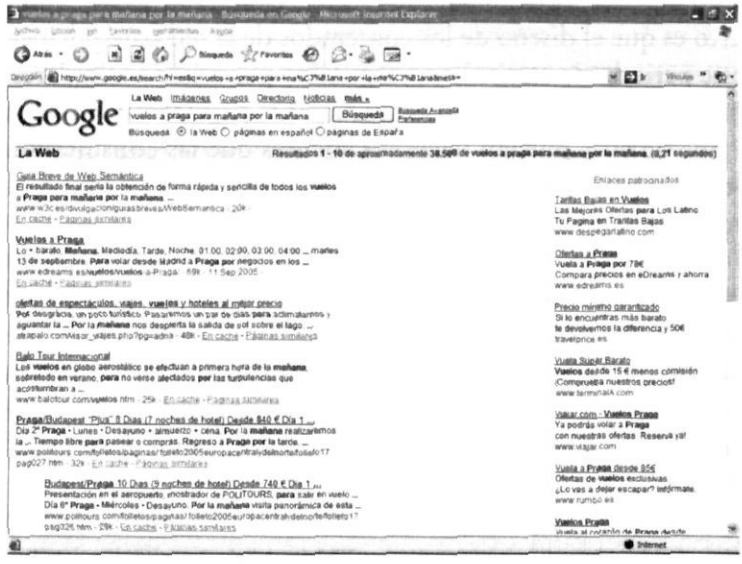
La protesta de estos autores apunta, entre otras cosas, a la escasa eficiencia de las búsquedas que se efectúan en la red mediante programas típicos como Google. Si se acude a la Oficina española del consorcio W3C y se consulta el apartado correspondiente al concepto «web semántica» se puede encontrar una defensa justificada de este punto[3]. Se ofrecen ejemplos de una búsqueda *clásica* orientada

[3] http://www.w3c.es/divulgacion/guiasbreves/WebSemantica

a obtener información sobre los vuelos existentes a una determinada ciudad europea, Praga en este caso. El resultado, tal y como se puede ver en la siguiente figura, es decepcionante.



Lo peligroso de ofrecer ejemplos reales es que alguien los verifique algún tiempo después. El resultado de esa misma búsqueda a día de hoy es el siguiente:



No quiero defender con esto la eficacia de Google minando la línea abierta por Berners-Lee y lo que se ha dado en llamar la Web semántica, ya que lo cierto es que la forma que tenemos de enfrentarnos a una búsqueda resulta en la actualidad extremadamente simple y en algún sentido primitiva. La forma en que abordamos esta tarea pasa por la elección de palabras clave. No hay diálogo, no hay posibilidad de formular preguntas complejas, ni de rastrear alternativas con igual valor. Esto es lo que se intenta paliar a través de la incorporación de una estructura semántica en la Web. Nuestras búsquedas son más o menos eficaces porque elegimos bien las palabras y porque quienes diseñan las páginas se preocupan por incluir suficientes expresiones clave en ubicaciones estratégicas.

Aunque la tendencia seguida en el diseño de páginas web ha amenazado en algún momento con repetir la pauta adoptada en su día por los procesadores de texto hay un hecho que parece limitar la posibilidad de continuar en esa línea. Es cierto que algunos editores de páginas Web han intentado ofrecer en pantalla salidas finales de los proyectos que el usuario crea, pero ninguno se ha atrevido por completo a prescindir de la presentación del código html subyacente.

Estamos en todos los casos ante *editores de pantalla partida* que permiten visualizar simultáneamente código y diseño. La razón por la que se ha detenido la tendencia hacia presentaciones finalistas de los documentos html es la relativa independencia de este lenguaje con respecto al editor elegido. Un texto generado por un editor al uso tenía como formato final una hoja impresa. En el caso de un documento para la Web el formato final es un archivo html. Y siempre va a ser posible retocar un documento html desde cualquier editor por primitivo que este sea.

La moraleja que se obtiene de este somero análisis de la Web es la existencia de tendencias contrapuestas y hasta cierto punto contradictorias. La existencia de un lenguaje universal que como html es independiente del editor con que se trabaja ha fomentado una concepción de los documentos web en los que la estructura y el contenido son presentados con relativa independencia. Se trata de *documentos estructurados* en los que es posible acceder tanto al contenido textual como a toda aquella información que hace referencia a lo que hay que hacer con ese contenido. Para entender este punto basta comparar el aspecto de una página web con su código fuente, siempre accesible desde cualquier navegador. Pensemos por un momento en lo que sucede con un documento de texto estándar. Es obvio que contiene mucha información acerca del formato final del texto, tipo

de letra, notas a pie, resaltados, etc. No obstante nada de esto es visible. No se puede retocar con independencia del propio texto.

Disponer por un lado del contenido de una página y por otro de su formato ha llevado a fomentar la progresiva independencia de ambos elementos como se muestra en la proliferación de las páginas de estilos .css.

Lo paradójico de esta situación es que estando así las cosas se haya permitido que dicha estructura sólo suministre información gráfica incapaz de aportar un valor añadido a la gestión automática de la información. Entiendo que esta es la protesta de la que surge la demanda de una Web semántica. Es posible incorporar una gran cantidad de información no visible a través de la estructura subyacente de un documento web destinada a que determinados agentes automáticos puedan realizar tareas en las que hasta ahora no había sido posible pensar.

En resumen, identificar la distinción entre contenido y estructura en un documento supone en la actualidad una vía de desarrollo cuyo potencial aún está por descubrir, pero que ya ha empezado a ser explotado por algunas iniciativas.

4. Documentos etiquetados: los nuevos lenguajes de la Web

Puede parecer que exageramos al insistir en la pobre utilización que se ha hecho de la estructura que subyace al texto de un documento html. Al fin y al cabo hay muchas cosas que se pueden hacer pinchando en los botones apropiados o rellenando los casilleros que aparecen ante el usuario. Cosas que, ciertamente, nunca figuran explícitamente en el contenido textual de un documento

html. Pero el problema no se refiere tanto a las cosas que sí se pueden hacer, como a las que podrían hacerse sin cambiar de forma drástica algunos de los elementos ya existentes.

El lenguaje html que todos conocemos en mayor o menor medida adopta la forma de lo que se ha venido a denominar *lenguaje de etiquetas o marcas*. La idea es extremadamente simple y hereda, aunque parezca mentira, la estrategia que en su origen emplearon los procesadores de texto más conocidos. Una etiqueta tiene el aspecto general <etiqueta> texto </etiqueta>. Es, muchos podemos recordarlo, el formato empleado antaño por los procesadores al uso para aplicar formato a un texto previamente marcado como bloque. El bloque seleccionado simplemente queda inscrito entre los distintivos de la correspondiente etiqueta quedando afectado por la acción que el intérprete sabe reconocer.

El lenguaje html presenta un acerbo de etiquetas que, pese a cambios más o menos periódicos, se ha mantenido estable en el tiempo. Este considerable logro está ligado a la organización conocida por las siglas W3C —World Wide Web Consortium— que vela por mantener la coherencia de la Web adaptando las distintas versiones de los lectores —Browsers— a las novedades habidas en el código html. La existencia de esta especie de *Real Academia de los lenguajes de Red* no se debe, como es fácil suponer, al denodado esfuerzo de ninguna minoría ilustrada, sino a la necesidad de mantener plataformas universales y estables en las que poder desarrollar las nuevas formas de comercio y de circulación de la información.

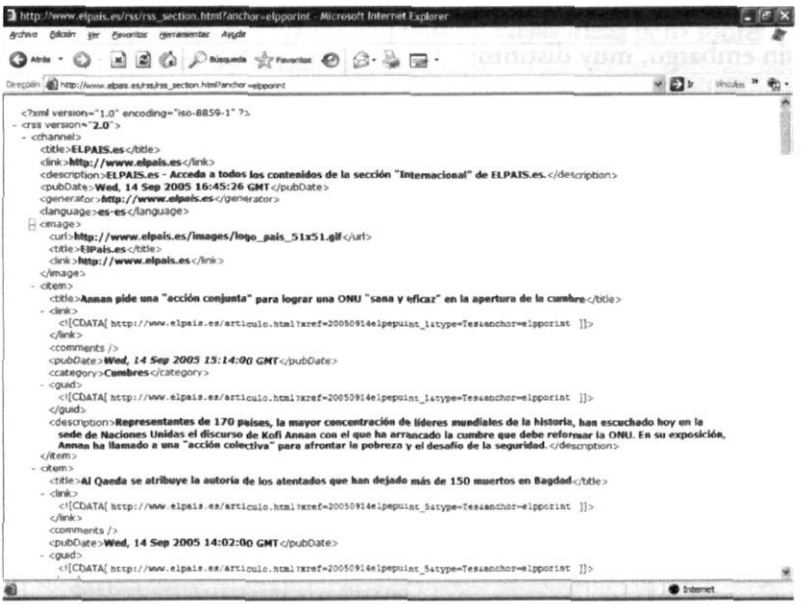
Una de las actividades a las que con más ahínco se ha dedicado el Consorcio W3C en los últimos años es al desarrollo de un nuevo estándar más estricto que el seguido hasta ahora por html y que sirva como sintaxis general para el desarrollo de cualesquiera otros lenguajes específicos. Esta herramienta se conoce por las siglas XML obtenidas a partir de la expresión **EX**tensible **M**ark-up **L**anguage. Hay quienes consideran esta herramienta como un *metalenguaje* que vuela por encima de todos los posibles lenguajes de red, y también quienes la toman como una sintaxis carente de significado. Sea como fuere hoy por hoy sólo marca las convenciones a las que debe ceñirse cualquier lenguaje que quiera tener propagación en la red.

Lo que ahora me interesa destacar de XML es su filosofía. Se trata de un estándar para diseñar etiquetas, es decir, de un conjunto de convenciones acer-

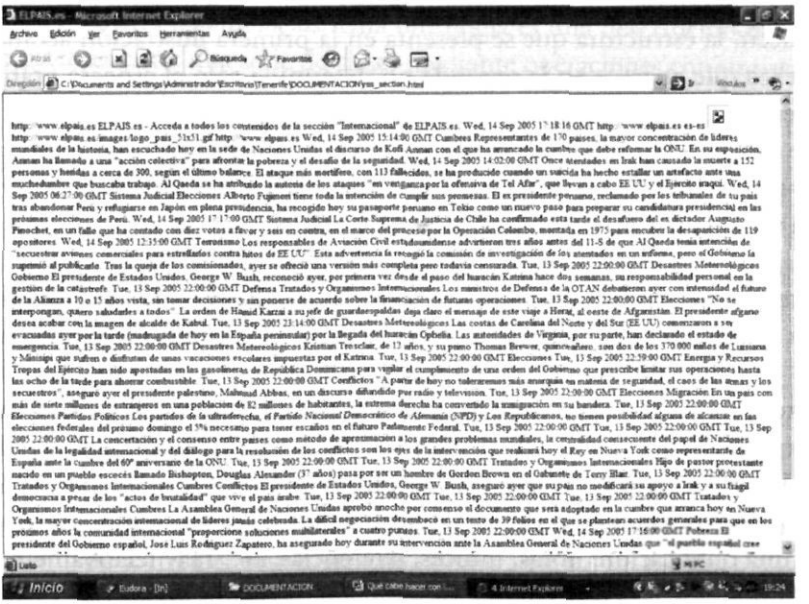
ca de cómo aplicar estructura a un documento. Insistir en la diferencia entre lo que podría denominarse el *contenido textual* de un documento y su *estructura* supone un cambio de actitud con respecto a algunas de las pautas seguidas hasta ahora cuyas consecuencias aún están por llegar.

Una buena forma de entender en qué medida XML es una herramienta puramente sintáctica es considerar el siguiente ejemplo. La etiqueta <a>... es válida para XML, pero carece de significado. No incorpora ninguna acción al texto que acota. En el momento en que se fija un significado para esa etiqueta —centrar en la página el texto afectado, por ejemplo— se obtiene un *lenguaje de marcas*. Me resisto a pensar en un lenguaje de marcas como en un lenguaje de programación porque ciertamente no lo es. Un texto construido con un lenguaje de marcas determinado puede ser ejecutado a continuación en una aplicación diseñada con el fin de leer las etiquetas citadas y realizar las acciones correspondientes. Es posible que una aplicación interprete una etiqueta de forma no totalmente idéntica a otra —como a menudo sucede en los navegadores de Internet— por la sencilla razón de que cada programador puede ceñirse más o menos al estándar marcado por el lenguaje de marcas sobre el que opera.

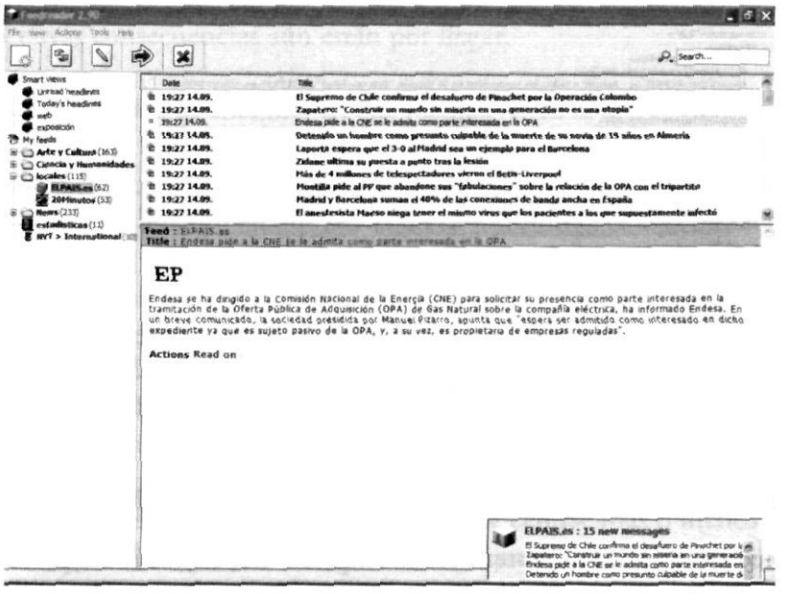
Comentaré sólo algunos ejemplos de lenguajes de marcas basados en XML. RDF es un lenguaje que, según declara la oficina española del Consorcio W3C «proporciona información descriptiva simple sobre los recursos que se encuentran en la Web y que se utiliza, por ejemplo, en catálogos de libros, directorios, colecciones personales de música, fotos, eventos, etc.» Una de las aplicaciones más logradas de este estándar es el lenguaje RSS que ahora se aplica a la denominada sindicación de contenidos. Muchas páginas web, pero principalmente aquellas ligadas a medios de comunicación u organismos que editan información de forma regular, han incorporado en su estructura contenidos rss. La existencia de este servicio se suele identificar mediante una etiqueta como  la cual enlaza a las direcciones desde las que se ofrece. Una página escrita en este lenguaje puede presentar el siguiente aspecto:



Un lector html, al no entender las etiquetas del estándar rss presentará el documento ignorando todas las acciones que afectan al texto, lo que arroja el siguiente resultado:



Cuando el documento es leído por un intérprete de rss el resultado obtenido es, sin embargo, muy distinto:

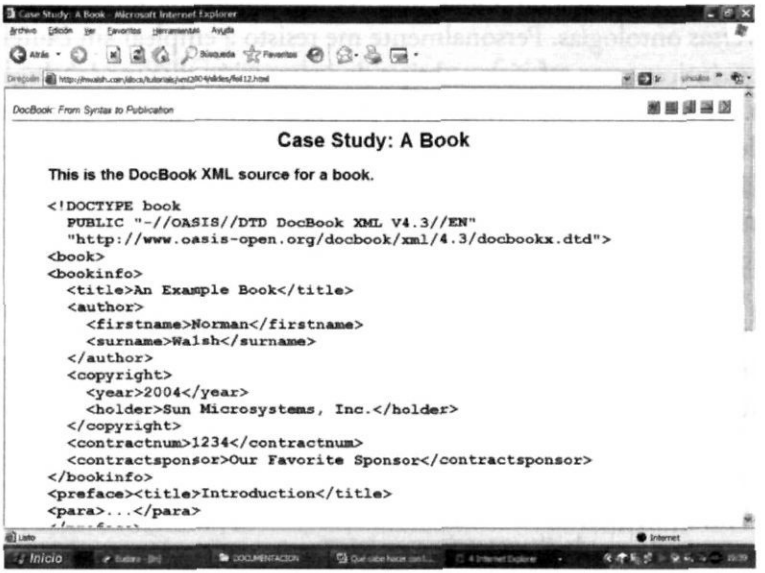


Las acciones que se incorporan en un documento rss no se limitan, como podría desprenderse de estos ejemplos, a una mera presentación gráfica del texto. Es decir, la estructura que se presenta en la primera ilustración acompañando al texto que aparece en la segunda no determina sólo el aspecto gráfico que se presenta en la tercera. Aparte de eso indica a la aplicación que ejecuta la página que renueve la información agregada en una ventana auxiliar emergente —pop-up— como la que se ve en la esquina inferior derecha de la tercera figura cada vez que la página original es modificada.

Este ejemplo indica con bastante claridad cuál es la potencia de los documentos etiquetados con respecto a las acciones que se pueden incorporar en ellos.

Otro ejemplo de lenguaje basado en XML es Docbook. Este caso tiene interés porque el proyecto hace referencia a un procesador de texto ajustado al estándar XML. Las etiquetas ofrecen una amplia variedad de formatos tipográficos y multitud de funciones destinadas a aportar la estructura que generalmente acompaña a una publicación: notas, índices, referencias, etc. Hay relativamente pocas aplicaciones basadas en XML que miren en la dirección de los procesadores de

texto. Y menos aún que lo hagan recuperando viejas ideas pero sobre un soporte que, como XML, es universal.



El último de los lenguajes basados en XML que quiero comentar es OWL. OWL es, con diferencia, el más ambicioso de los lenguajes basados en XML. Su objetivo es el diseño de *ontologías*. Una ontología es una descripción de las relaciones existentes entre una serie de clases mediante operaciones conjuntistas básicas. Así podemos leer lo siguiente:

> OWL distinguishes six types of class descriptions:
>
> 1. a class identifier (a URI reference)
> 2. an exhaustive enumeration of individuals that together form the instances of a class
> 3. a property restriction
> 4. the intersection of two or more class descriptions
> 5. the union of two or more class descriptions
> 6. the complement of a class description

Mediante estos elementos es posible incorporar en un documento, generalmente destinado a circular por la red, una gran cantidad de información sobre la estructura de los conceptos que incorpora.

Se supone que una red apoyada en un entramado de ontologías podría permitir a un agente artificial comportarse de modo eficiente manejando el inmenso volumen de conocimiento estructurado que ha sido volcado en la red a través de las diversas ontologías. Personalmente me resisto a emplear sin cautelas el término «semántico» para referirme al tipo de interacción que tendría lugar en este contexto. La apelación a la «semántica» parece esconder un uso altamente oportunista de un término que implica más cosas de las que realmente hay en juego, Prefiero, si es posible, pensar en este tipo de mecanismos como en *sistemas estructurados de datos* capaces de desplazarse de un término a otro o de una tarea a otra empleando la información suministrada por esa estructura.

Es previsible que el desarrollo de lenguajes basados en XML continúe su expansión en los próximos años y veamos cómo cada vez se abren más líneas de investigación. Pero, ¿qué hay de todo esto en la actualidad?

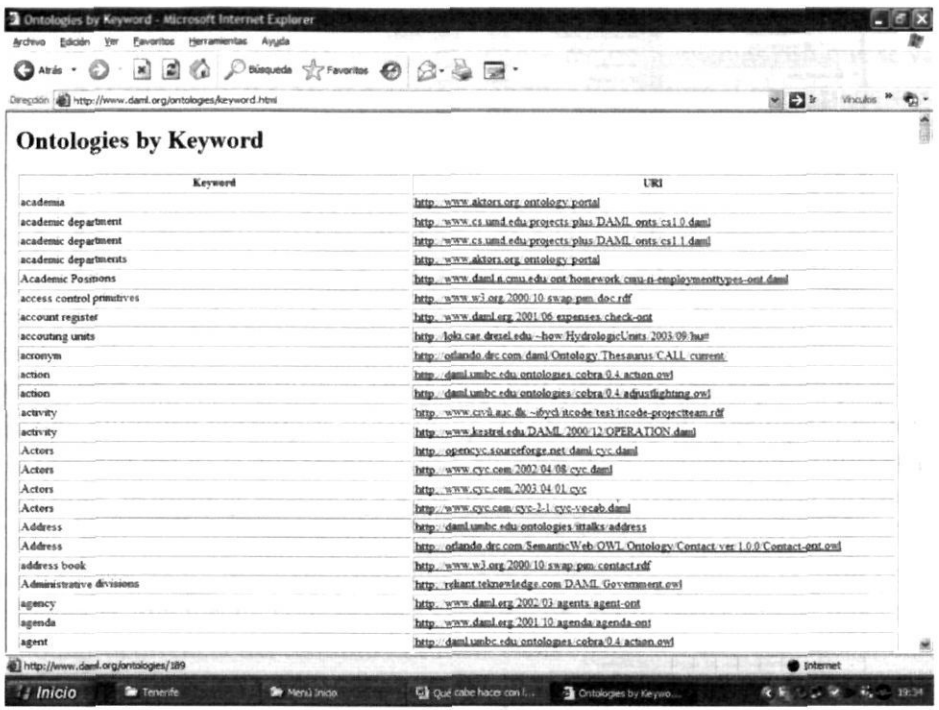
5. La Web semántica

El desarrollo práctico alcanzado por el lenguaje RSS y los servicios de sindicación de contenidos supone un importante espaldarazo para la filosofía en que se apoyan los documentos estructurados. Parece fuera de toda duda que incorporar a los documentos información suficiente sobre la forma en que se encuentran estructurados sus contenidos y qué hacer con ellos supone un considerable avance sobre una concepción puramente visual y finalista de la red. La situación del proyecto estrella de la Web semántica, que no es otro que la incorporación de ontologías en la red en número suficiente como para ampliar su eficiencia en la dirección sugerida por Berners-Lee, es, sin embargo, muy distinta. Para empezar, es muy difícil hacerse una idea siquiera aproximada del estado real de la cuestión.

Existe un gran volumen de documentación relativa al desarrollo de la Web semántica como se puede comprobar recorriendo los enlaces e informes del W3C. Pero tras un escrutinio de todo ese volumen de datos se obtiene la impresión de que existen más sombras que luces en su implantación real. Analizaré este problema centrándome en el que parece el componente más básico de este proyecto: las búsquedas semánticas. Se supone que una búsqueda semántica es aquella que se sirve de la ontología que subyace a un documento para mejorar

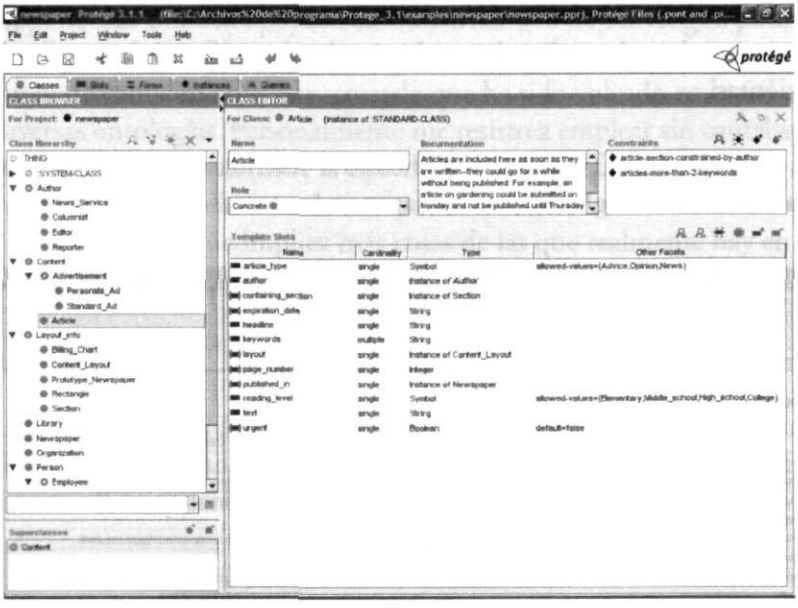
la identificación de la información relevante. Existen numerosas ontologías disponibles ya en la red y aplicaciones destinadas *ex profeso* al desarrollo de estas herramientas.

En el consorcio W3C se pueden encontrar referidas con sus correspondientes enlaces 282 ontologías distintas relativas a los temas más diversos.

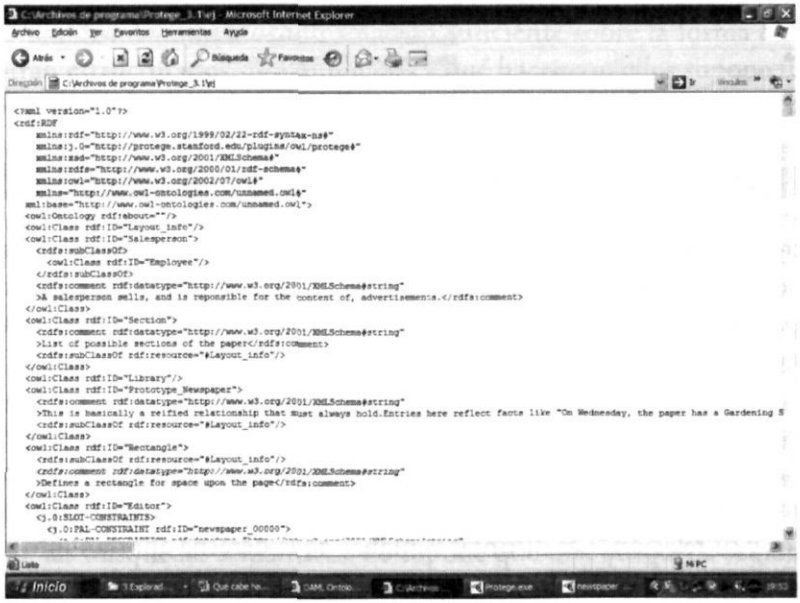


¿Pero qué es y qué aspecto tiene una ontología? El mejor modo de apreciar en qué consiste esta herramienta es trabajar con alguna de las aplicaciones diseñadas para construir ontologías. La más famosa de todas ella es, casi con certeza, Protégé.

La siguiente ilustración muestra la ontología denominada *newspaper* en la que se incluye información relevante para desplazarse por el campo semántico del término *periódico*. En la columna de la izquierda de muestran los conceptos incluidos y su relación. Las distintas ventanas de la derecha aportan información sobre los conceptos marcados.



El aspecto que esta ontología tiene *por dentro* es muy distinto. Se trata, como se ve en la siguiente ilustración de un documento etiquetado generado en lenguaje OWL siguiendo los estándares de XML.



Una vez alcanzado este punto parece que lo que corresponde es explicar de qué forma se aplica una ontología a un documento html para que se convierta en una auténtica página con contenido semántico. Sin embargo, poco podemos aportar ahora salvo vagas intuiciones. Son pocas las páginas de incorporan este tipo de recursos y no parece fácil localizar herramientas de búsqueda que actúen sobre ellas. Parece, en definitiva, que a fecha de hoy el proyecto semántico se encuentra aún dentro del laboratorio.

Sí es posible, no obstante, imaginar algunos de los problemas con que se va a enfrentar este proyecto. Su comentario conduce finalmente al objetivo de este estudio: la identificación de las estrategias que puedan resultar más beneficiosas de cara a la previsible evolución de la red.

En primer lugar, no queda nada claro cuál es la relación que guarda una ontología con el contenido textual de un documento. Las categorías que figuran en una ontología pueden ser clases o instancias de esas clases. Protégé permite elaborar ontologías con clases vacías, clases instanciadas o con otras que comparten instancias. Si la ontología que acompaña a una página actúa como un paquete adjunto o incorporado en una sección aparte, su forma de operar será completamente distinta de la que tendría si se distribuye como una anotación sobre ese contenido. Podríamos referirnos a estas dos estrategias como *ontologías de bloque* o *anotadas*. Una ontología de bloque se incorpora en la página para ofrecer información general sobre los términos que luego aparecen en ella. Si se indica que «El País» es una instancia de la clase «Periódico», esa información se aplicará a cualquier ocurrencia del término «El País» en esa página. Una ontología anotada utiliza las clases para calificar instancias. No hace falta especificar en un espacio separado que El País es un miembro de la clase periódico. Basta con etiquetar las ocurrencias de El País como instancias de Periódico en el texto del documento. Abundaremos en ello más adelante.

¿Qué importancia puede tener esto para el futuro semántico de la red? Si las ontologías indican todas sus instancias comportándose como bloques, el valor semántico del documento dependerá de que en su contenido se utilicen *exactamente* los términos que se emplean en la ontología. Si voy a hablar de un periódico de nueva creación simplemente tendré que incorporarlo en la ontología actuando directamente sobre ella. Al menos si deseo que el nombre de ese nuevo periódico adquiera valor semántico en el documento. Por otra parte, es obvio

que al actuar de este modo el idioma en que se redacte el texto ha de guardar completa coherencia con aquel en que se ha elaborado la ontología.

En segundo lugar, y con independencia de cualquiera de las opciones anteriores, parece claro que el establecimiento de una web semántica tenderá a crear *comunidades semánticas o de conocimiento*. La eficacia de una búsqueda dependerá del número de documentos que compartan la misma ontología, a menos que las búsquedas semánticas se conciban de entrada como tareas circunscritas a un cierto ámbito temático o como he dicho antes, a una determinada comunidad semántica.

En tercer lugar, no hay una idea clara de cuál pueda ser la forma de elaborar consultas en una comunidad semántica. Lo cierto es que la experiencia con la Web clásica ha llevado a que desarrollemos una especie de *lógica de la consulta* que maximiza los logros. Nuestras preguntas se orientan mediante palabras claves con las que ya cuentan los diseñadores —aunque esto nos pueda jugar a veces malas pasadas.

Todo parece indicar la existencia de problemas que sitúan el proyecto semántico en una peculiar encrucijada. ¿Estamos, en definitiva, ante una idea viable, o se trata de un callejón sin salida que tardará más o menos tiempo en agotar su encanto?

6. Nuevos escenarios

Volvamos, tal y como prometimos líneas atrás, al apartado 2 de este trabajo. En él se trataba la situación de los documentos de texto alcanzado la conclusión de que su presentación electrónica no parecía suministrar ventajas sobre el soporte papel debido a una filosofía claramente equivocada. La situación de los documentos html no es, como puede verse, muy distinta. Y la causa parece la misma en ambos casos: la tendencia a elaborar documentos finalistas ideados para ser vistos por nuestros ojos tal y como haríamos sobre un soporte clásico. El problema es más serio en el caso de los documentos de texto como se desprende del hecho de que no existan buscadores documentales[4] de extensión y uso comparables a Google. Las bibliotecas .doc son, como parece evidente, carne de impresora.

[4] No digo que en el ámbito de la bibliteconomía y documentación no existan técnicas para localizar texto. Quiero dar a entender más bien la ausencia de aplicaciones que puedan ser ejecutadas sobre bibliotecas de documentos de texto con cierta eficiencia.a

El estudio de los problemas a que se enfrenta la Web semántica no permite que seamos del todo optimistas. Hay numerosos problemas y no se aprecian líneas de desarrollo claras. Domina, en definitiva, la incertidumbre.

Por ello quizá sea bueno intentar aclarar la naturaleza del problema al que en el fondo nos enfrentamos. La *crisis de crecimiento* que parece haber alcanzado el uso de soportes documentales informáticos -.doc y .html, y sus derivados- tiene que ver ante todo con la recuperación y gestión de la información vertida en esos documentos. Mientras el volumen de datos almacenado en estos soportes se mantuvo por debajo de una cierta cantidad crítica, la forma de recuperar la información contenida nunca pudo ser vista como asunto prioritario. Hemos aprendido a conformarnos con estrategias de recuperación de datos relativamente primitivas cuyas posibilidades ya no pueden ser llevadas mucho más lejos. Y este es el problema que nos ha tocado en suerte: el intentar describir, y si es posible diseñar, sistemas de recuperación de datos universales y eficientes.

Mi propuesta, alcanzado este punto, es que tomemos el cambio de filosofía planteado por Berners-Lee e imaginemos con cierta audacia el tipo de servicios que nos gustaría obtener de todo ello. Creo que sólo entonces podremos responder con cierta convicción a la pregunta que se plantea en este trabajo: ¿qué decisiones estratégicas debemos o podemos adoptar ante el desarrollo de la Web y las tecnologías asociadas?

Voy a partir del problema que me parece más severo que es el de la recuperación de información a partir de documentos tipo .doc. Lo que se pueda decir de ello tiene igual aplicación a los documentos .html ya que en la actualidad no podemos verlos como objetos realmente muy distintos, salvo por el uso de que ellos se hace.

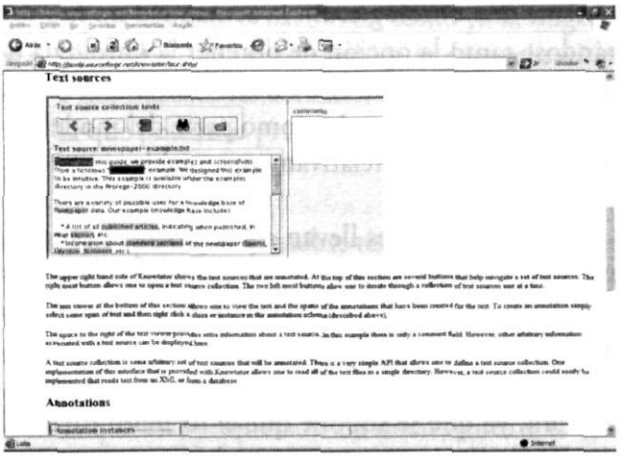
El estudio de aplicaciones para recuperar información de documentos de texto queda en la actualidad bajo el dominio de los laboratorios de *Lingüística Informática*. La filosofía seguida mayoritariamente por este tipo de equipos es la de fabricar herramientas que puedan emplearse sobre textos que, a todos los efectos, muy bien podrían estar en papel. Se realizan complejas tareas para establecer los campos semánticos —lematización— de los términos frecuentes y a ello se añade algo similar a una ontología. Enfrentarse con cierta garantía de éxito a recuperaciones semánticas de la información desde esta perspectiva supone, como

fácilmente podemos imaginar, un trabajo inmenso soportado además por equipos de investigación numerosos y altamente especializados. Tiene la ventaja, eso sí, de que una vez elaborada la lematización y la ontología a la que se aplica no importa el volumen de documentos que se puedan llegar a incorporar en la biblioteca de búsqueda. Pese a que no tengo nada en contra de esta forma de ver las cosas creo que es justo decir que se aferra a la vieja filosofía del formato papel. Por un lado se siguen elaborando textos con formatos finalistas —impresos— para que luego, por otro, sean tratados por equipos de especialistas que en un plazo más o menos dilatado ofrecerán versiones tratables sobre las que efectuar búsquedas avanzadas.

Pero existe otra posibilidad que claramente se sigue de la tendencia que hemos identificado en el desarrollo de los lenguajes de red. Se trata de aprovechar las posibilidades de generar texto etiquetado *en origen*. Esta idea no es del todo novedosa ya que herramientas como DocBook o en cierto modo el propio entorno LaTex actúa de ese modo, por no hablar de los viejos procesadores de texto que eran, en realidad, editores de texto etiquetado o como diré a partir de ahora *anotado*. Lo que sucede con todas estas herramientas es que se orientan al formato tipográfico pero no a la estructura semántica del texto. Se supone que ésta viene dada por la propia lengua y no por ninguna superestructura que se pueda imponer al texto.

Tiene sentido, por tanto, no perder de vista la posibilidad de participar en la elaboración de una nueva generación de editores orientados a la producción de documentos semánticamente anotados. En la actualidad sólo he podido identificar una iniciativa en esta dirección que además sólo puede considerarse en fase de desarrollo. Se trata de un proyecto asociado a Protégé denominado Knowtator. Esta aplicación se ejecuta en Protégé y permite, al menos en teoría, emplear las ontologías disponibles en ese mismo entorno para anotar texto. Son muchas las

dudas que surgen acerca del modo de operar de este procesador dada la dificultad de hacerlo funcionar sin considerables conocimientos acerca de las aplicaciones que lo soportan. Adoptaré, al igual que antes, la filosofía de imaginar qué es lo que nos gustaría tener si ello estuviera en nuestra mano.



La filosofía del texto anotado adjudica al usuario tanto la elección de las etiquetas que marcan el texto, como el número de veces que éstas se emplean. El resultado puede ser un texto profusamente anotado, o no. Pueden marcarse todas las instancias de un término o sólo algunas. Puede que la anotación resulte coherente o claramente incoherente. Todo ello queda, al menos en principio, en manos del usuario, redefiniéndose de paso su papel, es decir, el nuestro.

Podemos imaginar muchas opciones destinadas a mejorar la amigabilidad de la técnica de anotación semántica de un texto. Mencionaré solo algunas. En primer lugar, hay que ser conscientes de que la anotación de texto se ajusta a un formato del tipo clase/instancia. Una etiqueta es siempre una clase y el bloque al que se aplica es siempre una instancia. Es dudoso que Knowtator opere de este modo, ya que al heredar la estructura OWL las instancias están ya incluidas en la ontología, forman parte, por así decirlo, del modelo. El esquema que se sigue aquí hace que la ontología sólo esté formada por clases —las etiquetas disponibles— mientras que las instancias son el modelo suministrado por el texto y la acción del usuario. El sistema seguido por la mayoría de los interfaces de texto actuales que actúan de forma inteligente resulta aquí claramente pertinente. El procesador almacenaría las instancias que cada etiqueta ha marcado sugiriendo en la próxima aparición de esa instancia la inclusión automática de la etiqueta. Otra opción nada desdeñable es permitir que al final de la edición del texto sea el propio procesador el que procede a rellenar las etiquetas que falten pidiendo confirmación al usuario.

Al igual que sucede con el desarrollo de ontologías basadas en OWL —DAMM+OIL, etc.—se plantea el problema de tomar alguna decisión con respecto al lugar en el que se ubica la ontología aplicada al texto. Incorporarla en el propio documento es, como se puede imaginar, la peor de las opciones. El otro problema ligado a la composición de texto anotado es el lugar desde el que se obtiene una ontología. En la actualidad no hay una tendencia definida, apuntándose tanto la opción de obtener la ontología desde un banco o repositorio como de componerla uno mismo. Limitar el alcance de una ontología a las clases dejando las instancias como parte del modelo permite, por otra parte, hacer de éstas un recurso relativamente estable.

Estas reflexiones llevan a imaginar una combinación de factores bastante novedosa. Parece obvio que para poder anotar texto tiene que existir una cierta coherencia entre el procesador de texto y las ontologías que suministran las etiquetas disponibles. El componente más valioso de esta ecuación es, a buen seguro, el relativo a los medios para anotar texto, no el correspondiente a su edición, ya que se supone que el formato final del documento sigue el estándar XML. Creo, por tanto, que debemos estar atentos a la posibilidad de constituir *comunidades de conocimiento* a las que el usuario accede para tomar de ellas las ontologías que en cada caso precisa. La forma de adscribirse a una de estas comunidades es mediante el método de *sindicación* gracias a lo cual el usuario recibe acceso a una biblioteca de ontologías, un procesador de texto estándar y, como ya veremos, un servicio de búsqueda semántico. El texto anotado contiene las correspondientes referencias a las ontologías empleadas y al lugar de sindicación donde deben encontrarse. Esto permite una cierta universalidad de la información.

He hablado de una *comunidad de información* para referirme al servicio que soporta una biblioteca de ontologías. Es obvio que este sitio debería ocuparse igualmente del mantenimiento de sus ontologías, de su renovación, del estudio de su coherencia y de sus posibles relaciones. Aquí podríamos encontrar en campo de desarrollo para las técnicas formales de tratamiento de la información bastante insospechado.

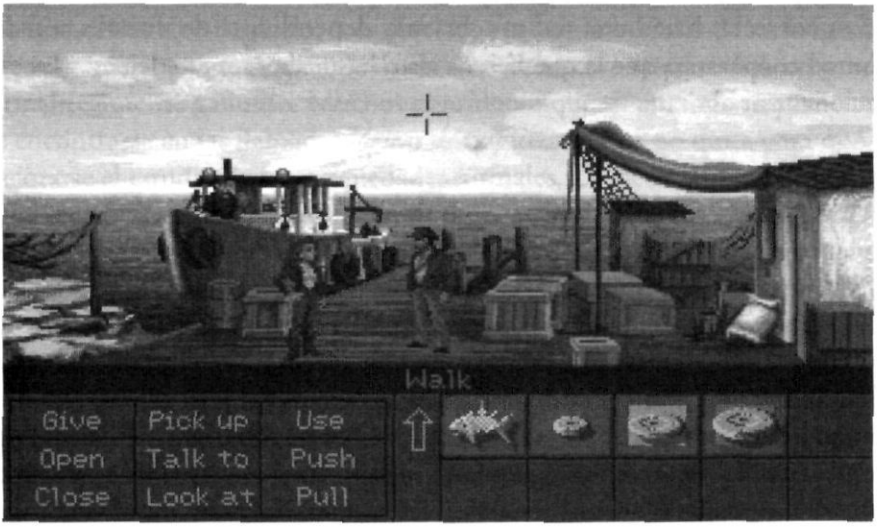
El escenario que dibujo no está destinado a crear textos para cualquier fin. No se me ocurre qué interés podría tener anotar una carta personal u otro recurso parecido. Pienso, más bien, en lo que se suele denominar documentación cor-

porativa. Es decir, aquella que con más probabilidad tiene que ser consultada para extraer de ella información relevante para una comunidad u organismo. Actas de reuniones, documentos de trabajo, normativas y disposiciones legales son firmes candidatos a este tipo de tratamiento.

El último elemento que forma parte de esta ecuación es, como ya he sugerido, los buscadores semánticos o avanzados —he visto también la denominación de *buscadores de 2.ª generación*—. Los buscadores actuales, tipo Google, incorporan la conocida ventana sobre la que podemos probar cualquier tipo de combinación de términos.

Es obvio que un buscador diseñado para trabajar sobre texto anotado no puede seguir una estrategia tan generalista. Lo primero que debe hacer este buscador es situarse ante la biblioteca de textos que debe analizar. No hay por qué pensar que todos ellos compartan la misma ontología, ni siquiera que éstas procedan de la misma comunidad o servicio. Lo que importa es que el buscador sepa en cada momento la ontología con la que está etiquetado un documento y el uso que de ellas puede hacer.

Una búsqueda compleja no tiene por qué venir expresada por una intrincada instrucción elaborada en el lenguaje ordinario del usuario e inscrita en la correspondiente ventana del buscador. Disponer de etiquetas puede simplificar mucho



la cuestión al permitir combinaciones complejas regidas por reglas más o menos estrictas. Pienso en el tipo de instrucciones o diálogos que suelen emplearse en los juegos interactivos con los que tanto pudimos disfrutar hace ya algún tiempo. Esta posibilidad abre una interesante vía de estudio puramente teórico. ¿De qué forma podemos estructurar consultas complejas a partir de las categorías dadas por una ontología? Se trata, en definitiva, de estudiar una especie de *lógica de las consultas* que nos permita acotar el terreno de lo que es coherente preguntar y lo que no.

Siguiendo esta intuición no es muy difícil imaginar el acceso a preguntas de relativa enjundia como las siguientes: «¿Qué Departamentos votaron a favor de pintar de marrón los espacios comunes?» o «¿Por qué se eliminó al Sr. X del concurso Y?»

Lo dicho podría bastar para que nos hagamos una idea del tipo de tendencias que conviene no perder de vista y en las cuales aún podríamos participar de algún modo. Tengo también la impresión de que esta etapa de crisis –crisis de crecimiento en cualquier caso- conduce a una nueva forma de entender nuestra relación con las máquinas en las que trabajamos. Y con ello del propio concepto de «usuario». No se trata de que seamos capaces de realizar complejas tareas informáticas ocupando parte del lugar que los expertos han tenido hasta ahora, sino de que vertamos en nuestros documentos mucha más información de la que hasta ahora nos hemos preocupado por incorporar. Caminaríamos, si esta intuición es correcta, hacia una red mucho más dependiente de nuestra actitud y voluntad cooperativa que la que hemos visto hasta ahora, pero dejemos las especulaciones por ahora.

7. Decisiones estratégicas

No me gustaría terminar dejando que mis conclusiones aparezcan desperdigadas entre consideraciones más o menos oportunas. En esta sección final me voy a preocupar simplemente de recogerlas y agruparlas enumerando el tipo de decisiones que a mi juicio podemos adoptar. No pretendo que las pocas líneas que restan sean entendidas como un listado de deberes o recomendaciones expertas. Tal cosa me parecería una profunda pedantería y una falta de respecto con el lector. Ruego que no se entienda así. Mi enumeración lo es tan solo de un lis-

tado de posibilidades que yo mismo me dispongo poner a prueba. Se trata de informar de un plan de viaje que quizá otros estén iniciando también, por el mismo punto, o por otro. Si podemos quedar en algún lugar del camino, mejor.

Hay una primera parte referida a la adquisición de conocimientos relevantes en materias aparentemente alejadas de nuestro cometido como filósofos. Puesto que no considero que mi condición de tal me aparte de nada empezaré por indicar qué tareas me parecen más urgentes en este sentido. Hay mucho por hacer en el diseño de ontologías. De hecho, casi todo está por hacer en este terreno. Conviene pues acercarse a los lenguajes que han sido propuestos para este fin. En la actualidad todo parece apuntar al posible predominio del lenguaje OWL, pero la velocidad de las iniciativas es tal que lo único que cabe es mantenerse cerca de los puntos de información que en la actualidad pueden marcar tendencia. Pienso sobre todo en las páginas del consorcio W3C. Resulta curioso ver cómo nuestra formación puede ser en estos casos de gran ayuda, anticipando problemas que, siendo triviales en nuestra tradición, resultan extraños para aquellos que desarrollan las herramientas informáticas que han de traducir estas iniciativas. Adquirir conocimientos sobre los modos en que el concepto de ontología ha adquirido una segunda vida no queda fuera de nuestro alcance: quizá cambia el lenguaje, pero no los problemas de fondo.

No pretendo que nuestras decisiones se limiten tan sólo al seguimiento, crítico quizá, de aquello que otros hacen. Hay un terreno en el cual podemos y debemos decir algo llegando, si es posible, a tomar la iniciativa. Los lenguajes de etiquetas son, como se desprende de todo lo dicho con anterioridad, un producto formal realmente genuino. Muchos consideran que su ancestro inmediato puede encontrarse en las llamadas *lógicas descriptivas* lo que no quita para que deba mejorarse el estudio de sus propiedades formales. De hecho, no está claro cuáles puedan ser estas propiedades. Oímos decir en ocasiones que HTML presenta problemas de coherencia que son subsanados por XML aunque este lenguaje debería ser considerado mejor como un metalenguaje. Ni el concepto de *coherencia* empleado aquí me parece claro, ni me lo parecen tampoco las alusiones al carácter metalingüístico de un cierto lenguaje de marcas. ¿No merece la pena decir algo de todo esto? Esa especie de extraño terreno a medio camino entre un lenguaje de programación —las etiquetas apuntan a acciones— y un modelo del mundo que hoy ocupan los lenguajes etiquetados que emplea la red constituyen un objeto formal al que la lógica académica no parece estar prestando toda la

atención que debiera. Si en el fondo no es sino uno de nuestros hijos, ¿por qué no hacernos cargo de él?

Sin abandonar el terreno del análisis teórico del problema de los lenguajes de red y su uso semántico, me preocupa y mucho el asunto de la coherencia entre ontologías. Se insiste con frecuencia en que el desarrollo de ontologías en la Web semántica es algo que queda, como todo proyecto que realmente participe de la filosofía de Internet, al libre criterio de sus usuarios. Esto entra en conflicto, sin embargo, con algo que parece inherente a un uso eficaz de las ontologías, a saber, su coherencia relativa. Dos comunidades de usuarios agrupadas en torno a ontologías incoherentes, pero referidas al mismo dominio, representan un problema que, de hecho, ya se da en la actualidad. ¿Es posible diseñar procedimientos eficaces para volcar —mapear— una ontología en otra? ¿Podemos diseñar mecanismos de integración viables, *superontologías*, por así decir?

La última cuestión que considero al alcance de nuestros méritos y posibilidades tiene que ver con la lógica de las búsquedas o quizá mejor aún, de las preguntas. Lo que en su día fue la *lógica erotética* parece tener ahora también un segundo momento de gloria. ¿Qué cuestiones pueden plantearse en un dispositivo de búsqueda y qué respuestas son las aceptables? ¿Deben estructurarse rígidamente las preguntas, o debemos permitir un amplio margen al lenguaje natural con lo que ello supone?

El abanico de cuestiones teóricas que el desarrollo de la red propone no termina aquí como es obvio. Estas son las que yo encuentro más prometedoras y ante todo más propias de nuestra formación y vocación fundamental que es, no se olvide, la filosofía.

Tampoco creo que debamos desentendernos de la construcción de herramientas —hablo de software— que traduzcan nuestras ideas, aunque admito que esta actitud es más bien personal y quizá poco apreciada en el ámbito de la Filosofía oficial. Nunca he simpatizado en exceso con divisiones tajantes entre el ámbito de la teoría y el de la práctica. No creo que, como filósofo, nada me esté vedado. Y mucho menos la posibilidad de enseñar, por medio del lenguaje que sea, el resultado de mis hipótesis. ¿Por qué dejar esta tarea a profesionales poco o nada dispuestos a participar de nuestras motivaciones más genuinas?

Bibliografía

BERNERS-LEE, T., HENDLER, J. y LASSILA, O., 2001: «The Semantic Web», *The Scientific American.* Mayo 2001.

DACONTA, M.C., OBRST, L. J. y SMITH, K.T., 2003 : The Semantic Web. Wiley Publishing, Inc.

DOMINI F. M., LENZERINI, M., NARDI, D., NUTT, W. The complexity of concept languages. Information and Computation, 134:1-58, 1997.

IAN HORROCKS and PETER F. PATEL-SCHNEIDER. Three theses of representation in the semantic web. In Proc. of the Twelfth International World Wide Web Conference (WWW 2003), pages 39-47. ACM, 2003.

IAN HORROCKS and PETER PATEL-SCHNEIDER. Reducing OWL entailment to description logic satisfiability. J. of Web Semantics, 1(4):345-357, 2004.

W3C OWL Web Ontology Language. http://www.w3.org/TR/owl-features

OWL : The Web Ontology Language. http://www.w3.org/2001/sw/WebOnt/

W3C RDF/XML Syntax Specfication. http://www.w3.org/TR/rdf-syntax-grammar

W3C Semantic web. http://www.w3.org/2001/sw/