INSTITUTO POLITECNICO NACIONAL
ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
(UNIDAD ZACATENCO)
SECCION DE ESTUDIOS DE POSTGRADO E INVESTIGACIÓN
POSTGRADO DE ESTUDIOS EN INGENIERÍA DE SISTEMAS
DOCTORADO ENCIENCIAS EN INGENIERÍA DE SISTEMAS

BUSCADORES SEMÁNTICOS PARA LA GESTION DEL CONOCIMIENTO

TRABAJO QUE:

COMO EXAMEN DOCTORAL

PRESENTA:

M. en C. GERSON VILLA GONZÁLEZ

DIRIGIDO POR:

DR. LUIS MANUEL HERNÁNDEZ SIMÓN

MEXICO, D.F., JUNIO DE 2011

SIP-14

INSTITUTO POLITÉCNICO NACIONAL

SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

ACTA DE REVISIÓN DE TESIS

En la Ciudad de México, D. F. siendo las 12:00 horas del día 17 del mes de Marzo del 2011 se reunieron los miembros de la Comisión Revisora de Tesis designada por el Colegio de Profesores de Estudios de Posgrado e Investigación de E.S.I.M.E.-ZAC. para examinar la tesis titulada:

" BUSCADORES SEMÁNTICOS PARA LA GESTIÓN DEL CONOCIMIENTO"

Presentada por el alumno:

VILLA	**GONZÁLEZ**	**GERSON**
Apellido paterno	Apellido materno	Nombre(s)

Con registro: | A | 0 | 8 | 0 | 4 | 2 | 4 |

aspirante de:

DOCTORADO EN INGENIERÍA DE SISTEMAS

Después de intercambiar opiniones los miembros de la Comisión manifestaron ***SU APROBACIÓN DE LA TESIS***, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

LA COMISIÓN REVISORA

Director de tesis

DR. LUIS MANUEL HERNÁNDEZ SIMÓN

Presidente

DR. ALEXANDER BALANKIN

Segundo Vocal

DR. AGUSTIN FRANCISCO GUTIÉRREZ TORNÉS

Tercer Vocal

DR. MIGUEL PATIÑO ORTÍZ

Secretario

DR. OSWALDO MORALES MATAMOROS

EL PRESIDENTE DEL COLEGIO

DR. JAIME ROBLES GARCÍA

ESCUELA SUPERIOR DE INGENIERIA MECANICA Y ELECTRICA
I. P. N.
SECCION DE ESTUDIOS DE POSGRADO E INVESTIGACION

INSTITUTO POLITECNICO NACIONAL

SECRETARIA DE INVESTIGACIÓN Y POSGRADO

En la Ciudad de México, D.F., día 25 de Mayo del mes de Mayo del año 2011 el que suscribe Gerson Villa González de Doctorado en Ingeniería de Sistemas Alumno del programa con número de registro ***A080424*** adscrito a la Sección de Estudios de Posgrado e Investigación de la **E.S.I.M.E Unidad Zacatenco**, manifiesta que es autor intelectual del presente Trabajo de Tesis bajo la dirección del **Dr. Luis Manuel Hernández Simón** y cede los derechos del trabajo titulado: "***Buscadores Semánticos para la Gestión del Conocimiento***" al Instituto Politécnico Nacional para su difusión, con fines Académicos y de investigación.

Los usuarios de la información no deben de reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección:

gogoyubarismooth@gmail.com o gvilla@ipn.mx

Si el permiso se otorga el usuario deberá dar el agradecimiento correspondiente y citar la fuente de la misma.

Dr. Gerson Villa González (c)

RESUMEN

La Web Semántica, según ha sido concebida por sus creadores, puede convertirse en la siguiente revolución de la sociedad de la información comparable con la reciente popularización de la Web en Internet. La Web Semántica ofrece contenido online semánticamente anotado de manera formal posibilitando así su procesamiento automático mediante software. Esta nueva Web, diseñada para ser utilizada por software, permitirá que los usuarios humanos puedan delegar tareas complejas a agentes de software o aplicaciones. En los preliminares de esta memoria y haciendo un paralelismo con la aparición de la WWW se identifican algunos retos para la Web Semántica. Entre todos ellos se destaca el reto de la disponibilidad de una masa crítica del contenido semántico como motivación principal de esta tesis.

Este trabajo propone una arquitectura de sistemas de adquisición automática de contenido para la Web Semántica. Su objetivo concreto es proporcionar un marco conceptual para el desarrollo de sistemas para el procesamiento de contenido actual de la WWW y convertirlo en contenido semánticamente anotado para los agentes y las aplicaciones de la Web Semántica puedan procesarlo. Se parte de un estudio del estado del arte de las tecnologías existentes en el área de extracción de información y se presentan algunas aplicaciones existentes con objetivos similares.

En la propuesta de arquitectura se parte del supuesto de la existencia de relaciones entre las estructuras que puedan presentar las fuentes y el uso de distintas tecnologías para la extracción de información. Entre ellas se incluyen: procesamiento de lenguaje natural para fuentes textuales con estructuras de párrafos o frases, procesamiento de aspecto visual para fuentes con estructuras visuales repetitivas (tablas, listas, etc.), uso de expresiones regulares para cadenas bien definidas, etc. La arquitectura propuesta permite el uso de todas ellas en un marco de cooperación basado en un control estratégico personalizable de la tarea de extracción e instanciación de contenido semántico.

Finalmente, se propone la construcción de una aplicación que hace uso de la arquitectura propuesta. Se trata de un portal semántico que, a partir de la presencia de contenido anotado semánticamente, ofrece funcionalidades avanzadas de búsqueda y navegación. El contenido de este portal es extraído de fuentes online y anotado automáticamente usando una implementación de la arquitectura propuesta.

ABSTRACT

The Semantic Web as it was designed by its creators could become the next revolution in Information society, similar to the WWW extension 20 years ago. This, so called, next generation Web was conceived for software instead of humans allowing human users delegating complex tasks to intelligent agents and applications. Making parallel analysis to the exploitation of the current Web this work identifies a set of challenges, mainly focusing on the challenge of the availability of a critical amount of semantically enabled content.

This thesis introduces an acquisition system architecture for generating Semantic Web content. The goal of the described system is to process the current content of the WWW and upgrade it allowing for processing of software agents and applications. The state of the art introduces available technologies for information extraction as well as a set of similar working applications.

The proposal starts with a hypothesis about the relations existing between the degree of structure present in the online sources and the technologies used for information extraction. In this sense, for instance, natural language processing becomes more efficient for texts where whole sentences or phrases are found, layout processing is really worthwhile for strongly visual structures (tables, lists, etc), and text engineering using all of them in a cooperation framework driven by some suitable strategy.

The end of this memory includes a proposal for a Semantic Web application. It is a semantic portal using an implementation of the acquisition system for semantic content provision and allowing for advanced search and browse functionalities according to underlying model.

INDICE

Capítulo 1

Capítulo 2

Lista de Figuras

Lista de Tablas

Capítulo 3

GLOSARIO

Se enumeran algunos términos con los que se trabaja en aplicaciones de la Web Semántica. No se trata de definiciones formales, sino más bien de explicaciones de términos en el contexto de la memoria de la Web Semántica.

Ontologías

Términos relativos a las ontologías.

- ***Clase (Concepto):*** Es una descripción formal de una entidad del universo del dominio. Constituye la pieza básica de estructuración del conocimiento. Es una decisión muy difícil y a veces subjetiva cuando crear un concepto en el modelado de un dominio. Debe ser tomada de acuerdo a los objetivos de la ontología (extraídos de las sesiones con los expertos, preguntas y de estándares existentes en el dominio).
- ***Propiedad (Atributo, Slot):*** Describe en más detalle la clase. Establece que el concepto posee una propiedad que se concretará dando un valor al atributo. Los valores de los atributos pueden ser tipos básicos como cadenas de carácteres, numerales, otros conceptos o instancias.
- ***Restricciones*** sobre las propiedades (Facetas): Son algunas propiedades de las propiedades. Por ejemplo, la cardinalidad que puede tener, si es obligatorio o no, etc.
- ***Axiomas:*** Son reglas que se añaden a la ontología que describen el comportamiento de los conceptos y se establecen sobre los valores de las propiedades. Se suele inferir un valor de un atributo, que no se ha introducido explícitamente, o simplemente que el valor introducido es coherente con las restricciones de la ontología.
- ***Instancia de un concepto:*** Representa un concepto concreto del dominio que se modela. La colección de instancias constituye las bases de hechos (también llamada base de datos, bases de conocimiento) del modelo.
- ***Herencia:*** propiedad de la relación 'es_un', que permite que las clases relacionadas (heredadas) cuenten con los atributos de la clase con la cual se relacionan (clase padre). Normalmente los lenguajes de las ontologías (como RDFS, OWL, etc.) definen algunas relaciones semánticas básicas entre conceptos. La más usada de estas relaciones es: subClassOf.
- ***Derivación:*** En una ontología la relación más usual es la relación de 'es_un' (también llamada: kind_of, is_a, o herencia). Esta relación organiza las clases en un árbol de herencia (permitiendo la herencia múltiple) con la propiedad de herencia. A veces se denomina derivación, y a las clases de nivel inferior, clases derivadas.

- ***Instancia Indirecta:*** Cuando una clase es instancia indirecta de otra, quiere decir que es instancia de alguna de sus clases derivadas.
- ***Clase Abstracta:*** Clase que no permite que existan instancias de ella. Se usa para agrupar conceptos, introducir cierto orden en la jerarquía, pero suelen ser demasiado generales para admitir instancias. Es una restricción dada por el diseñador de la ontología. (Por ejemplo, no tiene sentido crear instancias la clase superior de la ontología).

Creación de contenido semántico

- ***Anotación (Semántica):*** Normalmente las anotaciones se refieren a un texto o documento fuente que se procesa para ser incluido como contenido de la Web Semántica. A la información adicional necesaria para añadir el significado del contenido se le denomina anotación semántica. Las anotaciones pueden o bien ser etiquetadas correspondientes a una ontología que se añaden al texto tratado o bien pueden almacenarse a parte del texto tratado formando una base de hechos o colección de instancias.
- ***Meta-dato***: la definición de meta-dato dice que es un dato sobre el dato. En el contexto de esta memoria se usa como sinónimo de anotación: dato que aporta significado al contenido.
- ***Referencia documental***: Cuando las anotaciones se almacenan independientemente del texto tratado es necesario a veces mantener una relación (o enlace) entre el texto original y la instancia de la ontología. Se le suele añadir un atributo especial a los conceptos de la ontología que albergan un enlace a la parte del texto que dio origen a la propia instancia.
- ***Ocurrencia***: El valor de una referencia documental suele ser múltiple. Por ejemplo, la creación de una instancia de persona puede venir de varios sitios del texto donde aparece su nombre. Todas estas apariciones de nombre se refieren a la misma instancia de persona pero en distintas ocurrencias dentro del texto.

RESUMEN

La Web Semántica, según ha sido concebida por sus creadores, puede convertirse en la siguiente revolución de la sociedad de la información comparable con la reciente popularización de la Web en Internet. La Web Semántica ofrece contenido online semánticamente anotado de manera formal posibilitando así su procesamiento automático mediante software. Esta nueva Web, diseñada para ser utilizada por software, permitirá que los usuarios humanos puedan delegar tareas complejas a agentes de software o aplicaciones. En los preliminares de esta memoria y haciendo un paralelismo con la aparición de la WWW se identifican algunos retos para la Web Semántica. Entre todos ellos se destaca el reto de la disponibilidad de una masa crítica del contenido semántico como motivación principal de esta tesis.

Este trabajo propone una arquitectura de sistemas de adquisición automática de contenido para la Web Semántica. Su objetivo concreto es proporcionar un marco conceptual para el desarrollo de sistemas para el procesamiento de contenido actual de la WWW y convertirlo en contenido semánticamente anotado para los agentes y las aplicaciones de la Web Semántica puedan procesarlo. Se parte de un estudio del estado del arte de las tecnologías existentes en el área de extracción de información y se presentan algunas aplicaciones existentes con objetivos similares.

En la propuesta de arquitectura se parte del supuesto de la existencia de relaciones entre las estructuras que puedan presentar las fuentes y el uso de distintas tecnologías para la extracción de información. Entre ellas se incluyen: procesamiento de lenguaje natural para fuentes textuales con estructuras de párrafos o frases, procesamiento de aspecto visual para fuentes con estructuras visuales repetitivas (tablas, listas, etc.), uso de expresiones regulares para cadenas bien definidas, etc. La arquitectura propuesta permite el uso de todas ellas en un marco de cooperación basado en un control estratégico personalizable de la tarea de extracción e instanciación de contenido semántico.

Finalmente, se propone la construcción de una aplicación que hace uso de la arquitectura propuesta. Se trata de un portal semántico que, a partir de la presencia de contenido anotado semánticamente, ofrece funcionalidades avanzadas de búsqueda y navegación. El contenido de este portal es extraído de fuentes online y anotado automáticamente usando una implementación de la arquitectura propuesta.

ABSTRACT

The Semantic Web as it was designed by its creators could become the next revolution in Information society, similar to the WWW extension 20 years ago. This, so called, next generation Web was conceived for software instead of humans allowing human users delegating complex tasks to intelligent agents and applications. Making parallel analysis to the exploitation of the current Web this work identifies a set of challenges, mainly focusing on the challenge of the availability of a critical amount of semantically enabled content.

This thesis introduces an acquisition system architecture for generating Semantic Web content. The goal of the described system is to process the current content of the WWW and upgrade it allowing for processing of software agents and applications. The state of the art introduces available technologies for information extraction as well as a set of similar working applications.

The proposal starts with a hypothesis about the relations existing between the degree of structure present in the online sources and the technologies used for information extraction. In this sense, for instance, natural language processing becomes more efficient for texts where whole sentences or phrases are found, layout processing is really worthwhile for strongly visual structures (tables, lists, etc), and text engineering using all of them in a cooperation framework driven by some suitable strategy.

The end of this memory includes a proposal for a Semantic Web application. It is a semantic portal using an implementation of the acquisition system for semantic content provision and allowing for advanced search and browse functionalities according to underlying model.

Introducción

En poco más de una década desde su aparición, la WWW (World Wide Web) se ha convertido en un instrumento de uso cotidiano en nuestra sociedad, comparable a otros medios tan importantes como la radio, la televisión o el teléfono a los que aventaja en muchos aspectos. La web es hoy un medio extraordinariamente flexible y económico para la comunicación, el comercio, los negocios, ocio, entretenimiento, acceso a la información y servicios, difusión de cultura, etc. Paralelamente al crecimiento espectacular de la web, las tecnologías que la hacen posible han experimentado una rápida evolución. Desde las primeras tecnologías básicas: HTML[1] y HTTP[2], hasta nuestros días, han emergido tecnologías como CGI[3], Java[4], JavaScript[5], ASP[6], JSP[7], PHP[8], Flash[9], j2EE[10], XML[11] por citar algunas de las más conocidas, que permiten una web mejor, más amplia, más potente, más flexible ó más fácil de mantener. Estos cambios influyen y son al tiempo influidos por la propia transformación de lo que entendemos por WWW. La generación dinámica de páginas, el acoplamiento con bases de datos, la mayor interactividad con el usuario, la concepción de la web como plataforma universal para el despliegue de aplicaciones, la adaptación del usuario, son algunas de las tendencias evolutivas más marcadas de los últimos años.

La evolución de la web no termina aquí ni mucho menos. Son diversos los aspectos susceptibles de mejorar. Entre las últimas tendencias que pueden repercutir en el futuro de la web a medio plazo es que a finales de la década de 1990 surge la visión de lo que se ha dado en llamar la *web semántica* (Berners-Lee, 2001). Se trata de una corriente, promovida por el propio inventor de la web y el presidente del consorcio W3C[12], cuyo último fin es lograr que las máquinas que puedan entender, y por tanto, utilizar lo que la web contiene. Esta nueva web estaría poblada de agentes o representantes de software capaces de navegar y realizar operaciones por nosotros para ahorrarnos trabajo y optimizar los resultados. Para conseguir esta meta, la web semántica propone describir los recursos de la web con representaciones procesables (es decir, entendibles) no sólo por

[1] http://www.w3.org/MarkUp/
[2] http://www.w3.org/Protocols/
[3] http://hoohoo.ncsa.uiuc.edu/cgi
[4] http://java.sun.com
[5] http://www.mozilla.org/js/
[6] http://www.asp.net/
[7] http://java.sun.com/products/jsp/
[8] http://php.apache.org/
[9] http://www.adobe.com
[10] http://java.sun.com/j2ee/
[11] http://www.w3.org/XML/
[12] http://www.w3.org

personas, si no por programas que puedan asistir, representar o reemplazar a las personas en tareas rutinarias o inabarcables para un humano. Las tecnologías de la web semántica buscan desarrollar una web más cohesionada donde sea aún más fácil localizar, compartir e integrar información y servicios para sacar un partido todavía mayor de los recursos disponibles en la web.

La aparición de la WWW se puede situar en 1989 (Abrams 1998, Conolly 2000) cuando Tim Berners-Lee presentó su proyecto de ***"World Wide Web"*** (Berners-Lee 1989) en el CERN (Suiza) con las características esenciales que perduran en nuestros días. El propio Berners completó en 1990 el primer servidor web y el primer cliente y un año más tarde publicó el borrador de las especificaciones del HTML y HTTP.

El lanzamiento en 1993 de Mosaic el primer navegador de dominio público compatible con Unix, Windows y Macintosh por el ***National Center for Supercomputing Applications*** (NCSA), marca el momento en que al WWW se da a conocer al mundo, extendiéndose primero a universidades y laboratorios y en cuestión de meses al público en general, iniciando el que sería su vertiginoso crecimiento. Los primeros usuarios acogieron con entusiasmo la facilidad con que se podían integrar textos y gráficos y saltar de un punto a otro del mundo en una misma interfaz y la sencillez para contribuir a una web mundial.

Por estas mismas fechas se define la interfaz CGI para la generación dinámica de páginas web con lo que se consigue ofrecer información actualizada en tiempo real, enlazar con bases de datos o tener en cuenta entradas del usuario y, más aún, servir como punto de acceso y plataforma para la ejecución de aplicaciones distribuidas. En 1994 miembros del equipo que creó Mosaic desarrollan Netscape, un navegador con sensibles mejoras que contribuye a impulsar la propagación de la web. Este mismo año se celebra el primer congreso internacional de la WWW y unos meses más tarde se constituye el consorcio W3C que desde entonces y presidido por Tim Berners-Lee, se ha hecho cargo de estandarizar las principales tecnologías web. En 1995 Sun lanza oficialmente la primera versión de Java y un año más tarde Netscape presenta JavaScript. Estos lenguajes y otros posteriores permiten que las propias páginas web contengan programas enteros, dando opción a una mayor autonomía respecto del servidor, mayor eficiencia, capacidad dinámica y capacidad de interacción.

La disponibilidad de contenido formal o conocimiento para sistemas de software no es un problema nuevo. Los sistemas basados en conocimiento, y más tarde los agentes inteligentes entre otros, se han enfrentado ya este reto. El éxito alcanzado en el abastecimiento de contenido dependía en cada caso de los sistemas particulares en concreto y del dominio sobre el cual operaban. Cada sistema poseía su propio formato de almacenamiento del conocimiento adaptado a sus propósitos y funcionalidades concretas.

El paradigma de la Web Semántica propone un formato común estandarizado con sintaxis controlada (basada en XML y promovida por el W3C) y con semántica consensuada expresada en modelos formales. Estos lenguajes, a diferencia de los lenguajes actuales de la Web donde se expresa el "cómo" se debe presentar un contendido, permite describir la semántica el "qué" o el "cuál" es el significado del contendido publicado. Estos lenguajes semánticos se organizan en capas permitiendo así diferentes grados de comprensión por parte del software. Además, la Web Semántica está siendo una iniciativa que a nivel mundial engloba a un gran número de centros de investigación, organismos de estandarización, gobiernos y cada vez más empresas de diferentes sectores.

Aunque es sumamente difícil medir el tamaño de la web, se estima que hoy en día unos 10^9 usuarios utilizan la web y que esta contiene del orden de $4x10^9$ documentos, un volumen de información equivalente a entre 14 y 28 millones de libros [Bergman 2001]. Como dato comparativo, la asociación ***American Reserch Libraries***, que agrupa unas 100 bibliotecas en EE.UU., tiene catalogados unos 3.7 millones de libros. La biblioteca de la Universidad de Harvard, la mayor de EE.UU., contiene alrededor de 15 millones de libros. Estas cifras incluyen sólo lo que se ha dado en denominar ***la web superficial*** formada por los documentos estáticos accesibles en la web. Se ha calculado que la llamada ***web profunda,*** constituida por la bases de datos cuyos contenidos no directamente accesibles se hacen visibles mediante páginas generadas dinámicamente, puede contener un tamaño de información varios cientos de veces mayor y de mucha mejor calidad que la ***web superficial*** y crece a un ritmo mayor que ésta [O'Neill 2003]. Se estima que el tamaño de ***la web profunda*** ha superado ya el volumen total de la información impresa existente en todo el planeta.

Hoy casi todo está representado de una u otra forma en la web y, con la ayuda de un buen buscador, podemos encontrar información sobre casi cualquier cosa que necesitemos. La web está cerca de convertirse en una enciclopedia universal del conocimiento humano. Por otra parte, la web nos permite realizar diferentes actividades de nuestra vida diaria con una comunidad, de forma económica y eficiencia sin precedentes. Sin movernos de casa podemos comprar todo tipo de productos y servicios, gestionar una cuenta bancaria, buscar un restaurante, consultar la cartelera, leer la prensa, localizar a una persona, matricularnos en la universidad ó trabajar desde nuestro domicilio.

No obstante este panorama tan favorable, hay espacio para mejoras. Por ejemplo el enorme tamaño que ha alcanzado la web, a la vez que es una de las claves de su éxito, hace que algunas tareas (por ejemplo, encontrar la planificación óptima de transporte, alojamiento, etc., entre todas las posibles para un viaje bajo ciertas condiciones), requieran un tiempo excesivo para una persona o resulten sencillamente inabarcables.

Desarrollar programas que realicen estas tareas en nuestro lugar es enormemente complicado, ya que es muy difícil reproducir y más costoso aún mantener en una máquina la capacidad de una persona para comprender los contenidos de la web tal y como están codificados actualmente.

La asombrosa eficacia de los buscadores actuales tiene también sus límites. Por ejemplo, si queremos conocer la historia de Netscape, los resultados de una consulta como "Netscape history" realizada al momento de escribir este introducción[13] nos informan sobre las herramientas de este navegador, pero no nos dicen nada sobre el origen y evolución del Netscape. Igualmente para averiguar qué organismo se ocupa de estandarizar CGI o en qué fecha apareció la primera versión de Java, necesitaremos realizar varias consultas y leer varios documentos y artículos hasta llegar indirectamente a la respuesta buscada. Si introducimos la palabra "Ketchup" para buscar información sobre el grupo de música del mismo nombre, obtendremos enlaces a restaurantes, recetas, fabricantes, distribuidores y clubes de aficionados al condimento y finalmente lo que buscábamos (posiblemente ni siquiera esto sí es el grupo fuese menos popular). Si buscamos un artículo sobre "J.J. Benítez", encontraremos decenas de artículos de J.J. Benítez pero ninguno que trate de este autor. Si preguntamos sobre estándares XML para la enseñanza ("XML education"), la mayor parte de los resultados se referirán a la enseñanza de XML.

Todos estos ejemplos son el síntoma de una causa común: la falta de capacidad de las representaciones en que se basa la web actual para expresar significados. Los contenidos y servicios en la web se presentan en formatos (HTML) e interfaces (formularios) comprensibles por personas, pero no por máquinas.

En estas condiciones es poco viable automatizar tareas mediante software en substitución del humano. Un programa puede llevar al usuario hasta lugares en la web, generar, transportar, transformar y ofrecer la información a las personas, pero la máquina sencillamente no sabe lo que esta información significa y, por tanto, su capacidad de actuación autónoma es muy limitada.

La web semántica[14] (Berners-Lee, 2001) propone superar las limitaciones de la web actual mediante la introducción de descripciones explicitas del significado, la estructura interna y la estructura global de los contenidos y servicios disponibles en la WWW. Frente a la semántica implícita, el crecimiento caótico de recursos y la ausencia de una organización

[13] Con los continuos cambios de la web y los algoritmos de los buscadores los resultados de estas pruebas pueden variar de un día a otro.
[14] http://www.semanticweb.org/, http://www.ontology.org/

clara de la web actual, la web semántica aboga por clasificar, dotar de estructura y recursos con semántica explícita procesable por máquinas. Actualmente la web se asemeja a un grafo formado por nodos del mismo tipo y arcos (hiperenlaces) igualmente indiferenciados. Por ejemplo, no se hace distinción entre la página personal de un profesor y el portal de una tienda online como tampoco se distinguen explícitamente los enlaces de las asignaturas que imparte un profesor de los enlaces a sus publicaciones. Por el contrario, en la web semántica cada nodo (recurso) tiene un tipo (profesor, tienda, etc.) y los arcos representan relaciones explícitamente diferenciadas (profesor- departamento, libro-editorial).

La web semántica mantiene los principios que han hecho éxito de la web actual como son los principios de descentralización, compartición, compatibilidad, máxima facilidad de acceso y contribución o la apertura al crecimiento y uso no previstos de antemano. En este contexto un problema clave es alcanzar un entendimiento entre todas las partes que han de intervenir en la construcción y explotación de la web: usuarios, desarrolladores y programas de muy diverso perfil. La web semántica rescata la noción de ontología del campo de la Inteligencia Artificial como vehículo para cumplir este objetivo.

Gruber define ontología como ***"a formal explicit specification of a shared conceptualization"*** (Gruber, 1993). Una ontología es una jerarquía de conceptos con atributos y relaciones que define una terminología consensuada para definir redes semánticas de unidades de información interrelacionadas. Una ontología proporciona un vocabulario de clases y relaciones para describir un dominio, poniendo la cadencia en la compartición del conocimiento y el consenso en la representación de este. Por ejemplo, una ontología sobre una pizza podría incluir clases como la base de la pizza, tipos de pizza, ingredientes y relaciones como el nombre de una pizza, tipos de pizzas pertenecientes a un país, etc.

La idea es que la web semántica esté formada (al menos en parte) por una red de nodos tipificados e interconectados mediante clases y relaciones definidas por una ontología compartida por distintos tipos de pizzas. Por ejemplo, una vez establecida una ontología sobre pizzas, un recetario virtual puede organizar sus contenidos definiendo instancias de nombres de pizzas, ingredientes, etc., interrelacionándolas y publicándolas en la web semántica. Así varios recetarios virtuales podrían colaborar para dar lugar a una gran meta-receta que integre los contenidos de todos ellos. Un programa que navegue por una red como ésta puede reconocer las distintas unidades de información, obtener datos específicos o razonar sobre relaciones complejas. A partir de aquí sí podemos distinguir entre una receta elaborada por un aficionado y una receta elaborada por un chef.

Por último, la web no solamente proporciona acceso a contenidos, sino que también ofrece interacción y servicios (comprar un libro, reservar una plaza en un vuelo, hacer una transferencia bancaria, simular una hipoteca, etc.). Los servicios web semánticos son una línea importante de la web semántica, que propone describir no sólo información, sino definir ontologías de funcionalidad y procedimientos para describir servicios web: sus entradas y salidas, las condiciones necesarias para que se puedan ejecutar, los efectos que producen o los pasos a seguir cuando se trata de un servicio compuesto. Esas descripciones procesables por máquinas permitirán automatizar el descubrimiento, la composición y la ejecución de servicios así como la comunicación entre unos y otros.

Existen varias formas de abordar el problema de disponibilidad de contenido formal para el software. Una de las posibles soluciones consiste en procesar documentos ya existentes y adaptarlos a los formatos necesarios para que el software lo entienda. Esta línea tiene especial atención en la iniciativa de la Web Semántica, ya que permitirá reutilizar y hacer disponible del contenido actual de la WWW. En la parte del estado del arte del presente trabajo se enumeran algunas áreas de investigación, tecnologías y sistemas existentes que permitan procesar documentos online y extraer conocimiento de ellos para propósitos de procesamiento automático. Así mismo, se establece una relación entre las distintas clases de fuentes o documentos procesados desde el punto de vista estructural y la eficiencia de las tecnologías estudiadas.

Este trabajo pretende contribuir a la superación del reto de la presencia de una concentración crítica de contenido para la Web Semántica. Se propone una arquitectura para un sistema de extracción de conocimiento de dominios acotados que sea capaz de instanciar conocimiento en modelos semánticos. Esta arquitectura permite la utilización de diferentes tecnologías existentes sistematizadas mediante estrategias de extracción de acuerdo al tipo de fuente, tipo de dominio procesado y objetivos establecidos por el modelo. La arquitectura define un sistema de extracción y estructuración de información que produce como resultado el contenido apto para los propósitos de la Web Semántica, es decir, produce instancias de un modelo conceptual expresado en formato de ontologías[15]. Tras un estudio de sistemas existentes, se hace hincapié en la combinación y extensibilidad del sistema hacia distintas tecnologías. El sistema incorpora elementos de áreas de procesamiento de lenguaje natural, tratamiento de estructuras del documento, visualización y procesamiento del texto. Todas estas áreas ofrecen funcionalidades que en

[15] Término tomado de la filosofía donde se define como la rama de la metafísica que estudia la naturaleza de la existencia. En el marco de la Web Semántica se usa para modelos semánticos que modelan dominios con conceptos, atributos, valores y relaciones entre ello. En el apartado de preliminares se define de manera formal.

su combinación pueden contribuir en la labor de creación automática de conocimiento estructurado y procesable.

Como se explicara más adelante la propuesta, de la arquitectura de este sistema prevé una descripción de las fuentes a priori. Esta descripción permite elaborar una estrategia de extracción y estructuración del conocimiento de acuerdo a las características de la fuente. El reto de este tipo de sistemas es la rentabilidad de la construcción de descripciones de las fuentes. Deben ser fáciles de construir y deben de tender a ser descripciones reutilizables en diferentes fuentes de un mismo dominio.

En la sección del uso del sistema se presenta una aplicación de las tecnologías de la Web Semántica para la construcción de un portal en la búsqueda de información sobre toma de decisiones. El contenido del portal está generado por un sistema implementado de acuerdo a la arquitectura propuesta.

Por último, se presentan las conclusiones y las posibles líneas de trabajos futuros sobre la arquitectura y las implementaciones del sistema.

Objetivos

En la actualidad, el conocimiento es visto como un activo más, debe ser gestionado de la mejor forma posible con el fin de agregar valor a los productos y servicios prestados por un organismo.

General

Proponer una arquitectura para un sistema extracción de conocimiento de dominios acotados que sea capaz de instanciar conocimiento en modelos semánticos, que permitan la utilización de diferentes tecnologías existentes sistematizadas mediante estrategias de extracción de acuerdo al tipo de fuente, tipo de dominio procesado y objetivos establecidos por el modelo.

Específicos:

- El sistema incorporara elementos de área de procesamiento de lenguaje natural.
- Tratamiento de estructuras de documento.
- Visualización y Procesamiento del texto.

Todas estas áreas ofrecen funcionalidades que en su combinación pueden contribuir en la labor de creación automática de conocimiento estructurado y procesable.

HIPOTESIS

Debido a la rápida evolución de la web (desde la primera generación o web 1.0 pasando por la web 2.0 y llegando a la web 3.0 o web semántica) y el gran incremento de contenidos presentes en Internet como red global, cada vez se hace más necesario tener métodos eficientes de recuperación de información.

La recuperación de información consiste en encontrar el material (normalmente documentos) de entre grandes colecciones de datos para satisfacer la necesidad de un usuario. El principal objetivo del sistema de recuperación de información debe ser obtener los documentos más relevantes posibles en relación a una consulta particular.

Hoy en día es imprescindible el uso de motores de búsqueda para realizar las consultas en Internet y es posible que un mismo motor de búsqueda proporcione respuestas diferentes para diferentes versiones de una misma pregunta. Estos resultados pueden dependen de las palabras claves utilizadas y no siempre son correctos.

Estas son las principales motivaciones para el desarrollo de métodos de búsqueda semántica, aprovechar las propiedades de la semántica (como el estudio del significado de las palabras) para orientar la búsqueda y así intentar obtener resultados óptimos.

La dificultad de este tipo de búsqueda recae en que para los seres humanos es fácil establecer equivalencias semánticas entre diferentes expresiones pero este proceso no es evidente para los sistemas automatizados. Un sistema de búsqueda semántica ideal tendría que emular un hipotético sistema de búsqueda humano con una memoria suficientemente grande para recordar y relacionar todas las preguntas y respuestas anteriormente consultadas. Es cierto que diferentes personas pueden dar diferentes respuestas a una misma pregunta pero por mucho que reformulemos la consulta la respuesta será similar ya que semánticamente serán consultas equivalentes.

Finalmente el objetivo definitivo para un sistema artificial de búsqueda semántica será obtener los mismos resultados y en el mismo orden de relevancia respecto a diferentes consultas semánticamente equivalentes.

[Hildebrand[1] et al 2008] proporciona una visión general enumerando sistemas de búsqueda semántica e identifica otros usos de la semántica en los procesos de búsqueda.

[1] http://swuiwiki.webscience.org/index.php/Semantic_Search_Survey

Desambigüación

Típicamente el caso que suele presentarse es el de un usuario con una necesidad de información más o menos concreta que propone una consulta a un motor de búsqueda, esta consulta contiene palabras clave que el usuario considera necesarias o correctas para obtener la información deseada. Entonces el motor de búsqueda convierte en metadatos (crea una representación) las palabras clave utilizadas en la consulta y realiza la búsqueda en su base de datos. Esta contiene la relación de metadatos con todos los documentos que conoce y devuelve una lista de resultados en función de la relevancia establecida por el orden de clasificación. Este sistema tiene dos limitaciones principales: a veces el usuario no es capaz de definir correctamente su objetivo mediante palabras clave además de que los motores de búsqueda no entienden el lenguaje natural.

El lenguaje natural es muy complejo debido, en gran parte, al gran número de sinónimos y palabras polisémicas que contiene. En este punto entra en juego la importancia de la aplicación de sistemas de búsqueda semántica en los motores de búsqueda. En general el proceso de búsqueda semántica es:

A. Interpretar la pregunta del usuario extrayendo los conceptos más relevantes de la frase.
B. Utilizar este grupo de conceptos para crear una consulta y utilizarla contra la ontología del sistema.
C. Presentar los resultados al usuario.

Con tal de entender que es lo que el usuario está buscando (punto A del proceso), se debe desambiguar el significado de las palabras clave utilizadas en la pregunta. Se considera que un término es ambiguo cuando este puede tener un considerado número de significados posibles, por ejemplo la palabra hoja como "la hoja de un árbol", "una hoja de papel" o "una hoja de afeitar". Gracias a los procesos de desambigüación se elije el significado más probable de entre todos los posibles.

Estos procesos tienen en cuenta el significado del resto de palabras presentes en la frase y el resto del texto de las webs. La determinación de cada significado influye en la desambigüación de los demás hasta llegar a una situación de máxima verosimilitud y coherencia para la frase inicial consultada. Toda la información fundamental para el proceso de desambigüación, es decir, todo el conocimiento utilizado por el sistema, se ve representada en forma de una red semántica organizada alrededor de un núcleo conceptual.

Red semántica

El principal objetivo de la investigación de redes semánticas es el desarrollo de una serie de lenguajes y la tecnología necesaria para expresar información semántica que pueda ser entendida y procesada por las computadoras para poder aplicarlo al entorno del trabajo en red.

Una estructura de este tipo pretende representar el conocimiento lingüístico mostrando las interrelaciones entre conceptos. Cada concepto léxico coincide con el nodo de una red semántica y está conectado con otros por relaciones semánticas especificas en una estructura jerárquica y hereditaria. De esta forma, cada concepto enriquece con sus características y su significado a los nodos cercanos.

Cada nodo de la red agrupa un conjunto de sinónimos que representan el mismo concepto léxico y pueden contener:

- Lemas simples ('asiento', 'vacaciones', 'trabajo', 'rápido', 'más', etc.).
- Compuestos ('guardaespaldas', 'pararrayos', 'aguardiente', etc.).
- Colocaciones ('plan de choque', 'paquete bomba', 'llevar a cabo', 'bajo consumo', etc.).

Los enlaces que identifican las relaciones semánticas entre los conjuntos de sinónimos son las directrices a seguir para la organización de la red semántica de conceptos.

De acuerdo se establece la proposición de una arquitectura para un sistema de extracción de conocimiento de dominios acotados que sea capaz de instanciar conocimiento en modelos semánticos, permitiendo la utilización de diferentes tecnologías existentes sistematizadas, mediante estrategias de extracción de acuerdo al tipo de fuente, tipo de dominio procesado y objetivos establecidos por el modelo.

JUSTIFICACIÓN

La Búsqueda Semántica: Cómo Los Buscadores Semánticos Pueden Traer Beneficios Y Ventajas Al Usuario

¿De qué modo la búsqueda semántica puede ayudarte a buscar y encontrar lo que te interesa? ¿Qué es un buscador semántico y cuáles ventajas ofrece? ¿Cómo es que pueden clasificar los contenidos usando los mismos criterios de relaciones y gustos que usas tú?

¿De qué modo la búsqueda semántica puede ayudarte a buscar y encontrar lo que te interesa? ¿Qué es un buscador semántico y cuáles ventajas ofrece? ¿Cómo es que pueden clasificar los contenidos usando los mismos criterios de relaciones y gustos que usas tú?

Figura 1 Búsqueda Semántica

La búsqueda semántica no es un tipo de inteligencia artificial que toma decisiones en tu lugar, sino una nueva manera de gestionar las informaciones dentro del proceso mental humano, potencializada sin embargo por una capacidad creciente de las computadoras de gestionar datos.

Estructura De Las Informaciones

Figura 2 Estructura de Informaciones

Apenas he visto el video demo de Powerset, el buscador semántico revisado por Robin Good y me he preguntado en el mismo instante, como usuario y desarrollador de buscadores semánticos, ¿para qué sirve un buscador semántico?

Powerset[2] ha elegido aplicar su tecnología en Wikipedia, y porque la enciclopedia online ya es estructurada y bien organizada, no resulta difícil encontrar los temas relacionados a una área específica. Además de eso, la Wikipedia posee un estándar para la relación entre los contenidos. En realidad, uno puede obtener mucho más de ella si extiende los criterios de asociación, pero por ahora tales criterios son elegidos únicamente por los editores de Wikipedia.

¿Cuáles son entonces los beneficios que tendría Powerset? Si busco informaciones acerca del puente de Brooklyn, Powerset encuentra todo el material disponible sobre el tema. Se puede argumentar que el índice simple de Wikipedia nos provee las mismas informaciones.

Pero la cuestión es otra.

Objetivos de la Búsqueda

¿Qué quiero saber cuando hago una búsqueda? La pregunta no es tan banal como puede parecer: cuando hago una búsqueda, no busco el tipo de la respuesta. Permítanme explicarme mejor: si un niño quiere un caramelo, busca su mamá; si quiere un consejo de cómo defenderse en las peleas, busca su papá; si un ciudadano quiere un préstamo, va al banco.

Figura 3 Objetivo de la Búsqueda

El proceso de búsqueda se basa siempre en la información que ya se tenga. Si no la hay, se la busca. Por ejemplo, si quiero ir a Canadá de vacaciones y no sé cuál agencia elegir para el viaje, busco en Internet.

[2] http://www.powerset.com/

Considerar banal el uso del conocimiento en el mundo significa no entender nada acerca de las estructuras de aprendizaje, porque si quiero ir a algún lugar, lo de elegir a quién preguntar cómo se llega ahí y qué preguntar no resulta para nada estúpido, sino que es fundamental.

Google no provee las respuestas a las preguntas existenciales de tipo "quién soy", pero ayuda a encontrar quién tiene las respuestas: un cura, la iglesia más cercana, la religión que me interesa, un libro sobre el tema, un blog, un foro. En fin, todos los lugares posibles en donde buscar alguna respuesta. La elección del lugar indica la voluntad de obtener una determinada respuesta más que otra.

La respuesta a la pregunta anterior, o sea, por qué debo usar el Powerset cuando me da las mismas informaciones que Wikipedia, es que Powerset tiene la capacidad de estructurar las informaciones. Es cierto que Wikipedia organiza las informaciones, pero Powerset puede usar la misma tecnología en otros campos de aplicaciones y seguramente con resultados mejores.

Métodos de búsqueda tradicionales

Figura 4 Métodos de búsqueda tradicionales

Google nos da una mano pero no puede dar una respuesta directa, ayuda a buscar otras personas que pueden proveer las respuestas a nuestras cuestiones. El objetivo de los buscadores como Google, no es entender el mundo (al menos por ahora), pero ayudarnos a buscar las cosas que nos sean útiles: y eso no es poco.

¿Quiénes de ustedes, cuando quieren comprar un televisor, se fiarían en una máquina que razona según los criterios mecánicos y automáticos? Queremos ser nosotros mismos a decidir qué queremos, aunque si a veces no conocemos perfectamente el objeto de nuestra búsqueda, pero queremos ser consientes de lo que podemos buscar y obtener.

¿Por qué un niño no puede comer caramelos? Porque le harán mal a las muelas. Pero si el niño no entiende la respuesta, será recibida como una imposición externa, lo que crea una experiencia inmadura que no mejora su persona. Buscar en cambio significa mejorar, incluso en el alma.

Uso Futuro De Los Buscadores Semánticos

En mi opinión, la real utilidad de los buscadores semánticos será comprendida por la publicidad online y por el e-commerce. Cuando estés buscando algo o quieras profundizar un tema, no quieres encontrar cosas que no te interesen: este es el objetivo primordial de cada buscador semántico. Y el hecho de que un buscador semántico esté basado simplemente en un trabajo de clasificación (sea manual sea automático) no significa que esté estructurado de forma banal.

Figura 5 Fututo de los buscadores semánticos

El buscador semántico además no es un tipo de inteligencia artificial, tiene tan sólo la tarea de ordenar las ideas y los contenidos. Cada uno de nosotros se divierte cuando tiene que poner las cosas en orden: ser ayudado está bueno, no significa ser sustituido, porque eso no nos enriquece.

Imaginemos llegar a casa y decir a nuestro televisor: "Enséñame lo que sabes que más me gustará." Seguramente, proseguiré a averiguar todo lo que hay en los varios canales y, como no tengo una personalidad clasificativa a perfección, la tendencia es que elija algo que realmente me guste.

Ahora imagina llegar a casa y pedir al televisor: "Dame una película romántica que no sea demasiado dulzona", en este caso, mediante la propuesta, la tendencia es que cualquier cosa me complaciera. El buscador podría buscar en mi disco duro y mostrarme cualquier cosa con base en el gusto de la gente aficionada por películas románticas. O también proponerme una presentación de fotos de los paisajes de mis vacaciones, con un fondo musical.

Esto es el resultado del uso de la semántica como principio de ordenación entre los miles de cosas que tenemos. No creo que un buscador semántico deba hacer una elección por mí, más que eso me encantaría poder insertar en un buscador semántico todo lo que leo, amo, encuentro y etiqueto con mis ideas: lo ideal sería poder recoger todos estos contenidos y organizar criterios de relación. De esta forma, cuando buscas algo, no sólo puedes encontrar los documentos, sino que también todas las ideas que hayas tenido, incluso las ideas asociadas a la idea inicial.

De acuerdo a lo anterior, si la semántica tiene que ver con la significación de palabras, en el caso de la Web debe estar dada principalmente por la creación de documentos digitales (Objetos digitales) pero con significado, de tal forma que ya no se hable de información sino de conocimiento.

Pero la revolución, como se sabe, no está solamente en la tecnología, pero más que todo en el modo en que usamos la tecnología. Esto puede hacer la diferencia.

UNA INTRODUCCIÓN A LA METODOLOGÍA DE SISTEMAS

INTRODUCCION

La metodología de sistemas es uno de los conceptos que, con el enfoque de sistemas y la intertransdisciplina, forman los tres conjuntos que interactúan formando un sistema que integra los conceptos básicos fundamentales para el desarrollo del estudio y aplicación de sistemas. En el presente trabajo se propone la arquitectura de un sistema de extracción de conocimiento de dominios acotados que sea capaz de instanciar conocimiento en modelos semánticos, la evolución histórica de la metodología de sistemas a través de las principales corrientes que se detectan. Se analizan sus características, así como sus tendencias de divergencia o convergencia y de síntesis que se presentan.

El énfasis metodológico se detecta desde los inicios del movimiento sistémico, pero quizás por falta de claridad de los conceptos y la supuesta mayor facilidad de comprensión y aplicación, las actividades académicas y profesionales enfocadas al desarrollo, aplicación y difusión de sistemas han dado diferentes énfasis, no sólo a alguno(s) de los tres conceptos básicos. En lo relativo a metodología, también se han dado diferentes énfasis a dos aspectos fundamentales, que por su propia naturaleza, deben considerarse inseparables y que ante la actividad específica de que se trate hay que cuestionar y buscar el balance más adecuado o el más apropiado, que debe darse a cada componente sin soslayar ninguno. Estos dos aspectos a que nos referimos son:

- El método
- Las técnicas y herramientas.

Esta falta de balance, en parte, ha propiciado un importante desarrollo de las técnicas y herramientas de sistemas, que si bien en su mayoría han contribuido a resolver problemas, también han contribuido a cometer el error de resolver el problema equivocado y generar más problemas, lo cual ha despertado la crítica y el señalamiento de las limitaciones de los conceptos sistémicos.

Esa falta de balance, la corrección de esos errores y el aprendizaje para evitarlos, sólo se logrará con el manejo apropiado de los tres conceptos básicos fundamentales de sistemas, en especial del conocimiento metodológico adecuando para su aplicación. Por eso, en este trabajo se comienza el conocimiento de la metodología por el camino de su evolución histórica a través de algunos de los principales autores cuyas obras tienen mayor relación con la metodología de sistemas.

METODOLOGÍA, MÉTODOS, TÉCNICAS Y HERRAMIENTAS.

El concepto básico fundamental de la metodología en sistemas es el relativo a la consideración del conocimiento, desarrollo, la aplicación, el estudio del método o métodos. La metodología se considera como parte de la filosofía, de la epistemología, de la filosofía de la ciencia y de la ciencia, que promueve la adopción de una actitud, el desarrollo de aptitud y un modo de proceder de indagación permanente, para utilizar y/o construir caminos, o sean métodos para contestar preguntas y resolver problemas.

A la Metodología también se integran las consideraciones aportadas por el diseño, considerando como los procesos de búsqueda creativa que genera tanto nuevos modos de percibir la realidad, como nuevos métodos para contestar preguntas y resolver problemas; generar nuevos conceptos, artefactos, objetos tangibles o intangibles; métodos que conscientemente promueven también el cambio de uno mismo y de nuestro contexto. El método, etimológicamente significa la vía, el camino (o dos) que guía más allá, más lejos (meta). Históricamente, la metodología desde los comienzos del movimiento sistémico, toma sus bases de la ciencia. Se reconoce que si bien la ciencia ha permitido alcanzar logros significativos en la generación de conocimientos y en la resolución de problemas, es necesario caracterizarla sistémicamente para mejorar sus aportaciones y vincularla con la filosofía y otras actividades.

Desde el punto de vista de la ciencia, el método, es el que le da su característica primordial, al definir el camino como el proceso controlado de indagar para alcanzar eficiente y eficazmente los objetivos deseados. Al definir el camino, el método

proporciona las maneras de seleccionar y usar las técnicas y herramientas. Por esto, para comenzar a aclarar los términos, las herramientas serán los instrumentos utilizados en el indagar científico y las técnicas serán la manera de usar esos instrumentos para lograr un objetivo. Se puede decir, que el método nos da las pautas para alcanzar eficazmente los objetivos deseados y que las técnicas y las herramientas coadyuvan a su logro, de manera eficiente. La eficacia y la eficiencia deben tener el balance apropiado. La metodología nos permite obtener ese balance entre el método, las técnicas y las herramientas.

Sin embargo, en buen número de casos, las actividades académicas y profesionales relacionadas con sistemas, han dado mayor impulso a la aplicación, desarrollo y difusión, tanto a la construcción de modelos matemáticos, como al manejo de las técnicas y herramientas de sistemas, soslayando al método como un todo. Resulta entonces primordial impulsar el conocimiento, el desarrollo, la aplicación, el estudio del método: de la metodología. Sólo así se estará en posibilidades de buscar y encontrar el balance apropiado antes mencionado, y coadyuvar más eficiente y eficazmente a la resolución de los problemas cuya solución tanto apremia.

La metodología tiene como fin el mejoramiento permanente de los procedimientos y criterios usados en la conducción de la indagación requerida para contestar preguntas y/o resolver problemas.

PRIMER CAMINO HISTÓRICO DE LA METODOLOGÍA

La metodología de sistemas al estudiar lo que ha acontecido en el pasado, descubre que no puede hablarse de la existencia de un solo método de sistemas. Conforme el tiempo ha transcurrido desde fines de los años 30 y principios de los 40, en que se puede decir se comenzaron a gestar los conceptos de sistemas, se han mencionado y definido varios métodos.

L. von Bertalanffy, considerado como uno de los fundadores de la Teoría General de Sistemas y uno de los precursores del movimiento de sistemas, es quien al partir de la necesidad de ver al organismo viviente como un sistema organizado, se dio a la búsqueda

de las leyes que regían su comportamiento, concibe así la idea de la Teoría General de Sistemas como una doctrina interdisciplinaria que elabora principios y modelos aplicables a sistemas en general y que determina las correspondencias o isomorfismos existentes entre sistemas de diferente naturaleza.

Estos esfuerzos abrieron la posibilidad de la unificación de las ciencias y más adelante llevaron a promover la investigación de sistemas generales, así como la ciencia y la filosofía de sistemas. Pero desde los comienzos de la Teoría General de Sistemas, buena parte de sus proponentes y promotores enfatizaron más las tareas de la construcción de modelos, en especial los matemáticos.

El enfoque matemático seguido en la Teoría General de Sistemas, se considera no como el único posible o el más general, sino que se complementa con enfoques modernos como teoría de la información, cibernética, teoría de juegos, decisiones, modelos estocásticos, investigación de operaciones, por mencionar algunos. Sin embargo el hecho de que las ecuaciones diferenciales cubren campos extensos en la física, la biología, la economía y las ciencias del comportamiento, las hace un acceso apropiado para el estudio de sistemas generalizados.

Bertalanffy delinea así un cambio de paradigma que si bien se apoya en la ciencia y sus métodos, define un nuevo modo de ver, sistémicamente, la ciencia y la realidad. Su consideración metodológica, en esas épocas eran más bien difusas. Explícita dos caminos posibles que llama métodos generales, los cuales pueden seguirse por separado o combinadamente:

1. *Método empírico-intuitivo*: que se mantiene cerca de la realidad y que fácilmente puede mostrarse y verificarse con ejemplos de campos individuales de la ciencia, pero al cual le falta la elegancia matemática y la fuerza deductiva, pareciendo ingenuo y no sistemático.
2. *Método deductivo de teoría sistémica*: que permite la formalización matemática de los conceptos, relaciones y transformaciones envueltas en un sistema.

Entre una de las primeras actividades que empezaron también a desarrollar y aplicar conceptos de sistemas, se encuentra la actividad que se le dio el nombre de Investigación de Operaciones. En esos comienzos, la definición de un método seguía siendo un asunto más bien difuso. En 1951 al publicar uno de los primeros libros de Investigación de Operaciones, P. M. Morse y G. E. Kimball enfatizan que para atacar problemas y encontrar soluciones definitivas hay que usar el procedimiento que consiste en:

- Estudiar las operaciones pasadas para determinar los hechos.
- Construir teorías para explicar los hechos.

Usar los hechos y las teorías para predecir las operaciones futuras. Este procedimiento podría considerarse el método de sistemas definido por esos autores, siendo obvia su relación con el método científico. Sin embargo, Morse y Kimball mencionan al mismo tiempo la utilización de otros métodos y herramientas, como métodos estadísticos, la experimentación y métodos analíticos (teóricos).

No cabe duda que a pesar de que en esos años no existía una definición precisa del método, los grupos científicos y técnicos que utilizaron esos conceptos tuvieron éxito al colaborar significativamente en la victoria de los países aliados en la segunda guerra mundial. La investigación de Operaciones también tuvo éxito al aplicarse en la industria productora de bienes y servicios tanto privada como pública.

En 1954 y 1956 J.G. Mc Closkey y F .N. Trefethen editaron dos volúmenes en que presentaron una amplia variedad de problemas resueltos aplicando modelos, técnicas y herramientas de la investigación de operaciones en la administración de sistemas. La consideración metodológica estuvo ausente.

En 1957 se publicó la Introducción a la Investigación de Operaciones de C.W. Churchman y R.l. Ackoff que contiene uno de los primeros esfuerzos sistemáticos más relevantes sobre la metodología de sistemas. Churchman y Acko con su formación en filosofía de la ciencia, su posición filosófica pragmática y sus experiencias prácticas, reconocen la necesidad de definir explicita, sistemática y sistémicamente el método de la Investigación de

Operaciones. Con sus bases filosóficas y concepto de ciencia, aprendidos de su maestro el filósofo Edward A. Singer Jr., se dieron a esa tarea.

Su mismo concepto de ciencia lo enfocan sistémicamente, estableciendo la necesidad de mantener la interacción de esa actividad con las otras actividades filosóficas del hombre. Su concepto de ciencia y de su método, son conceptos amplios y plenos de posibilidades. De estos conceptos derivan el método de la Investigación de Operaciones, considerando sus fases como componentes de un sistema para indagar y enfrentar los problemas de objetos de estudio ya existentes a diseñar. Consideran esos mismos objetos como sistemas en los que interactúan hombres y máquinas. También plantean la necesidad de que ese proceso de indagación, se lleve a cabo por grupos interdisciplinarios.

Las fases del método de la Investigación de Operaciones establecidas como interactuantes por Churchman y Ackoff son:

- Formulación del problema.
- Construcción de un modelo.
- Obtención de una solución.
- Prueba del modelo y la solución.
- Implantación y control de la solución.

En la fase de la construcción de un modelo Churchman y Ackoff explicitan la posibilidad de la utilización de diferentes tipos de modelos, no sólo los matemáticos. En el método que proponen, en sus fases interactuantes, se encuentra una consideración balanceada entre método, técnicas y herramientas en la búsqueda de una solución del problema considerando todo como sistema.

En 1957 H.H. Goode y R.E. Machol publicaron el primer libro sobre Ingeniería de Sistemas, otra actividad que, entre las primeras, también empezaron a desarrollar y aplicar conceptos de sistemas, pero aún cuando presentaban en él aspectos metodológicos le dieron mayor énfasis a la presentación de los modelos matemáticos, las técnicas y herramientas de sistemas. Su método, más orientado a la creación de sistemas tecnológicos de ingeniería lo describen compuesto de las fases:

- Organización
- Diseño Preliminar
- Diseño Principal
- Construcción de Prototipo
- Prueba, Entrenamiento y Evaluación

Se puede considerar que fue A.D. Hall quien más adelante, sistematizó y extendió de manera más comprensiva los aspectos del método para la Ingeniería de Sistemas. A. Kaufmann en 1959 comenzó a publicar su serie de libros sobre métodos y modelos de la investigación de operaciones. Enfatiza los conceptos de modelos, técnicas, y herramientas matemáticas y computacionales, considerándolos a todos ellos como métodos.

Ch, D. Flagle, W.H, Hugging y R.H. Roy en 1960 publicaron su obra en que integran la Investigación de Operaciones y la Ingeniería de Sistemas, pero dicha integración únicamente se explicita en la coincidencia del uso de modelos matemáticos, técnicas y herramientas en sistemas, confundiendo también todos estos conceptos con metodologías y sin enfatizar un concepto integrador de metodología de sistemas. En su libro sobre método científico de 1962, Ackoff amplía la relevancia de su aportación metodológica; en él explícito, integral mente y por separado, cada una de esas fases! del método de la investigación de operaciones publicado antes con Churchman. Ackoff! explicita de manera extensa, adapta y actualiza su obra metodológica sobre el diseño de la investigación social de 1953.

En esta nueva obra Ackoff caracteriza con amplitud, la ciencia sistémica como el proceso de indagación para contestar preguntas, resolver problemas y desarrollar procedimientos más efectivos para hacerlo. Reconoce la existencia de varios modos de indagar y los relaciona al método científico, como proceso controlado, dirigido para alcanzar los fines deseados. Construye así un sistema en que interrelaciona los aspectos sociales, tecnológicos, herramientas, técnicas, método científico, métodos y metodología. Coloca también la ciencia en relación con otras actividades que considera indispensables para el desarrollo integral del hombre en su búsqueda por los ideales: Científico, Político-Económico, ético-Moral y Estético.

En 1962 A. D. Hall hace otro de los primeros esfuerzos relevantes sobre metodología de sistemas. Con su formación de ingeniería y experiencias prácticas, también reconoce la necesidad de la definición del método y de proporcionar, lo que denomina las bases filosóficas de la Ingeniería de Sistemas. En base a su experiencia Hall integra los conceptos de ciencia, tecnología y creatividad en su definición de las fases de su metodología de la Ingeniería de Sistemas, señalando la existencia de similitudes en las fases del método de la Investigación de Operaciones. Esas similitudes las explica en base a que ambas actividades son derivadas del método científico moderno. Aún cuando Hall señala que estas actividades difieren de los fines que persiguen, ya que considera que la investigación de Operaciones, generalmente se preocupa de las operaciones de un sistema ya existente y que la Ingeniería de Sistemas de la creación, desarrollo y puesta en operación de nuevos sistemas; las similitudes entre ambas actividades son mayores y más importantes que las diferencias, ya que, no sólo en cuanto a método hay similitudes, sino a otras características como interdisciplinariedad, técnicas y herramientas usadas así como el considerar sus objetos de estudio como sistemas en que interactúan los aspectos sociales y tecnológicos.

Las fases del método de la Ingeniería de Sistemas establecidos por Hall son:

- Estudio de Sistemas (planeación de programa).
- Planeación exploratoria (planeación de proyecto I).
- Definición del problema.
- Selección de objetivos.
- Síntesis de sistemas.
- Análisis de sistemas.
- Selección la mejor alternativa.
- Comunicación de resultados.
- Planeación de desarrollo (planeación de proyecto II).
- Estudios durante el desarrollo (fase de acción).
- Ingeniería (fase de acción II).

Componente importante de cada una de estas fases, como se explícita en la planeación exploratoria, es la adopción de un procedimiento para la resolución de problemas y Hall toma como base la posición filosófica pragmática de John Dewey y la adapta explicitándola como un sistema en que interactúan:

- La definición del problema
- El análisis y la síntesis
- La toma de decisiones.
- La planeación de la acción.

En 1965 W. Boumol publicó su obra remarcando primordialmente la relación entre la Investigación de Operaciones, el Análisis de Operaciones y la Teoría Económica, a través sólo de los modelos matemáticos, técnicas y herramientas en el estudio de esos fenómenos y en la toma de decisiones para resolver problemas sin hacer algún énfasis metodológico.

En 1965 H. Chestnut publica su libro sobre herramientas de la Ingeniería de Sistemas y en 1967, el mismo Chestnut y Van Court Hare publicaron sus libros en los que también, a pesar de uno de sus títulos, el énfasis fue mayor en los modelos matemáticos, en las técnicas y en las herramientas mostrándose además, la prioridad que se pone en esas últimas al publicarse primero el libro sobre herramientas. Chestnut en su método de Ingeniería de Sistemas da también más énfasis a la creación, ingeniar un sistema y operarlo; establece los elementos genéricos que orientan el proceso. Van Court Hare expone implícitamente su método para el Análisis de Sistemas compuesto de:

- Definición del Sistema
- Análisis y Diagnóstico
- Tratamiento del Sistema (Implantación y Mejoramiento)

S. Beer en 1966 hace una aportación significativa al formalizar la interacción entre la Investigación de Operaciones, la Cibernética y la Administración a través de los conceptos de sistemas, complejidad, modelos, decisión, comunicación y control de manera integral, ampliando y consolidando las bases para el modelo general que en años subsiguientes construiría y propondría como el modelo de sistema viable.

En 1968, Ackoff junto con M. Sasieni publican una versión actualizada a la obra en que participó el primero en 1957. Ante la amplia difusión de publicaciones que si bien muestran la solución de gran variedad de problemas, en su mayoría se orientan al análisis de modelos matemáticos, técnicas y herramientas, en esta nueva obra Ackoff insiste en promover el conocimiento metodológico para tener el balance apropiado. También en esta obra Ackoff comienza a explorar las fronteras de la Investigación de Operaciones, proponiéndose ampliarlas para resolver no sólo problemas tácticos, sino poder enfrentar también problemas estratégicos, es decir, enfrentar no sólo problemas en sistemas, sino también poder enfrentar sistemas de problemas, a través de la planeación.

En 1969 G.M. Jenkins presentó otro de los esfuerzos significativos en la definición del método de la Ingeniería de Sistemas. Su trabajo, aún cuando corto en extensión presenta de manera destacada su definición del método. Las fases del método de Jenkins son:

- ***Análisis de Sistemas***: Formulación del problema. Organización del proyecto. definición del sistema, Definición del sistema más amplio. Objetivo del sistema más amplio, Objetivos del sistema, Definición del criterio económico global, Recolección de información de datos.
- ***Diseño de sistemas (síntesis):*** Predicción. Construcción de modelo y simulación. Optimización, Control, Confiabilidad.
- ***Implantación:*** Documentación y aprobación, construcción.
- ***Operación:*** Operación inicial. Vista retrospectiva, Mejoramiento de operación.

En 1971 R. de Neufville y J.H. Stafford enfatizando primordialmente la modelación matemática; así como las técnicas y herramientas, abordan lo que denominan el análisis de sistemas para la resolución de problemas y la toma de decisiones en los sistemas relacionados con la ingeniería y la administración. Neufville y Stafford consideran el análisis de sistemas como un conjunto coordinado de procedimientos para diseñar y administrar, congruente con el método científico, en el que las hipótesis del comportamiento de la realidad son los modelos que formulan una teoría. El método que proponen lo denominan un procedimiento básico analítico compuesto de los cinco pasos:

- Definición de Objetivos
- Formulación de Medidas de Efectividad
- Generación de Alternativas
- Evaluación de Alternativas
- Selección

Los métodos de sistemas antes descritos, a pesar de contener rasgos que los hacen similares, representan una muestra de la variedad de métodos que pueden definirse. Ante ésta situación Churchman publicó en 1971 su libro Diseño de Sistemas de Indagación, retornando y actualizando su trabajo con Ackoff de 1950 en que presentan su análisis de la filosofía y la naturaleza del método científico. Churchman insiste en la importancia de la metodología de sistemas y de su relación con los procesos de indagar, vistos éstos como sistemas, y de éstos con la filosofía. Se reconoce que la filosofía, además de definirse de manera general, como el amor por el conocimiento, por la sabiduría. su definición involucra primordialmente un modo de ver el Mundo, de ver la realidad, como sistemas en que se relacionan los aspectos naturales, sociales y tecnológicos y como interactuar con ellos para obtener conocimiento, contestar preguntas y/o para transformarlos, resolver problemas.

Churchman reconoce que metodológicamente han existido diferentes posiciones o corrientes filosóficas y presenta las cinco que considera primordiales como sistemas filosóficos. Para cada una de ellas describe su modo de interactuar con la realidad, es decir, lo que cada una de ellas toma como punto de partida, como insumos de los que parte para iniciar un proceso de indagación y el cómo cada una llega a contestar preguntas o resolver problemas, presenta entonces, las descripciones de sus procesos de indagación.

Las cinco corrientes que presenta Churchman son: el racionalismo, el empirismo, el criticismo, la dialéctica y el pragmatismo experimental de Singer. Churchman considera que esa última corriente filosófica sintetiza las cuatro primeras, resultando por tanto, la posición más comprensiva y plena de posibilidades, pero no niega la posibilidad de que existan situaciones en que alguna de las otras corrientes sea la más apropiada para

aplicarse. Ante una situación determinada, en que se necesita o desea contestar una pregunta o resolver un problema, el conocimiento metodológico ayudará a seleccionar que sistema filosófico es el más apropiado a usar. Ese conocimiento metodológico y la actitud inquisitiva permanente ayudarán a contestar las preguntas y resolver los problemas de manera más eficaz.

SISTEMAS DUROS Y SUAVES

A pesar de la existencia de los esfuerzos metodológicos antes mencionados, como se ha dicho, las actividades profesionales y académicas de sistemas, en la mayoría de los casos, han mantenido su énfasis en los modelos matemáticos, las técnicas y las herramientas destacando en su desarrollo y aplicación las relativas a la optimización, la probabilidad, la estadística y la computación. Con esto se ha producido la apariencia de que sistemas es equivalente o parte de las matemáticas o de las matemáticas aplicadas.

Sin el suficiente énfasis metodológico también se ha producido la apariencia de que las actividades de sistemas se basan en conceptos de ciencia convencional: reduccionista, analítica y mecanicista. Sin embargo, las actividades de sistemas se basan en una nueva ciencia sistémica: expansionista, sintética y teleológica. Con una base de ciencia convencional, no sorprende que se haya considerado que sistemas se ve limitado para afrontar problemas en determinado tipo de contextos, ya que al observar la contribución que se había logrado en la resolución de problemas de sistemas, en organizaciones productoras de bienes y servicios, las actividades de sistemas han incursionado a tratar de coadyuvar en la resolución de problemas sociales, encontrándose con dificultades que desencadenaron la crítica de las posibilidades de su aplicación.

En la búsqueda metodológica de encontrar las razones de las limitaciones de la aplicabilidad de sistemas, para superarlas, se ha identificado que los objetos de estudio, pueden clasificarse como sistemas duros y suaves. Los sistemas duros se identifican como aquellos en que interactúan hombres y máquinas. En los que se les da mayor importancia a la parte tecnológica en contraste con la parte social. La componente social de estos

sistemas se considera como si la actuación o comportamiento del individuo o del grupo social sólo fuera generador de estadísticas. Es decir, el comportamiento humano se considera tomando sólo su descripción estadística y no su explicación. En los sistemas duros se cree y actúa como si los problemas consistieran sólo en escoger el mejor medio, el óptimo, para reducir la diferencia entre un estado que se desea alcanzar y el estado actual de la situación. Esta diferencia define la necesidad a satisfacer el objetivo, eliminándola o reduciéndola, Se cree que ese fin es claro y fácilmente definible y que los problemas tienen una estructura fácilmente identificable.

Los sistemas suaves se identifican como aquellos en que se les da mayor importancia a la parte social. La componente social de estos sistemas se considera la primordial. El comportamiento del individuo o del grupo social se toma como un sistema teleológico, con fines, con voluntad, un sistema pleno de propósitos, capaz de desplegar comportamientos, actitudes y aptitudes múltiples. Al comportamiento no sólo hay que describirlo si no hay que explicarlo para conocerlo y darle su propia dimensión. Un sistema suave es un sistema con propósitos, que no sólo es capaz de escoger medios para alcanzar determinados fines, sino que también es capaz de seleccionar y cambiar sus fines. En estos sistemas se dificulta la determinación clara y precisa de los fines en contraste a los sistemas duros. Los problemas en los sistemas suaves no tienen estructura fácilmente identificable.

Resulta obvio que con el énfasis en los modelos matemáticos, las técnicas y las herramientas de sistemas sin o con muy poca consideración metodológica, la mayoría de los problemas de los sistemas duros se pueden atacar y resolver pero ¿se estará actuando correctamente desde el punto de vista ético, al considerar al hombre y al grupo social sólo como máquinas generadoras de datos estadísticos y no darles su propia dimensión?

Algunos autores consideran que varios de los métodos de sistemas, como los que hasta ahora se han presentado, son los que se han usado para los sistemas duros y han planteado la necesidad de desarrollar métodos apropiados para los sistemas suaves. Sin embargo, aquí se ha argumentado y la revisión de la literatura también lo demuestra, que la utilización de modelos matemáticos, las técnicas y de las herramientas de sistemas han

sido más preponderante. De ese énfasis se derivan los éxitos de las aplicaciones en los sistemas duros y sus dificultades en los sistemas suaves por su limitada consideración metodológica. Los éxitos observados podrán acrecentarse obteniendo soluciones más eficaces con el énfasis apropiado de la metodología.

Quizás pudiese ser cierto que alguno de los métodos hasta ahora vistos, tuvieran limitaciones para aplicarse a los sistemas suaves, pero las bases, posiciones y proposiciones metodológicas planteadas por algunos autores como Churchman y Ackoff, de mantener una actitud de indagación amplia y permanente, permite el progreso que coadyuva a resolver los problemas en sistemas duros y suaves más eficaz y eficiente. Manteniendo esa actitud, como ya se mencionó, Ackoff comenzó en 1968 a cuestionar la existencia sólo de problemas en la realidad. En la realidad lo que existe, a lo que nos enfrentamos, es a situaciones confusas, difusas, inciertas, obscuras, no estructuradas, que nos producen inquietud o perturban; nos enfrentamos a un "embrollo". Una de esas situaciones podríamos percibirla y estructurarla como un problema en un sistema; sin embargo, considerando la visión expansionista, a lo que nos enfrentamos es a un sistema de problemas, a una "problemática", a una situación no estructurada.

Para enfrentarnos a esa problemática, el conocimiento y aplicación del método para resolver problemas en sistemas pueden ser necesario, pero no suficiente. Se presenta sí la necesidad de contar con un método para enfrentarnos a sistemas de problemas. Así, desde 1968 Ackoff comenzó a presentar las características y método de planeación para satisfacer esa necesidad encontrada y contribuir más al desarrollo metodológico de sistemas. Churchman impulsando también esa actitud de indagación amplia y permanente, en 1979 publica su libro sobre el enfoque de sistemas y sus enemigos en el que, con su base filosófica y metodológica, explica algunos caminos para capturar la riqueza del enfoque de sistemas. Para entender una idea "rica" que involucra muchas connotaciones, para atrapar su significado, hay que seguir muchos caminos.

Churchman en esa obra explora los caminos de su historia o tradición, su estructura lógica, su ética o teoría del valor su potencial, sus enemigos y su futuro. Sobre todo, enfoca su exploración a la crítica interna y externa, al enfoque para identificar enemigos y aprender

de ellos. Solo con una posición así, es que el progreso y desarrollo puede lograrse. Hasta aquí hemos seguido un camino en el análisis de la historia de la evolución de la metodología de sistemas hasta llegar al planteamiento de la necesidad de desarrollar más métodos para problemas en sistemas suaves. Analicemos otro camino que evolucionó en forma paralela al anterior y abrió nuevas perspectivas.

SEGUNDO CAMINO HISTÓRICO DE LA METODOLOGÍA

Desde los años 40 un grupo de sociólogos del Instituto Tavistock de Londres de Relaciones Humanas, ante la necesidad y con los deseos de colaborar en la reestructuración y mejoramiento de la producción industrial, devastada por la segunda guerra mundial, inspirados y animados por el psicólogo social Kurt Lewin, empeñaron sus esfuerzos en hacerlo, interviniendo en las organizaciones, aplicando el proceso de investigación-acción desarrollado por Lewin, en base a su teoría del campo para explicar el fenómeno psico-social del cambio. El método de investigación-acción fue desarrollado y aplicado para estudiar fenómenos sociales. El Institute Tavistock lo adoptó como el camino para enfrentar problemas en organizaciones vistas como sistemas. La investigación-acción combina el esfuerzo de la generación de teoría del fenómeno y el esfuerzo de producir cambios en los sistemas sociales a través del proceso de actuar, interactuar en el sistema.
La investigación-acción toma como principios los proverbios:

- Si quieres conocer algo trata de cambiarlo.
- No hay algo tan práctico como una buena teoría.

La investigación-acción reconoce que un aspecto fundamental en el éxito de la intervención en un sistema, depende de la relación que se establezca entre quien desea ayudar a resolver el problema, el investigador como agente de cambio y el grupo social del sistema, el cliente. La investigación-acción pone especial cuidado en esa relación para no producir situaciones de dependencia del cliente respecto al investigador, sino más bien producir un incremento en las capacidades del sistema social para aprender a resolver los problemas, independientemente del agente de cambio.

Las fases de la investigación-acción son:

- Ganar acceso al sistema.
- Identificar los problemas con los miembros del sistema.
- Recolección de datos y diagnóstico preliminar.
- Retroalimentación del diagnóstico preliminar a los miembros.
- Diagnóstico conjunto del problema.
- Tomar acción acordada por los miembros.
- Evaluar resultados.

A través de estas fases de manera cíclica, se diagnostica identificando o definiendo un problema, se planea la acción considerando cursos de acción alternativos para resolverlo, se toma la acción seleccionando un curso de acción e implantándolo, se evalúa estudiando las consecuencias de la acción y se especifica el aprendizaje obtenido identificando los hallazgos principales. El método de la investigación-acción marca el camino para buscar y aplicar la dinámica grupal más apropiada que lleve a la resolución del problema y al mejoramiento de las capacidades del sistema social, para repetir el proceso permanentemente, cuando sea necesario.

Los investigadores del Instituto Tavistock a través de una extensa aplicación de la investigación-acción desarrollaron conceptos de sistemas que fueron una de las bases para el inicio de la encrucijada, el cruce, de los dos caminos históricamente casi paralelos. Esta encrucijada y sus perspectivas serán descritas más adelante. Los métodos de sistemas presentados en secciones anteriores, además de que la mayoría tienen sus bases filosóficas y científicas bien cimentadas, son también resultado de extensas experiencias de invención con problemas en sistemas, por lo que se podría decir que se basan en investigación orientada a la acción en esos sistemas; sin embargo, por las características propias de esas intervenciones y las del método de investigación-acción, no se puede decir que esos métodos estén relacionados estrechamente.

Desde 1972 P.B. Checkland comenzó a señalar la necesidad de desarrollar métodos apropiados para los sistemas suaves, y empeñar su esfuerzo en definir uno explícito que para ello se basó en la investigación acción entre otros conceptos. Los resultados de su esfuerzo los sintetiza y concretiza en su libro de 1981 en el que describe su metodología de sistemas suaves, cuyas fases son:

1. Partir de una situación no estructurada con fronteras inciertas.
2. Analizar la situación para comenzar a estructurarla sin comprometerse en soluciones.
3. Seleccionar el Sistema relevante y elaborar su "definición raíz", básica.
4. Construir modelos conceptuales del sistema relevante que satisfaga la "definición raíz", modelo de lo que debería ser, en términos sistémicos.
5. Comparar el producto de 4 con 2 como elementos para debatir posibles cambios con los actores.
6. Definir los cambios acordados por loa actores como deseables y factibles, a través de un debate.
7. Implantar la acción acordada para mejorar la situación.

Checkland considera que de estas fases algunas se llevan a cabo en el mundo real (1, 2, 5 y 6) y otras en el mundo del pensamiento sistémico (3 y 4). En la fase 4 se utilizan conceptos sistémicos formales y se consideran otras formas de pensar sistémicas. Checkland trata con su metodología de establecer la diferencia e interrelación entre el pensamiento sistémico, la realidad y la práctica.

Reconociendo la importancia de la relación entre filosofía y método, Checkland elaboró todas las bases necesarias para su método. Determinó que su metodología satisfacía las características que Churchman atribuye a los aspectos de indagación del pragmatismo experimental. Definió también, que su método se relaciona con los trabajos sobre sistemas apreciativos, con los que Sir Geoffrey Vickers desarrolló su teoría para describir y explicar los procesos que caracterizan los sistemas sociales, inconforme en considerar al individuo y los grupos sociales como simples entes que buscan sólo alcanzar metas,

actuando como máquinas. Los trabajos de Vickers han sido fundamentales para impulsar la consideración del hombre y el grupo social, como sistemas plenos de propósitos, para considerar toda la riqueza humana en sistemas.

Las relevantes aportaciones de Vickers al conocimiento de los sistemas suaves cobran mayor dimensión con el tiempo y se integraron a su obra en que insiste que los sistemas humanos son diferentes, publicada después de su muerte.

EL CRUCE DE CAMINOS METODOLÓGICOS

El cruce de los dos caminos históricos analizados, se inicia al reconocer que resulta difícil clasificar un sistema en duro o suave. Ante esta dificultad, se insiste en la interrelación entre los aspectos sociales y tecnológicos; se impulsa el percibir, el identificar y el estructurar los sistemas como sistemas socio-técnicos, buscando el balance apropiado tanto de los aspectos sociales, como de los tecnológicos.; se impulsa el percibir, el identificar y el estructurar los sistemas como sistemas socio-técnicos, buscando el balance apropiado tanto de los aspectos sociales, como de los tecnológicos.

Desarrollaron de acuerdo a un programa que cubre; el desarrollo de conceptos, los métodos para el estudio analítico de las relaciones tecnología y formas organizacionales en diferentes contextos, la búsqueda de criterios para obtener el mejor acoplamiento entre los componentes social y tecnológico, la investigación-acción para mejorar ese acoplamiento y los modos de medir y evaluar resultados a través de estudios comparativos y longitudinales. Trist y Emery empeñaron sus esfuerzos tanto a nivel micro de unidades productoras de bienes o servicios simples, hasta el nivel macro de sistemas en comunidades y sectores industriales e instituciones que operan a nivel social amplio. Trist y Emery, al igual que Churchman y Ackoff, empeñan sus esfuerzos metodológicos. Reconociendo que no es posible continuar dando:

a) Sólo preponderancia a la componente tecnológica y obteniendo soluciones óptimas únicamente para ella, manipulando todas las componentes de los sistemas como objetos no-humanos, basándose en la mayoría de los casos en

reglas de racionalidad meramente económicas, que no consideran además la dinámica de los contextos que hacen rápidamente obsoletas las soluciones óptimas; se falla así, en reconocer en el factor humano toda su dimensión.

b) Sólo preponderancia a la componente social basándose sólo en teorías no comprobadas experimentalmente, manipulándola en sus aspectos psicosociales para que se acople a la tecnología.

Trist y Emery remarcaron la necesidad de buscar la "optimización conjunta" de los social y lo técnico, desarrollar y usar conceptos, métodos, técnicas y herramientas que conjuguen los aspectos cualitativos y cuantitativos, lo objetivo y lo subjetivo, que consideren a las componentes humanas del sistema y del contexto como sistemas plenos del propósito interactuando con la tecnología.

De su análisis de los diferentes tipos de contexto derivan la necesidad de promover su concepto de ecología social con el que exploran las relaciones de los sistemas con sus supra-sistemas. El concepto de control que se ha utilizado en los sistemas, ha sido el basado en la retroalimentación negativa, con la que se atenúan o corrigen desviaciones observadas respecto a los resultados esperados; no se ha considerado que la retroalimentación positiva, con la que se estimula una desviación observada, pueda representar oportunidades que podemos aprovechar. El enfoque socio-técnico busca el balance adecuado de la retroalimentación negativa y positiva en los sistemas, tomándolos como abiertos, interactuando estrechamente con sus contextos.

El enfoque de sistema socio-técnico abre amplias perspectivas metodológicas para la resolución eficaz y eficiente de los problemas, al basarse en la investigación-acción no explícita. Se reconoce así que el que no exista un sólo método, no es un problema, es más importante la actitud de indagación, reflexión y el desarrolla de capacidades de aprendizaje y adaptación en el propio sistema. El enfoque de sistemas socio-técnico más bien explicita "un tema" para motivar el desarrollo de esas actitudes y aptitudes; un tema para el desarrollo de nuevos conceptos, métodos, técnicas y herramientas, un tema que coadyuva en la formulación y adopción de un nuevo paradigma de sistemas. Un

paradigma que además de contemplar el balance apropiado de los métodos, técnicas y herramientas de sistemas y de los aspectos sociales y técnicos permita hacer frente a los cambios rápidos y complejos del presente y del futuro.

El enfoque socio-técnico sintetiza en los sistemas sus aspectos que lo hacen únicos y trata de generalizar el conocimiento, la educación, el aprendizaje. La Intervención en un sistema socio-técnico no se maneja como una relación externa -experto-cliente", el Investigador desempeña el papel de un facilitador del cambio, que al mismo tiempo que promueve el aprendizaje y la adaptación dentro del propio sistema y de su contexto, buscando aprender él mismo. Se enfatiza así un proceso de aprendizaje mutuo, para lograr soluciones más eficaces y eficientes, en lo particular y en lo general, al enfrentar otras situaciones.

El proceso de aprendizaje que se promueve no se limita al de prueba y error, sino a corregir los errores y no repetirlos de nuevo, se busca el aprendizaje a través de la acción y el conocimiento. Considerando el principio de requisito de variedad de cibernética, para dar capacidades de control a los sistemas, el enfoque socio-técnico considera que dar redundancia, al sistema a través del número de sus elementos, teniendo la posibilidad de desechar aquellos que no funcionan, por un lado, produce limitación en la posibilidad del mejoramiento y desarrollo de los elementos, y por otro lado, al desechar los inoperantes, contamina el ambiente eco-social. Por esto se busca dar redundancia a los sistemas reconociendo o desarrollando la capacidad multi-funcional de sus elementos; así la redundancia funcional da a los elementos humanos su propia dimensión primordial como elementos plenos de propósito.

El estudio de sistemas socio-técnicos considerados como campos, dominios y redes, donde los procesos son fluidos y sin límites claros, fue iniciando, previéndose como un modo organizacional para entender diferentes fenómenos sociales. El enfoque socio-técnico toma las características de un holograma, en el que el todo está representado en todas las partes y que cualquier parte puede representar el todo. Estas características se consideran íntimamente relacionadas con el concepto de sistemas, se propone desarrollar

características holográficas en los sistemas para que las funciones necesaria para el todo, estén también en las partes.

El aprendizaje, la redundancia en funciones de los elementos del sistema y las características holográficas permiten tener la flexibilidad de adaptación para responder a la dinámica del acelerado y complejo cambio social y tecnológico de nuestra época. El enfoque socio-técnico promueve enriquecer el modo de percibir y apreciar la realidad, hacerlo no solo a través de los sentidos sino a través de la intuición; no solo con el pensamiento sino con los sentimientos.

Al ampliarse las perspectivas metodológicas no sólo se abren posibilidades de ejercitar la creatividad a este respecto, se abren también posibilidades de la creatividad en lo social y en lo tecnológico de manera conjunta. Se considera que si bien la creatividad puede ser una capacidad innata para algunas personas, también puede aprenderse a ser creativo. A través del enfoque socio-técnico se ha procedido a estudiar el modo de actuar de profesionales en la resolución de problemas, para poder incorporar de manera explícita, en la formación de nuevos profesionales, su experiencia, sabiduría y arte que utiliza en la vida práctica.

Las aportaciones de Trist y Emery con los sistemas socio-técnicos son tan relevantes y amplias, que como ya se mencionó, se pueden considerar que con ellas se formula un nuevo paradigma de sistemas. Sin embargo, ante la falta de conocimiento de estos esfuerzos metodológicos desarrollados con este enfoque, en algunas ocasiones se le califican de ser cualitativos o informales; por esto y por la tendencia del tipo de formación que reciben algunas profesiones, preponderantemente cuantitativas o formales, no se les da la importancia que este trabajo trata de subrayar.

MÁS AVANCES METODOLÓGICOS

Los conceptos de planeación que Ackoff comenzó a difundir desde 1968 se ampliaron y explicitaron metodológicamente en su concepto de planeación de la empresa de 1970. Desde entonces Ackoff también comenzó a proponer y difundir el concepto de las Ciencias

de los Sistemas Sociales, renovando, enriqueciendo y abriendo fronteras conceptuales y metodológicas para enfrentar los problemas presentes y futuros de nuestra sociedad, en estos esfuerzos integró también los conceptos de sistemas socio-técnicos junto con su amplia formación y experiencia metodológica. En estos esfuerzos, Ackoff añade sus aportaciones singulares con su propuesta de una nueva formalización (no matemática) sobre los conceptos sistémicos, que presenta en su obra sobre sistemas con propósito Así como su caracterización de la era de los sistemas 41 con la que propone un renovado modo de ver e interactuar con el mundo, la realidad, las organizaciones y la planeación. Entre los conceptos que destacan en estas aportaciones están los de adaptación y aprendizaje, así como el de desarrollo, diferenciándolo de crecimiento y expresándolo como un concepto fuertemente relacionado a la calidad de vida.

En este camino, en 1974 Ackoff enriqueció su concepto de planeación estratégica analizando las posibilidades de diferentes filosofías, actitudes y tipologías de planeación, hasta llegar a proponer e impulsar lo que llamó la planeación interactiva para enfrentar sistemas de problemas. Su propuesta metodológica para enfrentar esas situaciones problemáticas parte de los principios de:

- participación
- proceso continuo y
- del holismo

y su método contempla las fases interactuales de:

1. Formulación del Sistema de Problemas
2. Planeación de Fines
3. Planeación de Medios
4. Planeación de Recursos
5. Diseño de la Implantación y el Control.

En 1989, A. D. Hall expande, adapta y actualiza su metodología de la Ingeniería de Sistemas en su metodología de meta-sistemas. Su metodología la refiere como el estudio de la planeación, la acción y el comportamiento humano para la conceptualización, la planeación, el diseño, la producción, el uso y desechar sistemas sin considerar de qué

disciplina se trate. Su metodología de sistemas la define como un proceso multi-paradigmático, creativo, eficiente, multi-fases, multi-niveles para encontrar definir y resolver problemas complejos. Hall señala que el proceso que propone tiene su aplicabilidad en el método científico, la ciencia de la acción, la investigación de políticas, la ingeniería de sistemas, la investigación de operaciones, las ciencias de la administración, la cibernética, en el análisis de impacto ambiental, las leyes, la contabilidad, la historia y en general en las ciencias aplicadas. Define así sus meta-sistemas. La estructura y forma (morfología) de su metodología la rebela en sólo 4 dimensiones fundamentales:

- Tiempo
- Lógica
- conocimiento o contenido
- cultura-política-comportamiento.

En 1991 J.P. Van Gigch adapta y actualiza su Teoría General de Sistemas Aplicada presentándola como la modelación y meta-modelación en el diseño de sistemas, haciendo claro su énfasis en los modelos matemáticos, técnicas y herramientas de sistemas, pero expandiendo sus conceptos a meta-sistemas y meta-modelos quedando soslayados nuevamente los aspectos metodológicos de sistemas.

A fines de los años 70 y comienzo de los 80 empezó a gestarse lo que se ha identificado como el movimiento crítico y pensamiento sistémico. En este movimiento se destacan por sus aportaciones: R. L., Flood, M. C. Jackson y W. Ulrich .La preocupación de estos autores ha sido contribuir a la evolución y desarrollo de sistemas a través de explorar sistemas de metodologías y el diseño de sistemas para resolver sistemas de problemas. A la critica antes mencionada de los sistemas duros, ahora se agregan sistemáticamente la crítica al énfasis en su esquema de medios y fines optimizantes, pues conociendo frecuentemente las consecuencias sociales, el cambio social, la dificultad de definir problemas no estructurales y de modelar pluralidad, suponiendo que los "hechos" sociales son objetivos y enfatizando la medición cuantitativa.

Surgió también la crítica a los esfuerzos para enfrentar sistemas suaves, primordialmente al propuesto por Checkland entre otros. Si bien ellos contribuían a enfrentar sistemas de

problemas, con una orientación a procesos, se les critica ahora por su orientación fenomenológica, no cuantitativa, interesada con mejoramiento, basada en una teoría social interpretativa preocupada por entender las reglas sociales para gobernar la realidad social. Estos esfuerzos también se criticaron por su debilidad de consecuencias no anticipadas, su falta de credibilidad tratando de manejar poder y conflicto, su falta de claridad de una teoría del cambio organizacional, así como su falta de capacidad del manejo de la relación entre racionalidad y legitimidad.

Si bien se consideró también que los esfuerzos de metodología en sistemas suaves no habían sido críticos ellos mismos, el surgimiento de las críticas antes mencionadas, surgidas en parte, de entre sus seguidores, demuestra lo contrario, Jackson se había formado y colaborado por Checkland. A estas críticas se agregó también el debate sobre modernismo y postmodernismo.

En 1983 W. Ulrich publica su promesa como heurística crítica de sistemas para la planeación social en que integra, en base a su formación y experiencia al lado Churchman, su enfoque de desarrollo de una filosofía práctica. Refuerza así la posición de Churchman que puede parafrasearse como: no hay nada más práctico que una buena filosofía.

Su crítica se amplía al reconocer que los sistemas duros y suaves están dominados por la metáfora mecanicista y organicista; las ideas de sistemas sólo son usadas como instrumentos de racionalidad ("que" debe hacerse) es necesario ser crítico para reflejar las suposiciones a considerar en la búsqueda de conocimiento y de la acción racional, las ideas de sistemas se refieren a la totalidad de las condiciones relevantes en la que dependen los juicios teóricos o prácticos, la heurística es el proceso para descubrir cualquier falsa apreciación y ayudar a planeadores y a otros actores a descubrir cualquier falsa apreciación a través de la reflexión crítica.

Para conducir esa reflexión crítica desde el punto de partida Kantiano de la polémica, Ulrich propone la consideración de 12 cuestionamientos límites en que partiendo del "debe ser", premisa normativa, se fluye al diseño concreto del sistema:

- ¿Quién es (debe ser) el cliente del sistema diseñado?

- ¿Cuál es (debe ser) el propósito del sistema diseñado?
- ¿Cuál es (debe ser) la medida del éxito?
- ¿Quién es (debe ser) el decisor?
- ¿Qué condiciones de planeación e implantación son (deben ser) controladas por el decisor?
- ¿Cuáles son (deben ser) las condiciones ambientales no controladas por el decisor?
- ¿Quién es (debe ser) involucrado como planeador?
- ¿Quién es (debe ser) involucrado como experto y cuál es la forma de su experiencia)?
- ¿Dónde busca (debe buscar) el involucrado garantía del éxito en la planeación?
- ¿Quién entre los involucrados representa (debe representar) los intereses de los afectados?
- ¿Tienen (deben tener) los afectados la oportunidad de emanciparse ellos mismos de los expertos?
- ¿Qué visión del mundo remarca (debe remarcar) el diseño del sistema? ,

FIood y Jackson desde principios de los 80 alentaron su ataque a construir las bases sólidas de conocimiento teórico del pensamiento sistémico crítico, caracterizándolas por:

- Buscar y demostrar conciencia crítica examinando suposiciones y valores asociados con el sistema actual o el diseño propuesto.
- Buscar y desplegar conciencia social reconociendo que presiones sociales llevan a preferencias para el uso de metodologías especiales.
- Enfocarse a la emancipación humana, buscando alcanzar oportunidad para el potencial del individuo y autodesarrollo.
- Comprometido con la complementariedad, con desarrollo informado de todas las posiciones del pensamiento sistémico a nivel teórico, viendo los diferentes enfoques como fortalezas y no como debilidades.

- Comprometido con la complementariedad, con el uso informado de metodologías de sistemas, sintiendo la necesidad de una meta metodología que respete todas las características transformando pensamiento en acción.

Así, en 1991 Flood y Jackson presentan su propuesta como un nuevo modo para planear, diseñar y evaluar en su Intervención Total en Sistemas, cuyas bases filosóficas principales son:

a) La complementariedad
b) La conciencia social
c) La emancipación humana

Basándose en los principios de: multi-modelos, metáforas, multi-metodologías, coclicidad sistémica, participación de facilitador y actores. Su meta-metodología consiste en la interacción de las fases:

- Creatividad
- Selección
- Implantación

Además de sus propuestas, estos últimos autores, conjuntamente con otros, plantean explorar en el futuro diversos caminos metodológicos que amplíen los prospectos sistémicos para su desarrollo. Entre estos caminos proponen el concepto de Liberalidad en Teoría de Sistemas en que se posibilita la "Liberalidad en Teoría" de Sistemas o la Liberalidad en "Teoría de Sistemas" o cualquier otra combinación.

En base a lo anterior los pasos a seguir son los siguientes:

1. Metodología

De acuerdo al paradigma de análisis de los sistemas duros y blandos tenemos:

Fases en el proceso de diseño de los sistemas o paradigma de sistemas

El ciclo de toma de decisiones puede dividirse en tres fases distintas y aplicarse al proceso del diseño de sistemas, como se muestra en la figura 1. Estas fases son como sigue:

1. Fase de diseño de políticas o pre-planeación
2. Fase de evaluación
3. Fase de acción-propuesta de implementación

Fase I. Diseñó de políticas o pre-planeación es la fase durante la cual:

- Se llega a un acuerdo de lo que es el problema.
- Los autores de decisiones llegan a una determinación de sus cosmovisiones (premisas, supuestos, sistemas de valor y estilos cognoscitivos).
- Se llega a un acuerdo sobre los métodos básicos por los cuales se interpretaran las pruebas.
- Se llega a un acuerdo sobre qué resultados (metas y objetivos) esperan los clientes (expectativas) y los planificadores (promesas).
- Se inicia la búsqueda y generación de alternativas

Fase 2. La evaluación consiste en fijar las diferentes alternativas propuestas para determinar el grado en el cual satisfacen las metas y objetivos implantados durante la fase anterior. La evaluación incluye:

1. Una identificación de los resultados y consecuencias derivados de cada alternativa.
2. Un acuerdo de los atributos y criterios elegidos con los cuales se evaluaran los resultados, representa verdaderamente las metas y objetivos preestablecidos a satisfacer.
3. Una elección de la medición y modelos de decisión, los cuales se usaran para evaluar y comparar alternativas.
4. Un acuerdo entorno al método por el cual se hará la elección de una alternativa en particular.

Fase 3. La propuesta de implantación de la acción es la fase durante la cual el diseño elegido se realiza, la propuesta de implantación incluye todos los problemas "malos" de:

1. Optimización, que describe dónde está la "mejor" solución.
2. Sub-optimización, que explica porqué no puede lograrse la "mejor" solución.

3. Complejidad, que trata con el hecho de que, de tener solución, debe simplificarse la realidad, pero para ser real, las soluciones deben ser "complejas".
4. Conflictos, legitimación y control, son problemas que afectan, pero no son exclusivos de la fase de implantación del diseño de sistemas.
5. Una auditoría o evaluación de los resultados obtenidos de la propuesta de implementación del diseño de sistemas, lo cual significa optimismo o pesimismo sobre si los objetivos pueden realmente satisfacerse y proporcionarse los resultados prometidos.

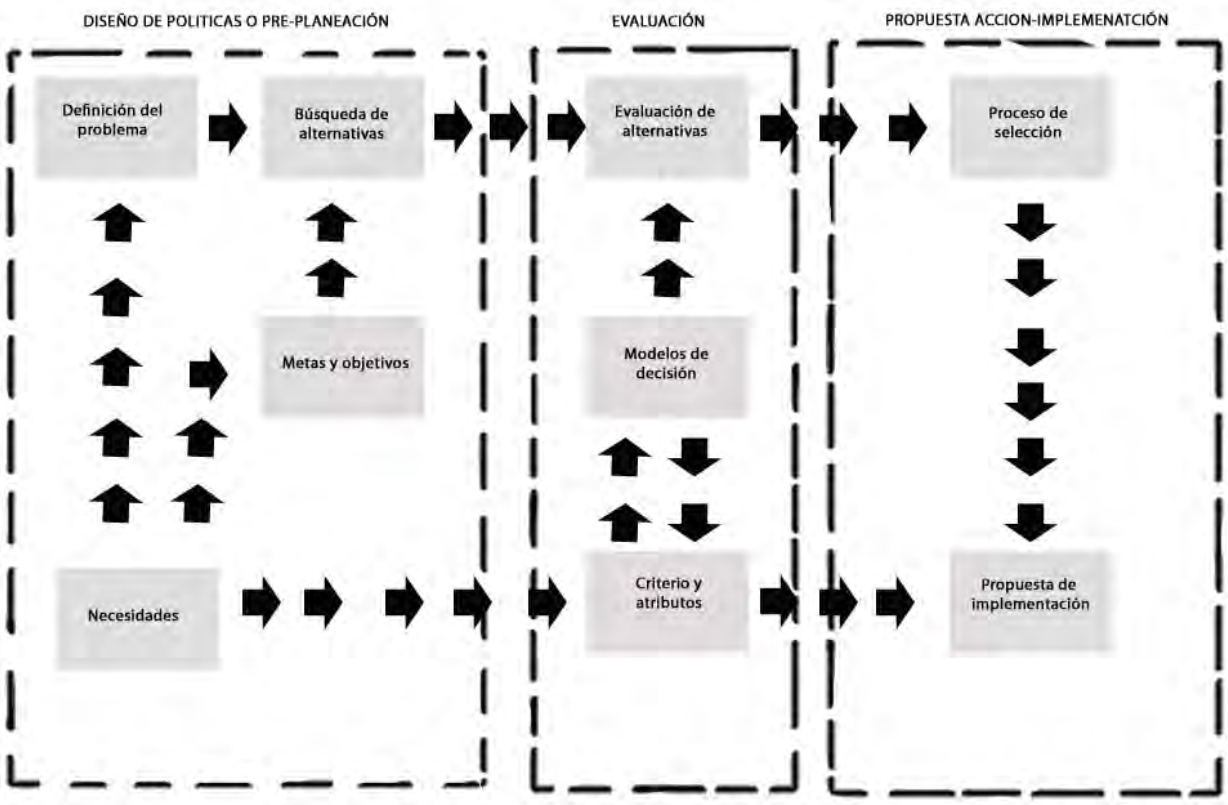


Figura 1 Ciclo de toma decisiones desintegrado en las tres fases del diseño de sistemas.

La metodología seleccionada es para sistemas duros: Metodología de Hall, la cual es la siguiente.

1.1. METODOLOGÍA DE HALL

Uno de los campos en donde con más intensidad se ha sentido la necesidad de utilizar conceptos y metodologías de Ingeniería de Sistemas es en el desarrollo de tecnologías. Esto se debe a que los sistemas técnicos, que sirven para satisfacer ciertas necesidades de los hombres, están compuestos de elementos interconectados entre sí de tal forma que se hace necesario pensar en términos de sistemas, tanto para el desarrollo de nueva tecnologías como para el análisis de la ya existentes.

1.1.1. METODOLOGÍA

Los pasos principales de la metodología de Hall son:

1. Definición del problema
2. Selección de objetivos
3. Síntesis de sistemas
4. Análisis de sistemas
5. Selección del sistema
6. Propuesta del Sistema

1. **DEFINICIÓN DEL PROBLEMA**: se busca transformar una situación confusa e indeterminada, reconocida como problemática y, por lo tanto, indeseable, en un apartado en donde se trate de definirla claramente. Esto sirve para:

a) Establecer objetivos preliminares.
b) El análisis de distintos sistemas.

De la definición del problema los demás pasos de la metodología dependen de cómo haya sido concebido y definido el problema. Si la definición del problema es distinta a lo que realmente es, lo más probable es que todo lo que se derive del estudio vaya a tener un impacto muy pobre en solucionar la verdadera situación problemática.

La definición del problema demanda tanta creatividad como el proponer soluciones. El número de posibles soluciones aumenta conforme el problema es definido en términos más amplios y que disminuyen al aumentar el número de palabras que denotan restricciones dentro de la restricción.

Existen dos formas en cómo nacen los problemas que son resueltos con sistemas técnicos:

a) La búsqueda en el medio ambiente de nuevas ideas, teorías, métodos y materiales para luego buscar formas de utilizarlos en el organismo.
b) Estudiar la organización actual y sus operaciones para detectar y definir necesidades.

Estas dos actividades están estrechamente relacionadas y se complementan una a otra.

Investigación de necesidades

Las necesidades caen dentro de tres categorías.

a) Incrementar la función de un sistema. Hacer que un sistema realice mas funciones de las actuales.
b) Incrementar el nivel de desempeño. Hacer que un sistema sea más confiable. Más fácil de operar y mantener, capaz de adaptarse a niveles estándares más altos.
c) Disminuir costos, hacer que un sistema sea más eficiente.

Investigación del medio ambiente

Se trata de entender y describir el medio ambiente en donde se encuentra la organización, “entre otras cosas, se realiza un peinado del medio ambiente en búsquedas de nuevas ideas, métodos, materiales y tecnologías que puedan ser utilizados en la satisfacción de necesidades”. De este último se desprende, el criterio para decidir si algo que existe en el medio ambiente es útil para el organismo y, está en función de las necesidades de esta última.

2. SELECCIÓN DE OBJETIVOS

Se establece tanto lo que esperamos del sistema, como los criterios bajo los cuales mediremos su comportamiento y compararemos la efectividad de diferentes sistemas.

Primero se establece, que es lo que esperamos obtener del sistema, así como insumos y productos y las necesidades que este pretenda satisfacer.

Ya que un sistema técnico se encuentra dentro de un supra-sistema que tiene propósitos, aquel debe ser evaluado en función de este. No es suficiente que el sistema ayude a satisfacer ciertas necesidades. Se debe escoger un sistema de valores relacionados con los

propósitos de la organización, mediante el cual se pueda seleccionar un sistema entre varios y optimizarlo. Los valores más comunes son: utilidad (dinero), mercado, costo, calidad, desempeño, compatibilidad, flexibilidad o adaptabilidad, simplicidad, seguridad y tiempo.

Los objetivos deben ser operados hasta que sea claro cómo distintos resultados pueden ser ocasionados a ellos para seleccionar y optimizar un sistema técnico.

Cuando un sistema tiene varios objetivos que deben satisfacerse simultáneamente, es necesario definir la importancia relativa de cada uno de ellos. Si cada objetivo debe cumplirse bajo una serie de valores a estos también debe a signarse un peso relativo que nos permita cambiarlos en el objetivo englobador.

3. SÍNTESIS DEL SISTEMA

Lo primero que se debe hacer es buscar todas las alternativas conocidas a través de las fuentes de información a nuestro alcance. Si el problema ha sido definido ampliamente, él número de alternativas va a ser bastante grande. De aquí se debe de obtener ideas para desarrollar distintos sistemas que puedan ayudarnos a satisfacer nuestras necesidades. Una vez hecho esto, se procede a diseñar (ingeniar) distintos sistemas.

En esta parte no se pretende que el diseño sea muy detallado. Sin embargo, debe de estar lo suficientemente detallado, de tal forma que los distintos sistemas puedan ser evaluados.

3.1. DISEÑO FUNCIONAL

El primer paso es listar los insumos y productos del sistema. Una vez hecho esto, se listan las funciones que se tienen que realizar para que dados ciertos insumos se obtengan ciertos productos. Estas funciones se realizan o sintetizan mostrando en un modelo esquemático de las actividades y como éstas se relacionan. Todo lo que se desea en este punto es ingeniar un sistema que trabaje, la optimización del mismo no importa tanto en este punto.

4. ANALISIS DE SISTEMAS

La función de análisis es deducir todas las consecuencias relevantes de los distintos sistemas para seleccionar el mejor. La información que se obtiene en esta etapa se retroalimenta a las funciones de selección de objetivos y síntesis de sistema. Los sistemas se analizan en función de los objetivos que se tengan.

4.1. COMPARACION DE SISTEMAS

Una vez que todos los sistemas han sido analizados y sintetizados, el paso siguiente es obtener las discrepancias y similitudes que existen entre cada uno de ellos. Existen dos tipos de comparación:

- Comparar el comportamiento de dos sistemas con respecto a un mismo objetivo.
- Comparar dos objetivos de un mismo sistema.

Antes que se lleve a cabo la comparación entre distintos sistemas, éstos deben ser optimizados, deben estar diseñados de tal forma que se operen lo más eficientemente posible. No se pueden comparar dos sistemas si aún no han sido optimizados.

5. SELECCIÓN DEL SISTEMA

Cuando el comportamiento de un sistema se puede predecir con certidumbre y solamente tenemos un solo valor dentro de nuestra función objetivo, el procedimiento de selección del sistema es bastante simple. Todo lo que se tiene que hacer es seleccionar el criterio de selección. Cuando el comportamiento del sistema no se puede predecir con certidumbre y se tienen distintos valores en función de los cuales se va a evaluar el sistema, no existe un procedimiento general mediante el cual se puede hacer la selección del sistema.

6. PROPUESTA DEL SISTEMA

Adquisición de componentes, creación e integración de los recursos necesarios para que el sistema funciones.

De acuerdo a la metodología seleccionada de sistemas a seguir se debe considerar una metodología para la construcción de ontologías, la cual nos ayudara al desarrollo de la propuesta.

Con las ontologías, los usuarios organizarán la información de manera que los agentes de software podrán interpretar el significado y, por tanto, podrán buscar e integrar datos mucho mejor que ahora. Gracias al conocimiento almacenado en las ontologías, las aplicaciones podrán extraer automáticamente datos de las páginas web, procesarlos y sacar conclusiones de ellos, así como tomar decisiones y negociar con otros agentes o personas. Por ejemplo, un agente inteligente que busque un vino que satisfaga las preferencias de un usuario, usará las ontologías vinícolas para elegir el vino (color, sabor, olor, embotellado) y empleará las ontologías empresariales para encargarlo a alguna tienda y regatear en el precio (siempre que se pueda). Otro ejemplo: mediante las ontologías, un agente encargado de comprar viviendas se podrá comunicar con agentes hipotecarios (de entidades bancarias) y con agentes inmobiliarios (de empresas constructoras e inmobiliarias). Por eso es necesario apoyarse en metodologías para la construcción de ontologías.

1.2. Metodologías para construir ontologías

En la literatura podemos encontrar diferentes metodologías para construir ontologías. Estas metodologías las podemos clasificar de acuerdo a diferentes parámetros. Por un lado, existen metodologías para construir ontologías desde cero. Pero, también existen metodologías para construir ontologías a través de procesos de reingeniería. Finalmente, podemos encontrar otras para la construcción cooperativa de ontologías.

1.2.1. Metodologías para Construir Ontologías a partir de cero

Sobre 1990 (Lenat el at.; 1990) se publicaron los pasos generales y algunos puntos interesantes relacionados con el proceso de desarrollo de CyC. Posteriormente, en (Uschold et al.; 1995) se comentó la metodología empleada para la creación de la ontología TOVE (Toronto Virtual Enterprise) para modelar organizaciones. Al año siguiente, estos autores propusieron unas reseñas metodológicas para construir ontologías (Uschold et al, 1996). Se presentó un método para construir ontologías para redes eléctricas como parte del proyecto KACTUS (Bernaras et al, 1996). Methontology (Fernández et al.; 1997) apareció simultáneamente y fue extendido posteriormente (Fernández-López et al.; 1999), (Fernández-López et al.; 2000). En 1997, se propuso otra metodología basada en la ontología SENSUS (Swartout et al.; 1997).

Procedamos a continuación a describir un conjunto de diferentes metodologías para la construcción de ontologías a partir de cero.

1.2.2. Metodología CyC

La metodología CyC (Lenat et al, 1990) consiste en varios pasos. En primer lugar hay que extraer manualmente el conocimiento común que está implícito en diferentes fuentes. Una vez que tengamos suficiente conocimiento en nuestra ontología, podemos adquirir nuevo conocimiento común usando herramientas de procesamiento de lenguaje natural o aprendizaje computacional. Así se construyó la ontología CyC. Esta metodología recomienda los siguientes pasos:

- Codificación manual de conocimiento implícito y explícito extraído de diferentes fuentes.
- Codificación de conocimiento usando herramientas software.
- Delegación de la mayor parte de la codificación en las herramientas.

1.2.3. Metodología de Construcción de Ontologías de Uschold y King

Esta metodología (Uschold et al.; 95) propone algunos pasos generales para desarrollar ontologías, a saber:

1) Identificar el propósito;
2) Capturar los conceptos y relaciones entre estos conceptos y los términos utilizados para referirse a estos conceptos y relaciones;
3) Codificar la ontología.

La ontología debe ser documentada y evaluada, y se pueden usar otras ontologías para crear la nueva. De esta forma se creó la Enterprise Ontology. Esta metodología recomienda los siguientes pasos:

a) Identificar propósito
b) Capturar la ontología
c) Codificación
d) Integrar ontologías existentes
e) Evaluación
f) Documentación

1.1.4. Metodología de Construcción de Ontologías de Uschold y King

En esta metodología, (Grüninger et al.; 95), el primer paso es identificar intuitivamente las aplicaciones posibles en las que se usará la ontología. Posteriormente, se usa un conjunto

de preguntas en lenguaje natural, llamadas cuestiones de competencia, para determinar el ámbito de la ontología. Se usan estas preguntas para extraer los conceptos principales, sus propiedades, relaciones y axiomas, los cuales se definen formalmente en Prolog. Por consiguiente, ésta es una metodología muy formal que se aprovecha de la robustez de la lógica clásica y que puede ser usada como guía para transformar escenarios informales en modelos computables. Esta metodología, que se usó para construir la ontología TOVE, recomienda los siguientes pasos:

- Escenarios motivantes
- Cuestiones informales de competencia
- Terminología formal
- Cuestiones formales de competencia
- Axiomas formales
- Teoremas de completitud

1.2.5. Metodología KACTUS

En esta metodología (Bernaras et al, 1996) se construye la ontología sobre una base de conocimiento por medio de un proceso de abstracción. Cuantas más aplicaciones se construyen, las ontologías se convierten en más generales y se alejan más de una base de conocimiento. En otras palabras, se propone comenzar por construir una base de conocimiento para una aplicación específica. A continuación, cuando se necesita una nueva base de conocimiento en un dominio parecido, se generaliza la primera base de conocimiento en una ontología y se adapta para las dos aplicaciones, y así sucesivamente. De esta forma, la ontología representaría el conocimiento consensuado necesario para todas las aplicaciones. Esta metodología ha sido utilizada para construir una ontología para diagnosticar fallos, y recomienda seguir los siguientes pasos:

- Especificación de la aplicación
- Diseño preliminar basado en categorías ontológicas top-level relevantes
- Refinamiento y estructuración de la ontología

1.2.6. METHONTOLOGY

Methontology es una metodología para construir ontologías tanto partiendo desde cero como reusando otras ontologías, o a través de un proceso de reingeniería. Este entorno permite la construcción de ontologías a nivel de conocimiento, e incluye:

- Identificación del proceso de desarrollo de la ontología donde se incluyen las principales actividades (evaluación, gestión de configuración, conceptualización, integración, implementación, etc.);
- Un ciclo de vida basado en prototipos evolucionados; y
- La metodología propiamente dicha, que especifica los pasos a ejecutar en cada actividad, las técnicas usadas, los productos a obtener y cómo deben ser evaluados.

Esta metodología está parcialmente soportada por el entorno de desarrollo ontológico WebODE. Esta metodología ha sido usada en la construcción de múltiples ontologías: ontología química, ontologías hardware y software, etc. Se proponen los siguientes pasos:

- Especificación
- Conceptualización
- Formalización
- Implementación
- Mantenimiento

1.2.7. Metodología SENSUS

La metodología basada en Sensus (Swartout et al, 1997) es un enfoque top-down para derivar ontologías específicas del dominio a partir de grandes ontologías. Los autores proponen identificar un conjunto de términos semilla que son relevantes en un dominio particular. Tales términos se enlazan manualmente a una ontología de amplia cobertura. Los usuarios seleccionan automáticamente los términos relevantes para describir el dominio y acotar la ontología Sensus. Consecuentemente el algoritmo devuelve el conjunto de término estructurados jerárquicamente para describir un dominio, que puede ser usado como esqueleto para la base de conocimiento. Esta metodología sirvió para construir la ontología Sensus y recomienda los siguientes pasos:

- Tomar una serie de términos como semillas.
- Enlazarlos manualmente.
- Incluir todos los conceptos en el camino que va de la raíz de Sensus a los conceptos semilla.
- Añadir nuevos términos relevantes del dominio.
- Opcionalmente, añadir para aquellos nodos por los que pasan más caminos su subárbol inferior.

1.1.8. Metodología On To Knowledge

El proyecto OTK (Staab et al, 2001) aplica ontologías a la información disponible electrónicamente para mejorar la calidad de la gestión de conocimiento en organizaciones grandes y distribuidas. La metodología proporciona guías para introducir conceptos y herramientas de gestión de conocimiento en empresas, ayudando a los proveedores y buscadores de conocimiento a presentar éste de forma eficiente y efectiva. Esta metodología incluye la identificación de metas que deberían ser conseguidas por herramientas de gestión de conocimiento y está basada en el análisis de escenarios de uso y en los diferentes papeles desempeñados por trabajadores de conocimiento y accionistas en las organizaciones. Cada una de las herramientas de la arquitectura de OKT se centra en el desarrollo de aplicaciones dirigidas por ontologías y finalmente describe el uso y la evaluación de la metodología mediante casos de estudio, como por ejemplo, la ontología Proper o AIFB. Los siguientes pasos son recomendados por esta metodología:

- Estudio de viabilidad
- Comienzo
- Refinamiento
- Evaluación

1.1.9. TERMINAE

Terminae (Aussenac-Gilles et al, 2002) aporta tanto una metodología como una herramienta para la construcción de ontologías a partir de textos. Se basa en un análisis lingüístico de los textos, el cual se realiza mediante la aplicación de diferentes herramientas para el procesamiento del lenguaje natural. En particular se usan dos herramientas:

1. Syntex para identificar términos y relaciones; y
2. Caméléon para identificar roles o relaciones.

Estas herramientas se basan en la misma hipótesis lingüística: el significado de las frases y las palabras es específico para un dominio y puede ser inferido de la observación de regularidades en documentos. La metodología funciona como sigue. Mediante la aplicación de Syntex obtenemos una lista de posibles palabras y frases del texto y algunas dependencias sintácticas y gramaticales entre ellas. Estos datos se usan como entrada para el proceso de modelado junto con el texto original. De esta forma, la identificación de conocimiento se basa en dos tareas que se realizan alternativamente:

- Explorar los resultados Syntex para identificar conocimiento importante o decidir cómo representar alguna información de acuerdo al uso de las palabras en el texto.
- Extraer sistemáticamente del texto tanto conocimiento como sea posible.

Cada pieza de conocimiento puede ser representada en el modelo de conocimiento de Terminae, cuyo lenguaje de representación de conocimiento posee las siguientes primitivas: fichero terminológico (términos), conceptos genéricos (clases), conceptos primitivos (instancias), y roles (relaciones). El siguiente paso es normalizar el conocimiento para obtener una ontología bien estructurada, donde cada concepto quede justificado por sus relaciones con otros conceptos. Esta metodología sugiere aplicar criterios diferenciadores para hacer explícitas las propiedades comunes y diferentes de un concepto con sus respectivos conceptos padre y hermanos debidas a sus roles. La última etapa es la formalización de la ontología en el lenguaje formal Terminae, que es un tipo de lógica descriptiva. Una función de clasificación sirve para comprobar la corrección de las definiciones de conceptos genéricos, ya que sólo pueden ser definidos si tienen roles diferenciados.

1.3. Metodologías para Reingeniería Ontológica

La reingeniería ontológica es el proceso de recuperar y mapear un modelo conceptual de una ontología implementada en otro modelo más adecuado, el cual sería re-implementado. Un método para la reingeniería ontológica fue presentado por el grupo de ontologías del Laboratorio de Inteligencia Artificial de la Universidad Politécnica de Madrid. Dicho método adapta el esquema de reingeniería de Chikofsky al dominio ontológico, teniendo tres actividades principales:

- Ingeniería inversa
- Reestructuración
- Ingeniería hacia adelante

1.3.1. Evaluación de ontologías

En se presenta una metodología para evaluar taxonomías (Welty and Guarino, 2001). Esta metodología hace uso de una serie de principios filosóficos basados en los conceptos de rigidez, unidad, identidad y dependencia. En particular, el usuario realiza anotaciones en cada propiedad de la taxonomía. Por ejemplo, se puede especificar si hay algún criterio que identifique a cada entidad de la propiedad, si la propiedad depende de otra propiedad, etc. De esta forma, el usuario evalúa la ontología teniendo en cuenta estas

anotaciones. Usando esta metodología, el grupo de Guarino ha construido ontologías top level de particulares y universales. Esta metodología de evaluación también da una serie de recomendaciones para construir ontologías:

- Etiquetar cada propiedad con meta-propiedades
- Centrarse en las propiedades rígidas
- Evaluar la taxonomía teniendo en cuenta restricciones entre meta-propiedades
- Considerar las propiedades no rígidas
- Completar la taxonomía con otras propiedades

Se presenta otra metodología de evaluación, ésta referida a la construcción correcta del contenido de la ontología, esto es, asegurar que sus definiciones implementan correctamente los requerimientos y cuestiones de competencia de la ontología (Gómez-Pérez, 2001). Se proponen los siguientes criterios para evaluar estas ontologías:

- ***Consistencia:*** Se refiere a obtener conclusiones contradictorias a partir de definiciones de entrada válidas. Una definición dada es consistente sí y solo si la definición individual es consistente y no se pueden inferir contradicciones usando otras definiciones y axiomas.
- ***Completitud:*** Este es un problema fundamental en las ontologías. Sólo se puede probar la incompletitud de una definición y, por tanto, la incompletitud de la ontología.
- ***Concisión:*** Una ontología es concisa si no contiene definiciones innecesarias o inútiles, redundancias explícitas y además no se pueden inferir redundancias usando otras definiciones y axiomas.
- ***Extensibilidad:*** Esta propiedad se refiere al esfuerzo necesario para añadir nuevas definiciones a una ontología y más conocimiento a las definiciones sin alterar el conjunto de propiedades bien definidas.
- ***Sensibilidad:*** Está relacionada con cómo afectan pequeños cambios al conjunto de propiedades bien definidas.

Esta metodología también se ocupa de los posibles errores realizados al estructurar el conocimiento del dominio en taxonomías, ontologías, y bases de conocimiento: errores de circularidad, errores de particiones de clases exhaustivas y no exhaustivas, errores de redundancia, errores gramaticales, errores semánticos y errores de incompletitud.

Esta metodología sugiere los siguientes pasos para construir ontologías:

- Evaluación de cada definición y axioma individual

- Evaluación de cada colección de definiciones y axiomas explícitamente en la ontología.
- Evaluación de las definiciones importadas de otras ontologías
- Evaluación de las definiciones que se pueden inferir usando otras definiciones y axiomas

1.2.2. Papel de las ontologías en el desarrollo de sistemas de información

En Inteligencia Artificial, la mayoría de metodologías existentes para construir ontologías de aplicación incluyen estos pasos:

- Construir la ontología del dominio
- Seleccionar los métodos de resolución de problemas (PSMs) para realizar las tareas
- Relacionar el conocimiento para PSMs con el dominio
- Añadir conocimiento factual del dominio

Algunos pasos por ejemplo, los dos primero se pueden realizar en paralelo y es probable que los tres primeros se realicen varias veces hasta que converjan PSMs y conocimiento del dominio. Para el cuarto paso se suelen emplear herramientas de adquisición automática de conocimiento. El resultado final será la ontología de aplicación para el sistema de información en cuestión.

Por lo que respecta al papel desempeñado por las ontologías en el diseño de Sistemas de Información, dicho papel no es muy diferente al desempeñado por otros enfoques como modelado semántico, metadatos, análisis y diseño de patrones, y librerías de módulos software reusables. Sin embargo, existen diferencias entre esos enfoques y el modelado ontológico. Así, las ontologías de método son similares a módulos software reusables, aunque el modelado ontológico permite especificar relaciones entre tareas genéricas y del dominio, así como decir qué partes del conocimiento del dominio usamos. El modelado ontológico se parece más al modelado semántico, puesto que los diagramas E/R son una especificación conceptual de datos usados por un sistema de información que incluye conceptos, propiedades y relaciones entre entidades, permitiendo la definición de subclases y herencia. Un diagrama E/R puede ser expresado en una ontología, pero las ontologías permiten una especificación más rica del dominio de conocimiento.

Las ontologías también pretenden capturar meta conocimiento y presentan la ventaja frente al modelado en base a metadatos que se puede expresar el conocimiento de forma más estructurada. Por último, el análisis orientado a objetos y el diseño de patrones son

similares a las ontologías del dominio, puesto que proporcionan descripciones de modelos usuales de objetos a nivel de meta conocimiento. Las ontologías tienen la ventaja de que su conocimiento está a un nivel superior y no tienen las mismas limitaciones que los métodos orientados a objetos.

2. Marco Teórico

El objetivo de este capítulo es exponer la motivación de este trabajo situándolo en el contexto de la naciente Web Semántica que constituye la visión de muchos grupos de investigación, organizaciones de estandarización y empresas. En la primera parte de esta capitulo se explican los objetivos y el concepto de esta siguiente Web diseñada para máquinas. Como se verá uno de los retos más importantes que tienen que superar para poder emular en éxito que tuvo internet es el acervo de masa crítica de contenido. En la segunda parte de este capítulo se hace un recorrido por las aproximaciones, aplicaciones y tecnologías existentes que trabajan en este sentido.

2.1. Contexto

Este trabajo pretende contribuir a la construcción de una masa crítica[1] de la siguiente generación de la Web, llamada Web Semántica. En esta sección se describe la iniciativa de la Web Semántica desde sus orígenes hasta su estado actual. Haciendo una analogía con la WWW se identifican algunos problemas que se deben resolver para que esta iniciativa tenga éxito.

2.2. La Web Semántica

En los últimos años la Web ha adquirido rasgos característicos que sugieren su propia evolución: una mejor interacción con los usuarios, la inclusión de bases de datos, la generación dinámica de páginas , una mejor adecuación ante el usuario y su constitución como plataforma común de múltiples aplicaciones.

No obstante, son variados los factores de la Web en que aún es posible mejorarla. Una propuesta de finales de los noventa del propio Berners Lee es la Web semántica con la que se pretende que las máquinas entiendan lo que contiene la Web, por medio de "agentes o representantes software, capaces de navegar y realizar operaciones por nosotros, para ahorrarnos trabajo y optimizar los resultados".

Berners Lee propone hacer una descripción de los "recursos de la web con representaciones procesables" para las personas y los programas con el propósito de que estos sustituyan a aquellas en trabajos rutinarios o irrealizables. El resultado sería una Web con más cohesión para facilitar más la localización, el intercambio y la integración de la información y los servicios con el fin de hacer más eficiente los recursos de la Web (Castells, 04).

[1] Se define como masa crítica de un fenómeno el número de individuos involucrados a partir del cual dicho fenómeno adquiere una dinámica propia que le permite sostenerse y crecer por si mismo.

En la Web actual puede localizarse información de cualquier tipo a través de un buscador de una manera cómoda, eficiente y económica sin acudir a un sitio específico. Sin embargo, debido al gran crecimiento a que ha llegado la Web en ocasiones una persona emplea demasiado tiempo o no obtiene algún resultado al realizar una búsqueda. Es muy complejo por medio de programas sustituir a los usuarios de la Web al efectuar búsquedas porque así de complejo es reproducir "en una máquina la capacidad de una persona para comprender los contenidos Web tal y como están codificados actualmente" (Castells, 04).

La Web ha revolucionado nuestra forma de trabajar dándonos acceso a contenido en cualquier lugar del mundo a cualquier hora. No obstante, este crecimiento desmesurado también nos trae problemas; al igual que sucede con las publicaciones tradicionales en papel del mundo somos incapaces de leerlas todas, seguir las novedades, filtrar las interesantes, resumirlas o hacer uso de ellas. Hoy en día la explotación de este inmenso potencial está muy limitada y el mayor freno a su explosión en términos de utilidad somos nosotros, nuestra capacidad de asimilarlo sin ayuda automática.

En los comienzos la WWW se ceñía al ámbito científico, ofreciendo un número abordable de páginas. El crecimiento del contenido en los años sucesivos dio lugar a los buscadores de internet que nos permitían elegir de toda la inmensa cantidad de documentos, aquellos que más nos puedan interesar. El 18 de setiembre del 2008, ComScore Inc. (NASDAQ: SCOR) líder en la medición del mundo digital publicó su análisis mensual **ComScore qSearch** referido al mercado de búsqueda norteamericano y latino. El análisis concluye que en agosto de 2009 los usuarios habían realizaron 11,7 millones de búsquedas, número similar al total del mes de julio sin embargo, Google ha aumentado su ventaja en la cuota de mercado del mercado de búsqueda en 1,1%.

El ranking de motores de búsqueda de agosto 2009 realizado por ComScore EE.UU., muestra que YouTube ha logrado un mayor nivel de tráfico de búsqueda respecto a Yahoo!. Si se consideran que YouTube cuenta con un motor de búsqueda independiente, quiere decir esto que Google cuenta con los dos principales puntos de tráfico de motores de búsqueda. Por lo tanto Google tiene cerca de cuatro veces el tráfico de búsqueda de Yahoo y más de diez veces el tráfico de búsqueda de los sitios de MSN de Microsoft.

En el informe se detallan un total de 11,7 mil millones de búsquedas formuladas por los usuarios de las cuales el 63% corresponde a Google en el segundo lugar fue a Yahoo (19,6%) Microsoft se encuentra en el tercer lugar (8,3%), Ask.com y AOL cuarto y quinto respectivamente ambos con menos del 5% de la cuota de mercado como se muestra en la Figura 1.

Figura 1 Reporte de búsquedas de agosto 2009 por ComScore EE.UU

Estos buscadores localizan palabras en documentos mediante algoritmos sofisticados de búsquedas y con una gran inversión en hardware. Y el contenido crece de manera exponencial.

En cuanto al número de usuarios pasamos de los 1000 en 1990 a los 200 millones del año 2000 Como se muestra en la Figura 2. Mientras en el mundo del papel nos hemos resignado a este desbordamiento en la era de los documentos en formato electrónico como miembros de la sociedad de la información nos resistimos.

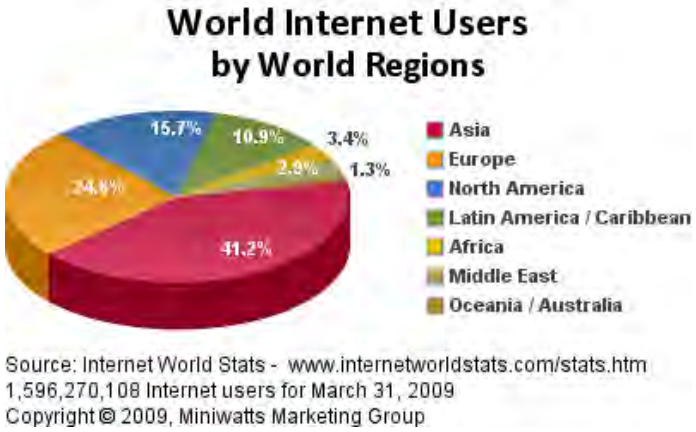


Figura 2 Estadísticas de usuarios por región realizada por Internet World Stats.

Uno de los padres de Internet y gran visionario, Tim Berners-Lee director del consorcio WWW (W3C) está promoviendo una solución para abordar esta explosión de contenido: dejar que los programas informáticos busquen y manejen la información de Internet por nosotros. Es decir, cambiar el paradigma actual de ***recuperación de documentos*** para y por humanos hacia la delegación de tareas a software. Los agentes de software como se llaman estos programas, son mucho más rápidos y eficientes para cierto tipo de tareas que el más hábil de los humanos. El problema que se plantea es que la Web, tal y como la conocemos hoy, fue concebida para humanos, contando con nuestra capacidad de entender el lenguaje natural, imágenes y sonidos. Los agentes de software van a tardar muchos años, si es que lo consiguen alguna vez, en llegar a este punto. Estos son las bases de la futura Web, concebida tanto para humanos como para agentes de software, la llamada: Web Semántica. En otras palabras la Web actual se basa en un lenguaje que define el "***como***": visualizar el texto e imágenes en una pantalla, mientras que la Web Semántica se basa en el "***que***": el significado de los textos e imágenes que vemos en las pantallas, permitiendo así su comprensión por agentes de software.

La forma en la que se procesará esta información no solo será en términos de entrada y salida de parámetros sino en términos de su SEMÁNTICA. La Web Semántica como infraestructura basada en metadatos aporta un camino para razonar en la Web, extendiendo a sus capacidades.

No se trata de una inteligencia artificial mágica que permita a las maquinas entender las palabras de los usuarios, es solo la habilidad de una máquina para resolver problemas bien definidos, a través de operaciones bien definidas que se llevarán a cabo sobre datos existentes bien definidos.

Para obtener esa adecuada definición de los datos, la Web Semántica utiliza **RDF**[2] y **OWL**[3], los dos estándares que ayudan a convertir la Web en una infraestructura global en la que es posible compartir y reutilizar datos y documentos entre diferentes tipos de usuarios.

A través de la Web Semántica se pretende clasificar, dar estructura y registrar los recursos cuya semántica expresa puede ser procesada por las maquinas, manteniendo los principios de la Web actual para descentralizar, compartir, compatibilizar y facilitar al máximo acceso. Así, el reto principal de la Web Semántica es propiciar un

[2] 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.

[3] Es un mecanismo para desarrollar temas o vocabularios específicos en los que asociar esos recursos. Lo que hace OWL es proporcionar un lenguaje para definir ontologías estructuradas que pueden ser utilizadas a través de diferentes sistemas.

entendimiento entre las partes que lo construyen y utilizan: usuarios, desarrolladores y software, usando una de las ramas de la inteligencia artificial: la ***ontología***.

2.3. Ontologías

El procedimiento de creación o conversión de contenido para que sea entendido por aplicaciones software consiste en el uso de etiquetas semánticas (meta-datos[4]) que indiquen el significado del contenido que encierran. Al contrario que los lenguajes usados en la producción de contenido para consumo humano, donde las etiquetas o los caracteres de control indican distintas formas de visualizar el contenido (HTML, Látex, etc.) independientemente de su interpretación y significado.

VIAJE A LA FICCION, EL

Autor:	VARGAS LLOSA, MARIO
Editorial:	SANTILLANA EDICIONES GENERALES
Año de Edic:	2009
Colección:	ALFAGUARA
Formato:	RUSTICA
ISBN:	9786071102294
Edición:	1
Páginas:	248
Precio:	$179.00

```
style="font-size: 14px">VIAJE A LA FICCION, EL</td>
        </tr>
        <tr>
          <td  align="left"> </td>
          <td  align="left"> </td>
          <td  align="left"> </td>
          <td  align="center"> </td>
          <td colspan="5"  align="center"> </td>
        </tr>
        <tr>
          <td  align="left"> </td>
          <td colspan="2"  align="center" valign="top"
                        <table width="100%" border="0"
cellpadding="0" cellspacing="0">
                            <tr valign="top">
                                <td valign="top"
```

Figura 3 Obra literaria codificada en HTML para consumo humano

Como se puede ver en el ejemplo, en este tipo de lenguajes los sistemas tienen dificultades para inferir cuál es el titulo o el autor de la obra (Figura 3). Sin embargo, en el contenido de la Web Semántica las etiquetas añaden significado a los datos que rodean (Figura 4). En la concepción de la Web Semántica se ha optado por uso de lenguajes basados en etiquetas XML para dotar de semántica el contenido Web.

El lenguaje XML es un estándar sintáctico[5] muy extendido y usado en las aplicaciones desarrolladas hoy en día que garantiza la interoperabilidad entre plataformas, sistemas y lenguajes de programación. El problema surge en la interpretación del mismo. Al igual que antes era difícil inferir el significado del contenido ya que solamente se disponía de

[4] Los metadatos consisten en información que caracteriza datos. Los metadatos son utilizados para suministrar información sobre datos producidos. En esencia, los metadatos intentan responder a las preguntas “quién”, “que”, “cuando”, “donde”, “por qué” y “cómo”, sobre cada una datos que se de las facetas relativas a los documentan en un proyecto.

[5] Son aquellos que cumplen con todas las definiciones básicas de formato y pueden, por lo tanto, analizarse correctamente por cualquier analizador sintáctico (*parser*) que cumpla con la norma.

información de aspecto sobre el contenido ahora el software debe inferir en el significado de las etiquetas.

VIAJE A LA FICCION, EL

Autor:	VARGAS LLOSA, MARIO
Editorial:	SANTILLANA EDICIONES GENERALES
Año de Edic:	2009
Colección:	ALFAGUARA
Formato:	RUSTICA
ISBN:	9786071102294
Edición:	1
Páginas:	248
Precio:	$179.00

```
<!-- ..............................................
<xsl:element name="meta">
<xsl:attribute name="name">dc.title</xsl:attribute>
<xsl:attribute name="content">
<xsl:value-of select="ead/eadheader/filedesc/titlestm
<xsl:text>  </xsl:text>
<xsl:value-of select="ead/eadheader/filedesc/titlestm
</xsl:attribute>
</xsl:element>
<xsl:element name="meta">
<xsl:attribute name="name">dc.author</xsl:attribute>
<xsl:attribute name="content">
<xsl:value-of select="ead/archdesc/did/origination"/>
</xsl:attribute>
</xsl:element>
</head>
```

Figura 4 Obra literaria codificada con etiquetas semánticas: anotaciones

La solución adoptada es la de asociar las etiquetas a modelos semánticos consensuados y estandarizados. Estos modelos se organizan en capas, donde el más básico permite modelar conceptos, atributos y relaciones y a medida que se especializa permite definir productos vendibles, libros, transacciones etc. Estos modelos se les denominan ***Ontologías.***

Originalmente el término ontología proviene de la filosofía, remontándose hasta los tiempos en los que Aristóteles trataba de clasificar los objetos que había en el mundo. Dentro de la Inteligencia Artificial, se adapto el término para describir la parte del mundo que puede ser representado en un programa. Una ontología es un modelo de almacenamiento compartido y común de un dominio descrito mediante conceptos, algunos de sus atributos y relaciones entre ellos (Weigand, 97). Las ontologías también pueden definirse como una especificación formal y explicita de una conceptualización compartida (Studer et al. 98), donde:

- La **conceptualización**[6] se refiere a la construcción de un modelo abstracto de algún dominio o fenómeno mediante la identificación de sus conceptos relevantes.
- Que sea **explicito** significa que los conceptos y las relaciones entre ellos se enumeran de manera explícita.

[6] Es una perspectiva abstracta y simplificada del conocimiento que tenemos del "mundo" y que por cualquier razón queremos representar. Esta representación es nuestro conocimiento del “mundo”, en el cual cada concepto es expresado en términos de relaciones verbales con otros conceptos y con sus ejemplos “del mundo real” (relaciones de atributo, etc., no necesariamente jerárquicas) y también con relaciones jerárquicas (la categorización o asignación del objeto a una o más categorías) múltiples (el objeto pertenece a diversas jerarquías contemporáneamente lo que quita totalmente el aspecto exclusivamente jerárquico a la conceptualización).

- Que se **formal** asegura que las ontologías son procesables por aplicaciones software.

- Que sea **compartido** hace alusión a que modelan el dominio de manera consensuada por alguna comunidad de usuarios.

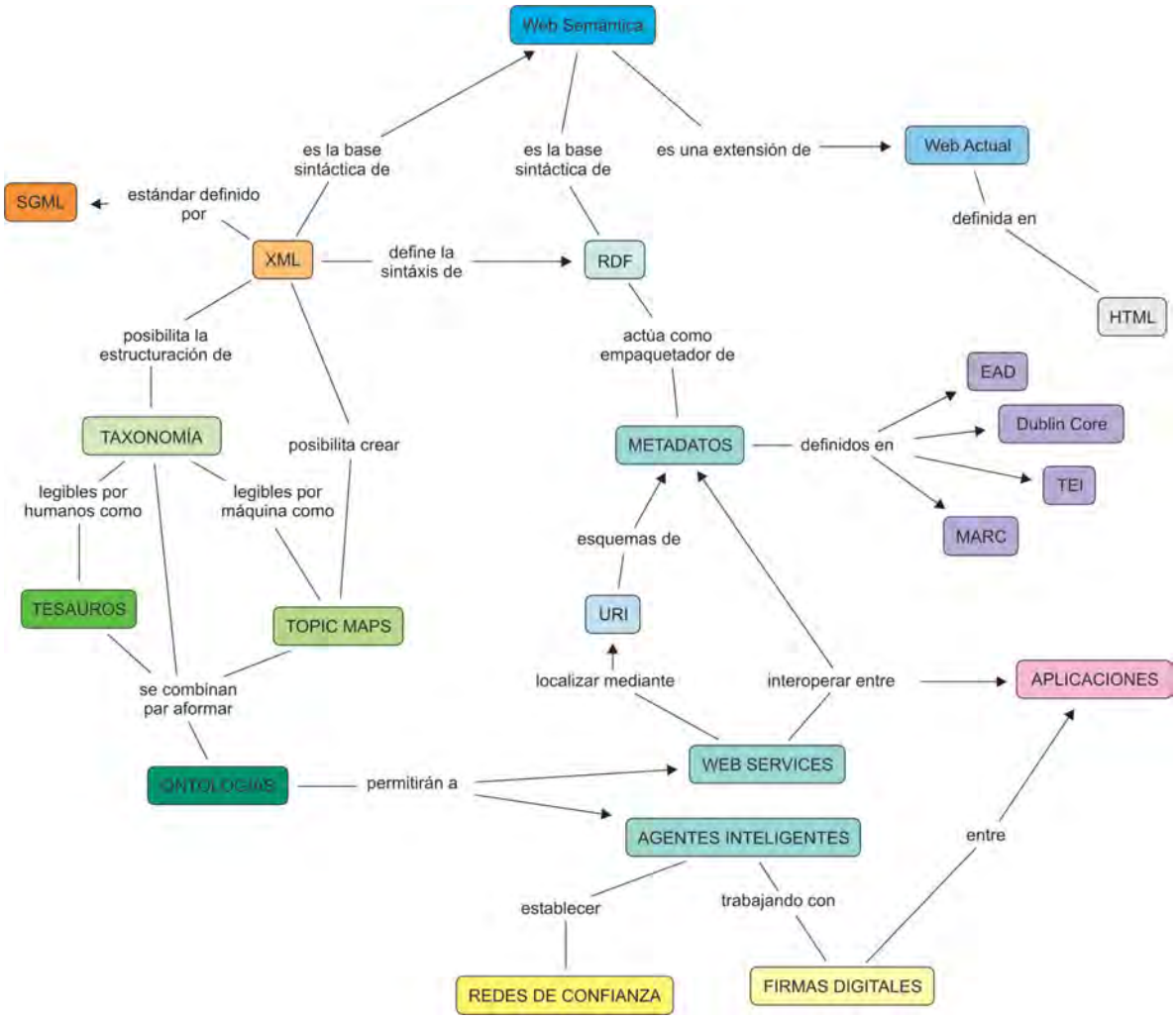


Figura 5 Papel de las Ontologías en la Web Semántica. Fuente: Mapa conceptual de la Web Semántica. Keilyn Rodríguez Perojo y Rodrigo Ronda León. "Web Semántica: un nuevo enfoque para la organización y recuperación de información en la web". *Acimed,* vol. 13, núm. 6, November-December 2005

Una ontología contiene conceptos que se describen mediante atributos y relaciones entre ellos así como también contiene axiomas que permiten validar e inferir contenido.

Actualmente las ontologías están reconocidas como una tecnología valiosa a la hora de mejorar la recuperación de la información Web (motores de búsqueda). La principal mejora proviene de la posibilidad de enriquecer las consultas con sinónimos, palabras más

especificas/generales, restricciones entre conceptos, etc. Esto proporciona una mejora significativa comparando con las técnicas utilizadas actualmente en la recuperación de la información, todas ellas basadas en palabras clave (términos no relacionados que aparecen en documentos). En OntoSeek (Guarino et al., 99) se utiliza una ontología para mejorar la calidad de la búsqueda dentro de las páginas amarillas. Hotbot [Hotbot] (como se muestra en la Figura 6) un importante motor de búsqueda, ha utilizado la ontología CYC para mejorar los resultados de la búsqueda. Por otra parte, observamos que la tecnología no está completamente madura para su uso dentro de un contexto comercial. Un ejemplo ilustrativo es que Hotbot ha deshabilitado esta característica a los pocos meses de su puesta en marcha.

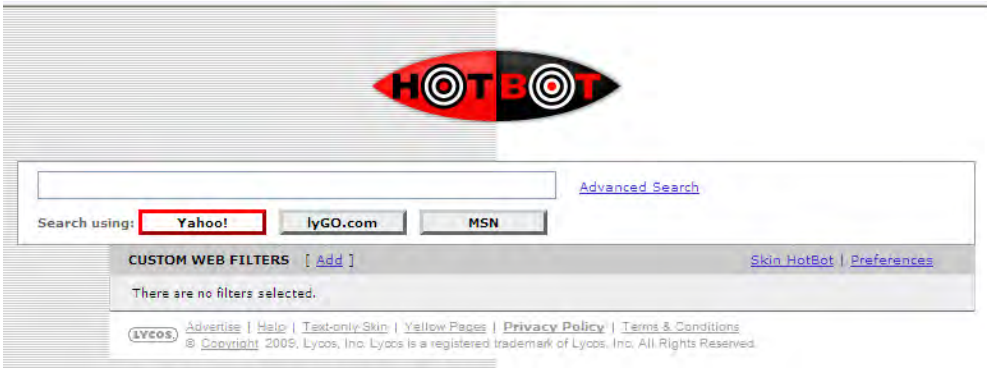


Figura 6 Motor de Búsqueda Hotbot

Las ontologías agrupan las ventajas tanto de gestores documentales, como de las bases de datos y representación del conocimiento:

- El ***conocimiento*** está almacenado en un formato consensuado por la comunidad habitual del dominio o por organismos de creación de estándares.
- ***Inferencia***: El conocimiento no tiene que estar explicito, sino que mediante reglas o axiomas se deduce información. Estos mecanismos se utilizan para las búsquedas o para comprobar semánticamente la validez de los datos.
- ***El contenido*** es procesable de manera automática. La creciente tendencia de creación de agentes de software autónomos se basa en la presencia de información anotada de acuerdo a ontologías consensuadas[7].

[7] Actualmente, los usuarios que buscan ontologías para incorporarlas a sus sistemas se basan únicamente en su experiencia e intuición y esto hace difícil que puedan justificar las elecciones tomadas. Esto es debido principalmente, a que no existe ningún método que indique al usuario qué ontologías son las más apropiadas para un nuevo sistema

El valor de disponer de un conocimiento modelado para su posterior explotación es incuestionable. Se puede proponer distintos formalismos para representar el modelo con distintos dominios formales y distintos grados de eficiencia según su propósito. El esfuerzo gastado en la codificación del contenido con anotaciones semánticas permite que la información sea explotada en un abanico de aplicaciones desde sistemas de publicación, buscadores avanzados, hasta sistemas de ayuda y tutores (García Serrano, et al 00). La comunidad de la Web Semántica invierte mucho esfuerzo en la creación y mejora de mecanismos de alimentación automática y semiautomática de estos modelos. Esta permite garantizar la calidad y homogeneidad de la información así como la posibilidad de la creación de una masa de contenido suficiente para la rentabilidad de los desarrollos de las nuevas aplicaciones.

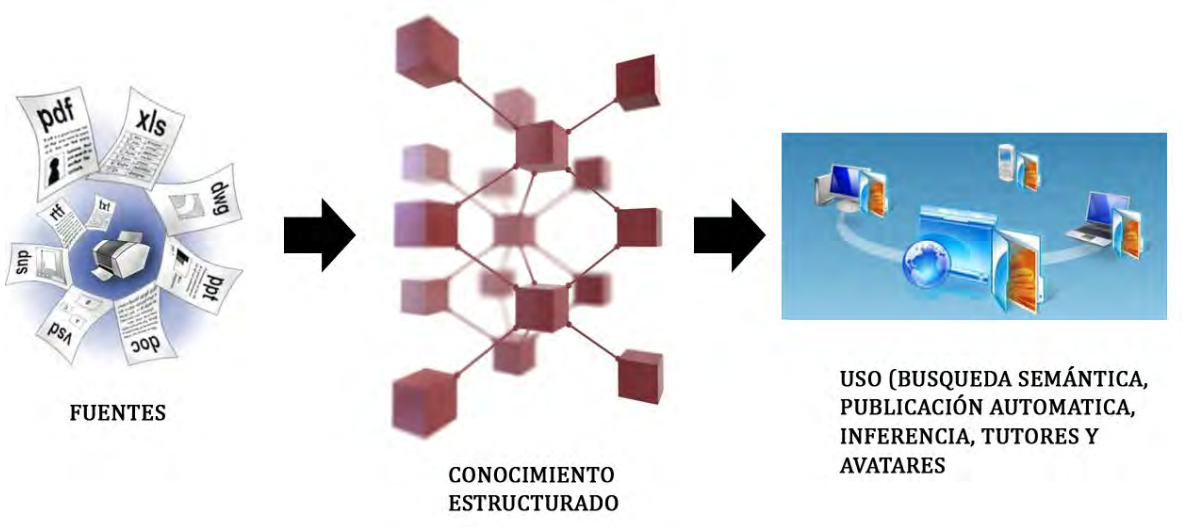


Figura 7 Aplicaciones de conocimiento estructurado

De acuerdo con (Miguel Ángel Abián, 2005), las ontologías son vocabularios comunes para los usuarios y las aplicaciones que pertenecen al campo de la inteligencia artificial. Agrega que cada persona utiliza ontologías con las que "representa y entiende el mundo que lo rodea", las que no son explicitas porque los significados son comúnmente conocidos y por lo tanto no es necesario tenerlas plasmadas en un documento, además de que no están organizadas jerárquica o matemáticamente en el cerebro de alguien.

Al igual que las personas, las maquinas carecen de ontologías explicitas, pero en su caso no comprenden el entorno y no pueden comunicarse entre sí. Si se pretende que las palabras sean procesadas por las maquinas, es necesario manejar las ontologías en forma explícita es decir, desarrollarlas en un documento o en "una forma que sea inteligible para las maquinas". En la elaboración de ontologías explicitas debe contemplarse como

mínimo un listado de términos con el significado de cada uno. De esta manera dos sistemas de información podrían interactuar por la ausencia de problemas semánticos.

Debido a que las ontologías almacenan conocimiento en una Web Semántica sería posible extraer información automáticamente y procesarla, como lo haría un agente de software[8] o agente inteligente cuando un usuario tiene el propósito de buscar, por ejemplo una computadora en razón de capacidad y calidad (Abián, 2005). Con las ontologías de carácter comercial se tendría que escoger un establecimiento para su adquisición y elegir el precio más conveniente.

2.3.1. Principales ventajas y desventajas de las ontologías

Hoy en día la principal ventaja que aportan las ontologías desde el punto de vista de las bases de datos y de la recuperación de información, tiene que ver con el empleo simultáneo de bases de datos de muy distinta naturaleza, características y formato, pudiendo examinar simultáneamente a las bases de datos para recuperar la información buscada. Estas bases de datos pueden ser de contenido tan dispar como documentación textual poco estructurada (la clásica que forma parte de los buscadores de internet), documentación fotográfica, documentación geográfica, museográfica, urbanística, espacios naturales, etc.

Otras de las grandes ventajas de las ontologías es la posibilidad de utilizar la información preexistente en dichas bases de datos a través de la propia ontología sin tener que renunciar a la base de datos original de partida, de manera que puedan convivir y seguir empleándose simultáneamente bases de datos iniciales y ontología. Tan solo debemos tener presente la necesidad de actualizar la ontología con los nuevos datos que vayan añadiéndose a las bases de datos.

Tampoco es necesario introducir de nuevo manualmente aunque podría hacerse, las bases de datos en la nueva estructura de la propia ontología que se pretenda desarrollar. Se puede automatizar el proceso de volcado de la información mediante el desarrollo informático de programas que transformen el formato de la base de datos inicial en el formato y lugar adecuado en la nueva estructura ontológica. Lógicamente, cuanto más estructurada se halle la información inicial, más sencilla es la transformación correspondiente.

Una de sus desventajas surge al tratar de sacar el máximo partido de las amplísimas posibilidades de recuperación que proporcionan las ontologías, lo que se consigue

[8] es una parte del software que actúa para un usuario u otro programa como agente. El agente tiene la autoridad de decidir cuándo una acción es apropiada (y si es apropiada). La idea es que los agentes no son estrictamente invocados para una tarea, sino que se activan ellos mismos.

mediante el desarrollo de ***buscadores denominados semánticos***. Estos buscadores semánticos involucran la programación de complejos algoritmos que echan mano de herramientas de procesamiento de lenguaje natural poco desarrollados aun. En consecuencia las ontologías posibilitarían una mejora sustancial en la precisión de la recuperación gracias a la posibilidad real de ***"entender"*** los términos y conceptos presentes en las preguntas, para ellos dependen de complejos algoritmos de comprensión del lenguaje natural que no han alcanzado todavía un desarrollo eficiente. En resumen, se supone una herramienta con grandes posibilidades que sin embargo, dependen para desarrollar todo su potencial de avances en otras áreas ajenas como la Inteligencia Artificial o el Procesamiento del lenguaje natural.

En síntesis las ontologías:

- Favorecen la comunicación entre personas, organizaciones y aplicaciones.
- Permiten la interoperación entre sistemas.
- Facilitan el razonamiento automático.
- Contribuyen a especificar los sistemas de software.

2.4. Agentes Inteligentes

El término ***"agente"*** describe una abstracción[9] de software, una idea o concepto similar a los métodos, funciones y objetos en la programación orientada a objetos[10]. El concepto de un agente provee una forma conveniente y poderosa de describir una compleja entidad de software, que es capaz de actuar con cierto grado de autonomía para cumplir tareas en representación de personas. Pero a diferencia de los objetos (que son definidos por métodos y atributos), un agente es definido por su propio comportamiento.

Los agentes son software que realizan operaciones para los usuarios con cierto grado de independencia o autonomía empleando para ello el conocimiento o la representación de los deseos del usuario (Gilbert et al 95). Otra posible definición define (Wooldridge, Jennings 95) los agentes como sistemas informáticos basados en hardware o software con las siguientes propiedades:

[9] Es un proceso mental que se aplica al seleccionar algunas características y propiedades de un conjunto de cosas del mundo real, excluyendo otras no pertinentes. En otras palabras, es una representación mental de la realidad

[10] basado en la idea de encapsular estado y operaciones en objetos. En general, la programación se resuelve comunicando dichos objetos a través de mensajes (programación orientada a mensajes).

- ***Persistencia***: el código no es ejecutado bajo demanda sino que se ejecuta continuamente y decide por sí mismo cuando debería llevar a cabo alguna actividad.
- ***Autonomía*****:** los agentes tienen la capacidad de seleccionar tareas, priorizarlas, tomar decisiones sin intervención humana, etc.
- ***Capacidad o habilidad social***: los agentes son capaces de tomar otros componentes a través de coordinación y comunicación que puedan colaborar en una tarea.
- ***Reactividad:*** los agentes perciben el contexto en el cual operan y reaccionan a este apropiadamente.

La investigación en tecnologías de agentes ha experimentado un gran impulso con la expansión del internet: ¿Quién no recuerda a todos esos bots para comprar CDs y libros en la Web? Todos esos agentes nunca llegaron a convertirse en un negocio real. Como una de las razones de su escaso éxito se apunta una baja disponibilidad de contenido y conocimiento en el formato que ellos hubiesen podido procesar. Fue una lección aprendida sobre trabajo en internet, que es una estructura abierta y sin control y los agentes tienen que encontrar y procesar el contenido ***ad-hoc***[11] para sus necesidades. Este tedioso proceso causo que la construcción de agentes fuese muy costosa y muy sensible a cambios en los documentos en la Web.

La aparición de la Web Semántica, da una solución natural a este problema y permite que los agentes inteligentes este en auge otra vez. El contenido está formalizado en lenguajes derivados del XML y con una semántica bien definida y consensuada. Mientras que el contenido anotado semánticamente constituye una base de hechos apta para procesamiento automático por parte de los agentes, la reciente iniciativa de los Servicios Web Semánticos, permite introducir el concepto de una funcionalidad definida y anotada semánticamente. Esto permite que los agentes no solo procesen el contenido estático si no que también invoquen funcionalidades sobre este contenido.

A pesar del desarrollo tecnológico y perfeccionamiento de robots y agentes, el crecimiento imparable de la Web se ha convertido en un verdadero problema para las

[11] Es una locución latina que significa literalmente «para esto». Generalmente se refiere a una solución elaborada específicamente para un problema o fin preciso y por tanto, no es generalizable ni utilizable para otros propósitos. Se usa pues para referirse a algo que es adecuado sólo para un determinado fin. En sentido amplio, ad hoc puede traducirse como «específico» o «específicamente».

técnicas de indización[12] y búsqueda de la información en la red y mantener actualizadas las bases de datos de los buscadores haciéndose cada vez más difícil. Los sistemas de indización centralizados no se pueden aplicar a toda la red debido al enorme tamaño de esta, a los recursos que se precisan para procesar y almacenar tal volumen de información y al ancho de banda que se consume. Todo ello conduce a la imposibilidad de que un único robot de indización cubra toda la Web. Por otro lado, muchos robots web causan tráfico extra y un desperdicio del ancho de banda porque varios robots recuperan el mismo documento para indizarlo y actualizarlo. Aunque existen muchas propuestas para resolver este problema tratándolo desde una perspectiva de red, la única opción por ahora, es elegir en cada caso concreto el buscador más apropiado para nuestros gustos y necesidades de información.

Como hecho curioso, a la derecha se muestra una imagen del chat-robot creado por ***Ikea*** que es capaz de responder a casi cualquier pregunta. Lo mejor es probar a hacerle alguna consulta para ver su funcionamiento como se muestra en la Figura 8. http://193.108.42.79/ikea-es/flash_files/bot.html

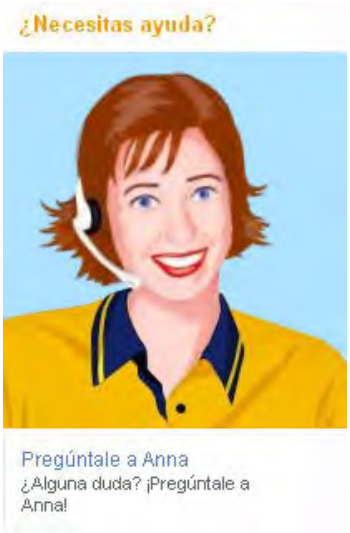


Figura 8 chat-robot creado por Ikea

La tabla presentada a continuación resume algunos de los aspectos más prometedores, limitaciones y retos de la tecnología de los agentes inteligentes.

	Web Semántica	***Servicios Web***	***Agentes Inteligentes***
Promesas	De palabras clave a	Interoperabilidad entre	Delegación de tareas

[12] De acuerdo a la norma ISO 5963 (1985) la indización es el proceso de describir o representar el contenido temático de un recurso de información. Este proceso da como resultado un índice de términos de indización que será utilizado como herramienta de búsqueda y acceso al contenido de recursos en sistemas de recuperación de información.

	aplicaciones inteligentes	aplicaciones en Internet.	
Limitaciones	Generación Manual de Metadatos	Semántica Compleja Composición Manual Descubrimiento Manual	Necesitan información en representaciones formales
Retos	Uso de ontologías para representar meta-datos	Automáticamente descubrir y componer servicios	Explotar el contenido de la Web Semántica

Tabla 1 Tabla Comparativa de tecnologías semánticas emergentes

2.4.1. Arquitectura

En la figura siguiente se muestra una posible arquitectura lógica de la futura Web, según está definida en uno de los nacientes proyectos cuyo objetivo es la construcción de la Web Semántica (Esperonto). En la parte superior se muestra la transición Web 1.0 a la Web 2.0 con diferentes tipos de contenido multimedia, cada uno de ellos expresado en formato y lenguajes distintos. Software como editores para la anotación o extractores semiautomáticos de datos puede convertirse este contenido en un lenguaje de la Web Semántica en consecuencia de lo anterior se muestra la transición de la Web 2.0 a la Web 3.0 o la llamada Web Semántica, este proceso deja rastro en los índices semánticos que actúan como índices de buscadores, actualizándolos para que contengan las últimas novedades y enlaces al nuevo contenido. Estos mecanismos en enrutamiento semántico ofrecen localizaciones de contenidos semánticos al software o agentes que lo soliciten como se muestra en la Figura 9.

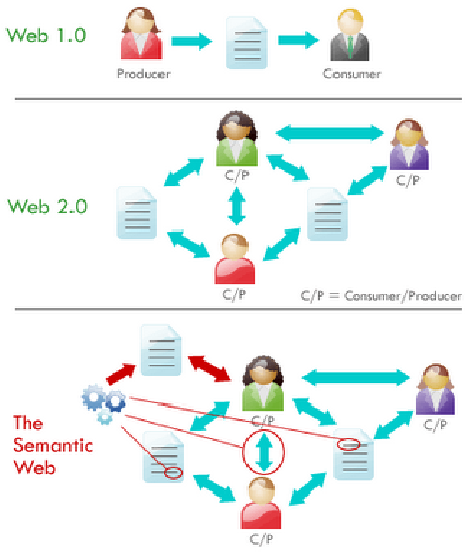


Figura 9 Visión de la Web Semántica (Esperonto)

2.4.2. Retos

Para entender hacia donde debe confluir la semántica en la Web y por qué es importante, se hace necesario entender grosso modo la estructura de lo que sería la Web semántica.

Si la semántica tiene que ver con la significación de palabras, en el caso de la Web, debe estar dada principalmente por la creación de documentos digitales (objetos digitales) pero con significado, de tal forma que ya no se hable de información si no de conocimiento. Ante esto se hace evidente el papel que cumplirán los metadatos para conseguir este fin, pero no pueden funcionar por si solos, requieren de unas bases tecnológicas que se explican a continuación.

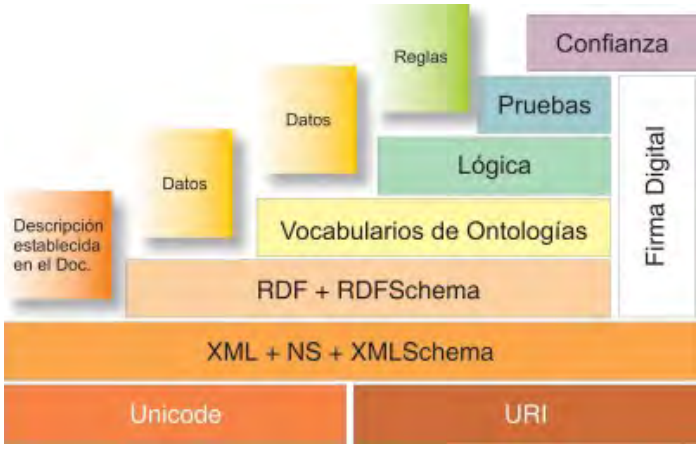


Figura 10 Capas de la Web Semántica

Como se aprecia en la Figura 10, la Web Semántica está conformada por varias capas, las superiores son las que dan significado a los contenidos en tanto que las inferiores dan el soporte tecnológico y son parte del usuario final en la pantalla de su ordenador o computadora. El profesional en información está de una forma u otra vinculado a cada capa, pero su principal papel está en la organización de los contenidos dentro de todo el proceso.

En vista que la recuperación por medio del XML y los metadatos no es suficiente precisa, se crea una capa superior, la de RDF (Resource Description Framework) que es el marco donde se indican los metadatos y la estructura semántica del recurso Web. Es aquí donde se inicia la Web Semántica, ya que es la primera capa con significado propio y que permite generar una primera relación entre el hombre y la máquina para que se entiendan. La idea del RDF es proporcionar un orden al XML de tal forma que la estructura del marcado sea coherente con lo que el ser humano realmente necesita.

De tal forma RDF tiene dos partes principales:

1. *El modelo de datos*: que es donde se ingresan los metadatos para que adquieran la semántica necesaria, basado en una estructura de tripleta donde aparecen 3 componentes: recurso, propiedad y sentencia.

2. *El esquema*: que es la parte donde se definen los vocabularios y se define la estructura y las relaciones que llevará el RDF para cada caso particular.

Las capas superiores son las que le dan ya el aspecto que se quiere de Web Semántica, sin embargo son igualmente las capas donde menos han intervenido en general profesionales en información. De estas capas las más importantes son las ontologías, estas se pueden definir como una especificación explicita y formal de una conceptualización tendiente a ser usada en un proceso basado en Inteligencia Artificial. Es por eso que el alcance máximo de la Web semántica es el uso de agentes inteligentes para la recuperación de información basados en el desarrollo de ontologías. Por último, toda la información de la Web Semántica debe tener validez que le den aspectos como las pruebas del sistema y la firma digital, todo esto debe llevar a una Web que sea confiable para el usuario, que no genere dudas. Es por esto que a la capa más superior se le denomina la Web de la confianza.

La Web Semántica no puede entenderse como un cambio en el paradigma de la profesión, sino más bien como una oportunidad de demostrar la adaptabilidad y la vigencia de la misma. La esencia del quehacer de la profesión no cambia; la tecnología ha estado presente siempre en la evolución de la misma, por tal razón no es momento para asustarse por los sistemas y la Internet. Por otra parte hay que tener en cuenta que la tecnología solo es una herramienta que permite mejorar la calidad de vida del hombre no es un fin en sí misma.

De acuerdo a lo anterior y si siguen los alcances que propone (Berners-Lee, 2001), la Ciencia de la Información se debe tender a conseguir la colaboración hombre-máquina, mediante el desarrollo de lenguajes documentales (Ontologías) que permitan llegar a esta meta y la unión de las teorías clásicas con las modernas.

Si se mira con detenimiento, existen trabajos que ya han tomado fuerza dentro de la Internet, las cuales pueden ser:

Desde la concepción de la Web:

- Creación de servicios Web (portales temáticos)
- Estudios de Accesibilidad
- Conformación de Objetos digitales
- Diseño de bibliotecas digitales
- Creación y gestión de contenidos en bibliotecas digitales, museos digitales, archivos digitales, etc.

Desde los usuarios:

- Capacitación de usuarios
- Alfabetización informacional
- Estudios de usabilidad
- Estudios de accesibilidad
- Servicios Web

Como ya se ha comentado, la promesa de la Web Semántica es hacer procesable el contenido en la red, para los agentes y las aplicaciones puedan actuar sobre él y así ofrecer servicios de valor añadido. La tecnología actual desarrollada en su mayoría en ámbitos académicos y sobre pequeños dominios de estudio (viajes, ocio, etc.), permite ver estas promesas hechas realidad, aunque no sean aplicaciones de la Web Semántica puras, sino mas bien aplicaciones pre-Web Semántica. Se catalogan así por que ofrecen funcionalidades avanzadas que requieren de conocimiento estructurado con cierto grado de consenso, pero este conocimiento se obtiene con costosos procesos elaborados ad-hoc (Knoblock et al 01).

Una de las maneras de obtenerlo aparte de codificación manual, es mediante software llamado Wrapper[13]. Los Wrapper se emplean en dos tipos de aplicaciones relacionadas: la integración de información y el acceso homogéneo en fuentes de información heterogéneas (bases de datos relacionales, bases de datos documentales, páginas Web, etc.…). Un Wrapper actúa sobre una clase de páginas para obtener un tipo de información, por ello en una aplicación real es precisa la construcción de un número grande de Wrappers lo cual, a pesar de que los lenguajes de descripción de Wrappers suelen ser sencillos y los Wrappers en si suelen ser programas bastante simples, supone un costo elevado. Por ello se ha abordado recientemente el aprendizaje automático de los Wrappers. Los Wrappers son muy sensibles a cambios en los documentos fuentes, aunque se trate de cambios mínimos relativos a la presentación o estructura. Existen varias propuestas y se han empleado varias técnicas de aprendizaje automático.

La Web Semántica en su visión ofrece el contenido ya codificado directamente entendible por agentes software, lo que permitirá la proliferación de aplicaciones semánticas de valor añadido. Para llegar al punto donde agentes inteligentes software comprenden el contenido de la red y realizan tareas por nosotros, hace falta plantear y realizar tareas por

[13] del inglés: envoltorio. De alguna manera envuelve el contenido de las fuentes online ofreciendo funcionalidades de acceso a su contenido como si este fuera estructurado (por ejemplo: usando lenguajes SQL [Ariadna] o usando especificaciones de acceso a ontologías). En el contexto de la Web Semántica podría traducirse como: extractor automático.

nosotros y resolver todavía muchos problemas. Algunos de ellos que se enumeran a continuación, constituyen verdaderos retos para los investigadores y las empresas de hoy.

2.4.3. Disponibilidad y Estabilidad de Lenguajes Semánticos

La Web actual está basada en el lenguaje HTML que especifica con qué aspecto debe visualizarse una página para que la puedan leer los usuarios humanos. El propio HTML no facilita mucho la labor de aplicaciones de recuperación de información y buscadores que solamente disponen del propio contenido y su aspecto para el proceso. En la Web Semántica, las páginas o documentos almacenarán no solamente el contenido sino que también la información consensuada sobre su significado y estructura. Esto permitirá que el contenido de la Web Semántica sea procesable por agentes o software, ofreciendo un gran abanico de posibilidades para nuevas tecnologías que realizarán en paso de aplicaciones de recuperación de documentos a aplicaciones a las que podamos delegar tareas.

Según la concepción de la Web Semántica por sus creadores, los lenguajes para la descripción de la Web Semántica forman una pirámide (figura 11), que en la base tiene el lenguaje XML que es puramente sintáctico y en las capas altas están lenguajes cada vez más ricos en semántica y más específicos de un dominio. Estos lenguajes están diseñados para representar significado y estructura del contenido (en forma de relaciones entre conceptos) ofreciendo un nivel adecuado de poder expresivo para representar la semántica de los recursos en la Web.

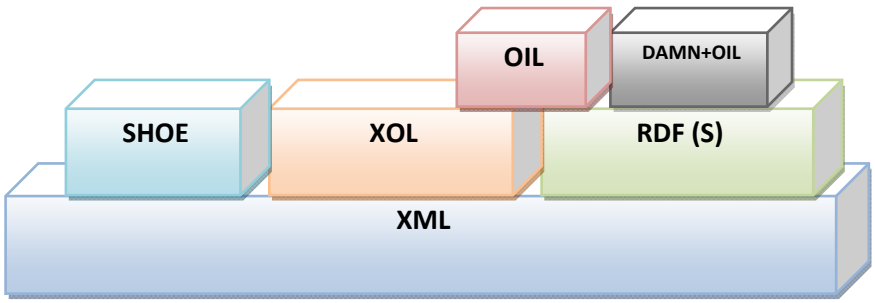


Figura 11 Pirámide de los lenguajes de la Web Semántica

Para poder edificar unas bases sólidas para el desarrollo de la Web Semántica y sus aplicaciones es recomendable monitorizar y seguir la evolución de los estándares en cuanto a lenguajes de representación. Esta labor de monitorización podrá prever los posibles problemas de compatibilidad y expresividad de distintos estándares y dirigir los nuevos desarrollos para que sufran el menor impacto posible consecuencia de un cambio.

2.4.4.Disponibilidad del Contenido

Al igual lo que sucedió con la World Wide Web, se puede prever que la existencia de una masa crítica será una condición necesaria para su extensión completa en las vidas de los usuarios y en los modelos de negocio de las empresas. Las primeras páginas creadas por usuarios de manera manual constituían todo el contenido existente en los comienzos de la WWW. Actualmente estas páginas en contraposición a las páginas creadas por procesos automáticos a partir de bases de datos (llamado contenido dinámico), son solo una mínima parte de la Web Actual. Según el estudio (Lawrence, Giles 99) y (Bergman, 01) la Web de contenido dinámico o también llamada Web profunda (Deep Web) es desde 400 a 550 veces más grande que la Web creada de manera manual.

Siguiendo con el paralelismo establecido con la WWW, el primer contenido de la Web Semántica es decir, contenido anotado semánticamente aparece en la comunidad científica y académica. Son pequeñas islas semánticas creadas de manera manual y anotada según ontologías simples. El proyecto $(KA)^2$ (Benjamins, et al 99) fue pionero en este campo, creando recursos de la Web Semántica en forma de anotaciones dentro de páginas personales de los miembros de la comunidad científica de adquisición de conocimiento. Hoy en día aunque la infraestructura de lenguajes para escribir contenido para la Web Semántica está siendo definida (RDFS, OIL, DAML; OWL, etc.), ya existe algún contendido disponible sobre todo en dominios experimentales de investigación y académicos.

La creación del contenido semántico de manera manual no es escalable para conseguir una masa crítica necesaria. Los procesos que automatizan y facilitan la creación de contenido se pueden dividir en dos grandes bloques:

- **Procesos de creación de nuevo contenido de la Web Semántica**: En este caso se engloban herramientas que permiten que un usuario cree nuevo contenido apto para la Web Semántica. Estas herramientas, en su mayoría editores, permiten un manejo amigable de la sintaxis de los lenguajes de la Web Semántica así como funcionalidades básicas para el trabajo con ontologías: pruebas de consistencia, navegación, visualización, etc.
- **Procesos de traducción de contenido existente a formato de la Web Semántica:** También se engloban todas las herramientas que permiten traducir contenido existente, aumentándolo con anotaciones semánticas. Dada la gran cantidad de contenido existente hoy en día, sobre todo en la WWW. Esta rama de provisión de contenido cobra mucha importancia.

La superación del reto de la disponibilidad de una masa crítica de contenido puede beneficiarse especialmente de métodos y herramientas que permiten anotar o convertir contenido existente de la Web actual, incluida la parte dinámica de la Web profunda para que forme parte del contenido de la Web Semántica. Dado el tamaño del contenido a convertir cabe contemplar especialmente métodos automáticos o semiautomáticos capaces de procesar distintos tipos de contenidos (estáticos, dinámicos, multimedia, servicios Web, etc.) y generar contenido en lenguajes de la pirámide de la Web Semántica.

Para el contenido generado de manera dinámica, generalmente mediante procesos automáticos de publicación desde bases de datos, existen dos opciones de traducción del mismo hacia la Web Semántica:

- **Anotar el resultado de la generación dinámica del contenido**. Se anota el contenido online ya publicado. Esta aproximación se parece a las técnicas utilizadas en la anotación de contenido estático con la ventaja de la presencia de estructuras más definidas en el caso del contenido dinámico.
- **Anotar los datos en su origen.** El contenido dinámico se almacena en bases de datos estructuradas cuyas definiciones y esquemas codifican cierta semántica del mismo. En el modelo Entidad/Relación se modelan conceptos que el diseñador pudo identificar en el dominio, etc. Ya existe software que permite traducciones del contenido de bases de datos en lenguajes de la Web Semántica (Bizer, 03) volcando todo su contenido en documentos entendibles por agentes de software.

Aunque la mayoría de procesos de traducción hacen referencia al contenido textual que ha de ser anotado por procesos para formar parte de la Web Semántica, otra gran parte del contenido de la Web actual son archivos multimedia que incluyen archivos de imágenes, videos, animaciones o sonido. En la actualidad existen dos aproximaciones para su análisis y anotación. La primera de ellas no analiza este contenido en sí, si no que estudia su contexto textual (documentos colaterales) para determinar los metadatos del contenido multimedia incrustado. La segunda opción más completa y más costosa, contempla la utilización de técnicas de análisis apropiadas para cada medio: análisis de imágenes mediante series de Fourier, reconocimiento de voz por patrones, entre otras muchas cosas.

En los últimos años, un nuevo contenido está encontrando su lugar en Internet. A diferencia de los descritos anteriormente este es orientado a ofrecer una funcionalidad, en vez de ofrecer contenido. Se trata de Servicios Web (Web Services). La inclusión de Servicios Web en el paradigma de la Web Semántica bajo el nombre de Servicios Web Semánticos (Semantic Web Services) (Bussier et al, 02) nos acerca un paso más a la meta propuesta de poder delegar tareas a agentes inteligentes. Las aplicaciones o los agentes

inteligentes podrán además de entender el contenido anotado semánticamente, invocar funcionalidades sobre él para cometer las tareas que les han sido encomendadas.

La anotación de servicios Web debe permitir traducir la descripción existente expresada en lenguajes como WSDL (WSDL) y generar una descripción semántica que exprese las funcionalidades del servicio para que este pueda ser usado por agentes de software. Varias iniciativas de los creadores de lenguajes semánticos contemplan extensiones con el propósito de describir los Servicios Web: UMPL (Fensei et al, 98), DAML-S (Ankolekar et al, 02), OWL-S (OWL-S). WSMF (Fensel, Bussier 02). La expresividad de estos lenguajes debe ser suficiente para permitir que agentes software encuentren y entiendan las funcionalidades de los servicios para poder componerlos en funcionalidades complejas útiles para el usuario final.

2.4.5. Disponibilidad de Ontologías

La parte formal del contenido está sujeta por el uso de ontologías, que otorgan la capacidad de comprensión a las aplicaciones o los agentes inteligentes. La disponibilidad, facilidad de gestión y divulgación de éstas es otro de los retos que se plantea. Las ontologías consideradas como repositorios formales de conocimiento, siguen un ciclo de vida [Motta 93] que modela desde su construcción, refinamiento, modificaciones, uso o explotación hasta su retiro. Alrededor de esta representación se sitúa el reto de la disponibilidad de las ontologías, que incluye la necesidad de metodologías de construcción, herramientas que las soporten, métodos de evaluación, comprobación y metodologías de evolución como gestión de cambios y de versiones entre otros. Las ontologías no son formalismos cerrados y están sujetos a procesos evolutivos. Es por eso que metodologías y herramientas que soporten estos procesos se hacen esenciales. El proceso de desarrollo de ontologías (Gomez-Perez, 96) debe tener el apoyo necesario tanto desde el punto de vista metodológico como de herramientas que lo faciliten.

Las ontologías se pueden clasificar teniendo en cuenta diferentes criterios. En la literatura pueden encontrarse muchas posibilidades. Veamos dos de ellas:

a) El alcance de su aplicabilidad:
    - **Ontologías de dominio**. Proporcionan el vocabulario necesario para describir un dominio dado. Incluyen términos relacionados con: los objetos del dominio y sus componentes, un conjunto de verbos o frases que dan nombres a actividades y procesos que tienen lugar en ese dominio, conceptos primitivos que aparecen en teorías, relaciones y formulas que regulan o rigen el dominio además de modelar las particularidades de las realidades de acuerdo a los propósitos de explotación impuestos.

- **Ontologías de Tareas**. Proporcionan el vocabulario para describir términos involucrados en los procesos de resolución de problemas los cuales pueden estar relacionados con tareas similares en el mismo dominio o dominios distintos. Incluyendo nombres, verbos, frases y adjetivos relacionados con la tarea (objetivo, planificación, asignar, clasificar, etc.).
- **Ontologías Generales.** Representan los datos generales que no son específicos de un dominio. Por ejemplo, ontologías sobre el tiempo, el espacio, ontologías de conducta, de casualidad, etc.).
- **Ontologías de nivel superior.** Permiten modelar los niveles más altos de una realidad, ofreciendo conceptos genéricos para la clasificación de términos. Algunas ontologías de estas características son: CyC (Lenat, 95), propuesta de Guarino, etc. Algunas ontologías construidas para propósitos de aplicaciones de procesamiento de lenguaje natural pueden considerarse de nivel superior WordNet (Miller, 95), SUMO (Niles et al, 01), etc.

b) La granularidad de la conceptualización (cantidad y tipo de conceptualización):

- **Terminológicas:** Especifican los términos que son usados para representar conocimiento en el universo de discurso. Suelen ser usadas para unificar vocabulario en un dominio determinado (contenido léxico y no semántico).
- **De información:** Especifican la estructura de almacenamiento de bases de datos. Ofrecen un marco para el almacenamiento estandarizado de información (estructura de los registros de una BD).
- **De modelado del conocimiento:** Especifican conceptualizaciones del conocimiento. Poseen una rica estructura interna y suelen estar ajustadas al uso particular del conocimiento que describen (términos y semántica).

La mayoría de las metodologías de construcción incluye pasos que permiten adaptar y modificar ontologías ya existentes para construir otras más adecuadas a los propósitos del dominio modelado. En los escenarios de modelos de negocio previstos (Forrester, 01) las ontologías de dominios específicos las proveerán los actores interesados en su explotación, normalmente empresas líderes en un sector concreto. El reto se plantea en la disponibilidad de ontologías de propósito general que incluyan términos generales como tiempo, espacio, términos abstractos, etc.

La evolución de ontologías y del contenido ya anotado por ellas es uno de los más importantes elementos que debe contar con apoyo tanto metodológico como de herramientas. Herramientas de gestión de configuración serán las encargadas de controlar las versiones de todas las ontologías y las anotaciones. Al mismo tiempo que las metodologías y las herramientas dan soporte para todo el ciclo de vida de una ontología, se debe asegurar que el lenguaje en el cual estas se expresan no sufra grandes cambio.

Por ejemplo, recientemente un nuevo lenguaje ha diseñado con el propósito de ser el lenguaje genérico para la Web Semántica OWL (Dean et al, 02).

Esto podría tener efectos no previstos si no se estudia con detenimiento la manera que migrar o hacer compatible el contenido existente. La aproximación por capas de lenguaje asegura una cierta compatibilidad sin tener que ejecutar grandes procesos de traducción.

2.4.6. Escalabilidad

Al igual que el problema de sobre-información en la Web actual, la Web Semántica deberá proveer mecanismos que permiten organizar los contenidos para ser fácilmente localizables recuperables por las aplicaciones. Éstas deberán localizar contenido relativo a un dominio o a una ontología en concreto para poder ejecutar las tareas que fueron encomendadas. En la actualidad las personas conocemos las direcciones de portales temáticos o usamos buscadores que nos proporcionan índices de recursos para palabras clave.

La superación del reto de la escalabilidad tiene por objetivo permitir que las aplicaciones construidas sobre la Web Semántica puedan trabajar sobre cantidades crecientes de contenido semánticamente anotados. Para eso es esencialmente permitir que estas aplicaciones sean capaces de localizar, evaluar y explotar el contenido de manera eficiente y fiable. Se exponen aquí dos tipos de arquitectura para organizar índices de contenido semántico junto con algunos ejemplos de lenguajes de recuperación de información sobre repositorios ontológicos.

Existen varias aproximaciones para ofrecer servicios de búsqueda semántica, ya sea orientada para consumo humano o por parte del software.

- ***Uso de arquitecturas distribuidas:*** Software especializado en la indexación de contenido, en una arquitectura de sistema multiagente, capaz de ofrecer enlaces a instancias de conceptos y contenido anotado semánticamente (Esperonto). La presencia de estos buscadores e índices organizados en paradigmas como P2P (Peer to Peer) permitirán la creación de islas de contenidos que agrupen información semánticamente cercana. Las arquitecturas de este tipo son muy escalables permitiendo que la cantidad de contenido presente no sea una barrera en su explotación.
- ***Uso de arquitecturas centralizadas:*** Consiste en servicios centralizados de localización e indexación de contenido, al estilo de buscadores conocidos (Google,

Yahoo) para los humanos o registros de Servicios Web (UDDI[14]), para software. Su escabilidad está puesta entre dicho por las cuotas de cobertura alcanzadas en una Web que no para de crecer.

Así mismo se debe definir el lenguaje de recuperación y la medida que permita calcular la similitud de contenidos para su localización. Los lenguajes de acceso a contenido como RDQL (Seaborne, 01), SeRQL (Broekstra et al, 01), RQL (Karvounarakis et al, 02) permiten formular interrogaciones en ontologías para recuperar instancias de conceptos. En las medidas usadas para definir islas de contenidos o para calcular la precisión de una búsqueda deben contemplarse términos como "similitud semántica" o "relación semántica" definidos en algunas aplicaciones de procesamiento de lenguaje natural (Budanitsky, 01).

2.4.7. Visualización

Otra manera de abordar la sobrecarga de información incluye nuevas técnicas de visualización, que permiten mostrar de forma intuitiva y amigable el contenido.

Técnicas de transformación automática de lenguajes de la Web Semántica (RDFS, DAML, OWL, etc.) hacia escenarios virtuales en 2D o 3D serán la clave de ese reto, como se ilustra en la figura 14.

En el ejemplo se muestra la traducción de una ontología en un modelo en tres dimensiones interactivo, que permite a usuarios navegar por las distintas instancias (Esperonto).

2.4.8. Multilingualidad

Uno de los problemas heredados es la variedad de idiomas usados en los recursos de la Web Online. Actualmente el inglés es la lengua predominante (con casi un 70% de proporción), pero las previsiones apuntan a un aumento en la proporción de otros idiomas. Aunque la Web Semántica no va dirigida a un consumo directo por humanos, sus recursos sirven en definitiva para que los agentes interactúen y trabajen con usuarios finales.

El problema de la Multilingualidad se presenta en varios niveles:

- ***Descripción de los recursos semánticos:*** Tanto las ontologías como las anotaciones están en su mayoría escritas en ingles. Desde el punto de vista del software que las

[14] son las siglas del catálogo de negocios de Internet denominado *Universal Description, Discovery and Integration*. El registro en el catálogo se hace en XML. UDDI es una iniciativa industrial abierta (sufragada por la OASIS) entroncada en el contexto de los servicios Web.

maneja esto carece de importancia ya que las relaciones y primitivas semánticas son estándares formales. El problema está en los posibles manejos de las ontologías por ingenieros en procesos de adaptación, integración, etc.

- ***Explotación de los recursos semánticos.*** Como se ha comentado anteriormente, las ontologías y anotaciones están principalmente manejadas por aplicaciones software independientes del idioma. El problema surge cuando estas aplicaciones tienen que interaccionar con el usuario mediante interfaces. Éstos deben permitir interactuar en alguna lengua conocida por el usuario que no tiene por qué coincidir con la lengua de codificación de los recursos.

Adquisición de recursos semánticos: En este caso el problema surge en posibles procesos automáticos de adquisición de ontologías o anotaciones semánticas a partir de recursos originalmente diseñados para consumo humano. Los recursos de la Web actual son multilingües, con claro predominio del inglés, pero no se puede descartar ninguna lengua. Existe una necesidad de fuentes para la construcción de recursos semánticos, como se ilustra la proporción de idiomas en la figura 14.

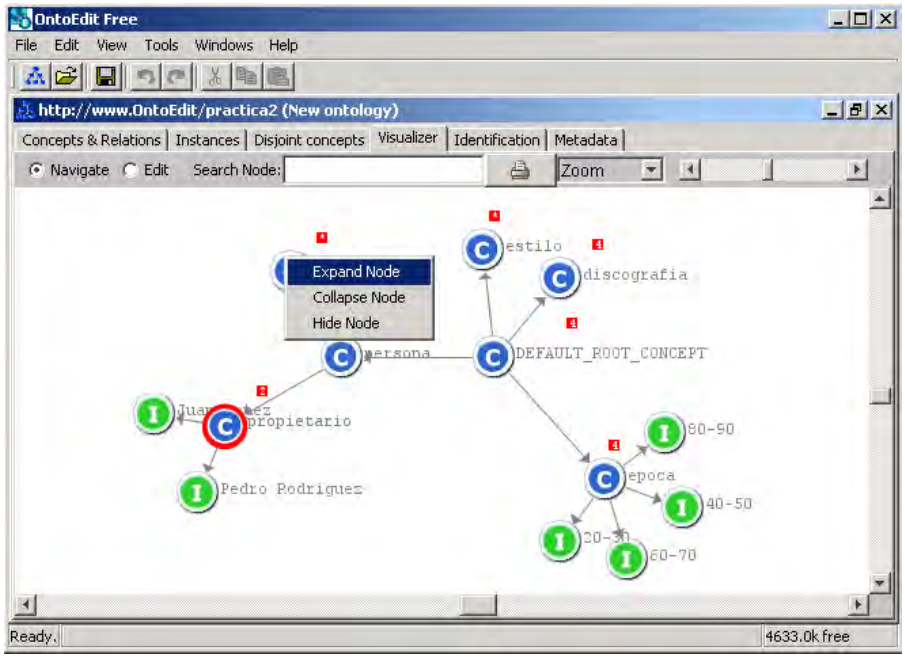


Figura 12 Visualización de una ontología en 2D (Bozak et al, 02)

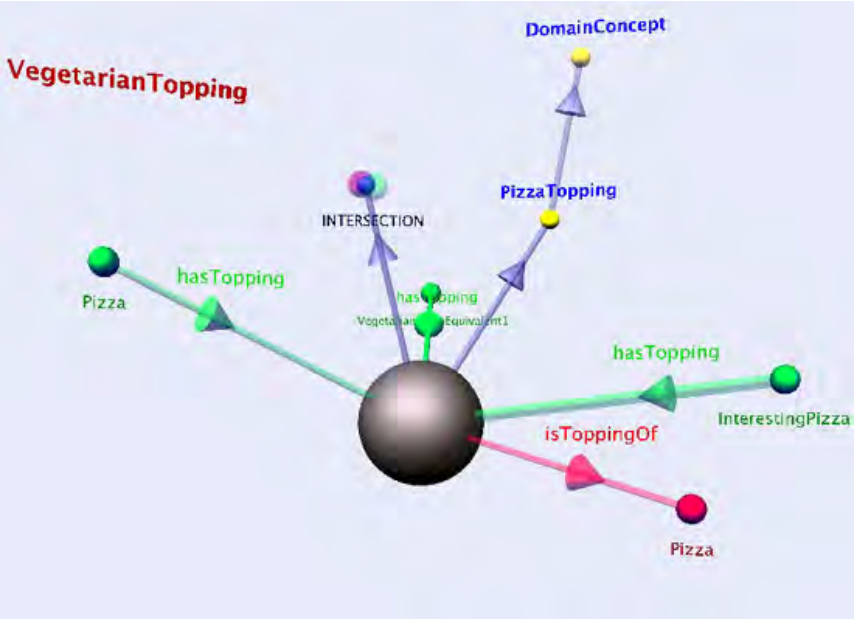


Figura 13 Escenario 3D de instancias de una ontología.

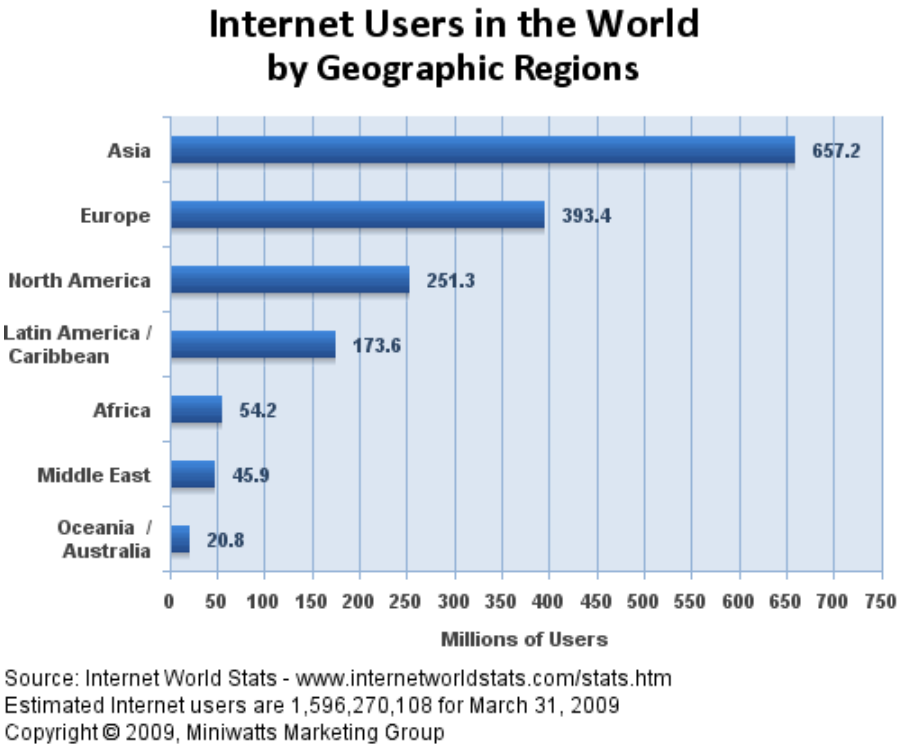


Figura 14 Proporción de idiomas en la WWW en el año 2009 (Internet World Stats)

2.4.9. Aplicaciones

No menos importante que los retos tecnológicos y de formalismos que se plantea el reto de la explotación y uso de la Web Semántica. Haciendo un símil con la Web actual que presenció su auge en cuanto se perfilaron nuevos modelos de negocio, se esbozan algunas posibilidades o visiones sobre los tipos de aplicaciones en la Web Semántica.

La tecnología de la Web Semántica ofrece la posibilidad de construir contenido de manera formal y completa a modelos semánticos consensuados. La existencia de estos modelos permite que las funcionalidades ofrecidas por estos sistemas abarquen, entre otras, las siguientes aplicaciones:

- ***Recuperación de información*** mediante buscadores semánticos: las búsquedas semánticas, al contrario que las tradicionales basadas en palabras clave, trabajan con el significado de las palabras de acuerdo al modelo subyacente asegurando la precisión del 100% en las búsquedas El resultado presentado al usuario pasa a ser la información solicitada en forma de conceptos del modelo, en lugar del los documentos posiblemente relacionados como lo hacen los buscadores presentes.
- ***Publicación de la información de acuerdo al modelo***. La navegación y la presentación de la información se podrá hacer de acuerdo a su contenido de manera que el usuario pueda visualizar los conceptos del modelo y consultar los conceptos relacionados independientemente de los documentos presentes ene l sistema.
- La presencia del modelo permite la incorporación de **interfaces** inteligentes como son los **basados en lenguaje natural**. La posibilidad de formular consultas en un lenguaje cercano al natural asegura la usabilidad del sistema final.
- ***Sistema de inferencia y compleción de información***. En base a los axiomas de los modelos de la Web Semántica es posible validar y aumentar la información mediante sistemas de inferencia automáticos.
- ***Intercambio de información*** a formatos de aplicaciones especificas. La posibilidad de traducir la información a formatos de otras aplicaciones como pueden ser aplicaciones educativas permite aumentar la rentabilidad de la codificación de la misma. Actualmente el gasto de las empresas en hacer compatibles a sistemas heterogéneos supone un 30% del gasto de toda la industria de tecnologías de la información.

A corto plazo, la adopción de ontologías como formalismo de estructuración y anotación de contenidos permitirá el desarrollo de herramientas de búsqueda y recuperación de información precisas. Los sistemas de gestión documental, intranets o documentos

inteligentes podrán ofrecer servicios de consultas cuyos resultados responderán a preguntas formuladas en base a la ontología.

El cuello de botella de la adquisición de contenido en formato adecuado podrá ser resuelto gracias a sistemas semiautomáticos de adquisición sobre dominios restringidos. Estos sistemas basados en tecnologías de recuperación y extracción de información, se ofrecerán en modo ASP (Application Service Provider: aplicación residente en un proveedor y que ofrece sus servicios a los usuarios de manera centralizada), para que los proveedores de contenido puedan obtener una versión semántica del mismo.

A medio plazo se incluirá la capacidad de inferir nuevo conocimiento mediante el uso de axiomas de la ontología o reglas definidas por el usuario. Esto generará un gran valor añadido en las aplicaciones de acceso y búsqueda de información. Los usuarios dispondrán de software configurable que podrá ejecutar tareas complejas en su lugar.

En los comienzos y debido a la falta de infraestructura de seguridad y mecanismos de autentificación estas aplicaciones proliferarán más entornos cerrados como las intranets o redes protegidas.

Los servicios de creación de contenido se incrustarán en aplicaciones de propósito general tales como editores, OCRs[15] y grabadores que dispondrán de una opción que exporte y traduzca su contenido en formato apto para agentes de la Web Semántica.

A largo plazo, las aplicaciones y el contenido poblarán la Web en toda su extensión para ser el lenguaje universal de intercambio de información y funcionalidades, especialmente en el área de B2B, integración de datos y aplicaciones entre empresas (EDI: Enterprise Data Integration, EAI: Enterprise Application Integration). Resuelto el problema de integración se procederá a encargar tareas a los agentes software que serán capaces de configurar o componer procedimientos para acometer los objetivos que les deleguemos como se muestra en la figura 15.

2.5. Masa Crítica: Adquisición Automática

En este trabajo se propone una arquitectura de un sistema de procesamiento de contenido textual de la Web actual a formatos de la Web Semántica. Actualmente existen una serie de aplicaciones clasificadas pre-Web Semántica que hacen uso de software llamado wrappers para obtener contenido cuyo significado puedan procesar. El desarrollo

[15] (Optical character recognition). Tipo de software que se encarga de reconocimiento óptico de caracteres. Se encarga de extraer de una imagen los caracteres de un texto y los guarda en un formato que pueda editarse como texto. Sirve por ejemplo el guardar en forma de texto imágenes escaneadas de un libro sin pasarlo a mano, o sea tipiar carácter por carácter en un editor de texto. Los software son relativamente fiables aunque suelen fallar si las imágenes o las letras no son claras.

de este tipo de software es muy caro tanto del punto de vista de recursos necesarios para su creación como para su mantenimiento. Estos wrappers hacen uso de técnicas de diferentes aéreas de recuperación y extracción de información, tales como el procesamiento de lenguaje natural, visualización de documentos, procesamiento de documentos estructurados, patrones de expresiones, etc. Para localizar información dentro de un documento.

Un Wrapper en el contexto de la ingeniería informática, se describe como un paquete de software de varios componente o recursos que traduce o amplia su modo de operar (interface) para adecuarlo a un propósito especifico. En el contexto de aplicaciones de la Web Semántica, el Wrapper se entiende como el software que permite tratar recursos de información como si fueran recursos de la Web Semántica. Es por eso que la principal funcionalidad de un *Wrapper*, en este ámbito, es extraer y recuperar información de fuentes online y ofrecerlas como instancias de una ontología.

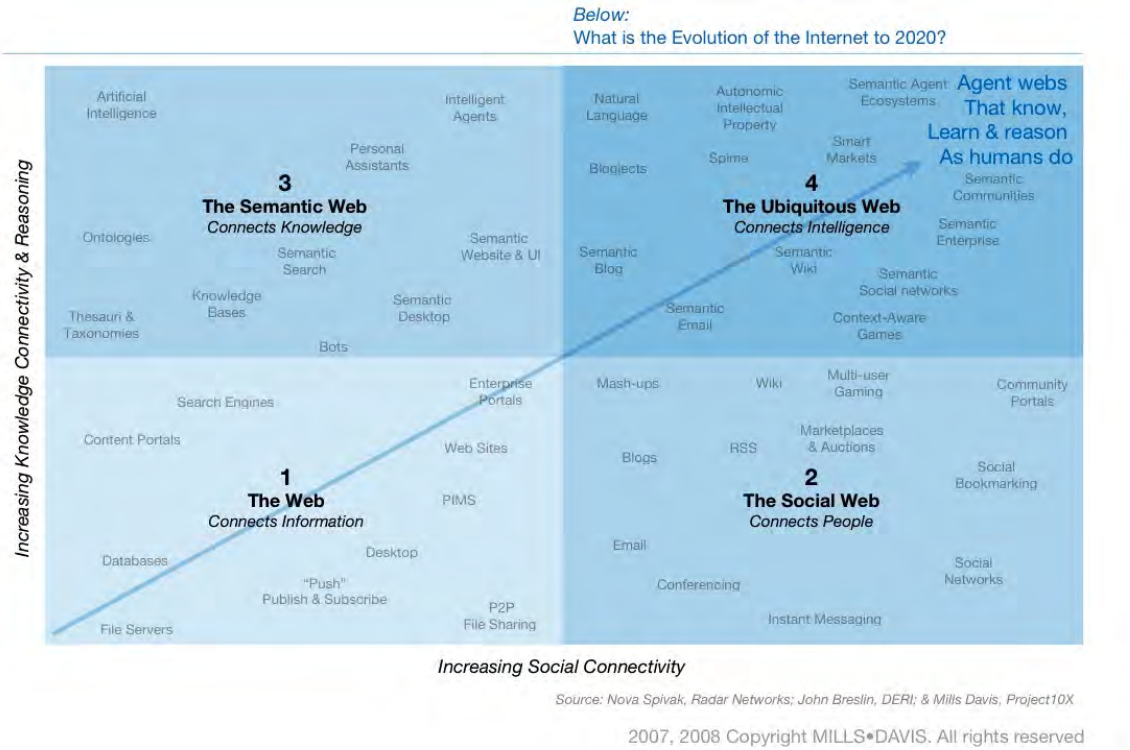


Figura 15 Evolución de las aplicaciones de la Web Semántica (figura extraída de la documentación de la W3C)

Los wrappers hacen uso de diferentes tecnologías para acometer las tareas de producir anotaciones semánticas, desde simples reglas basadas en expresiones regulares, pasando

por heurísticas de formato o aspecto hasta complejos procesos de análisis de lenguaje natural.

El objetivo principal de los wrappers es que permitan una rápida, eficiente y automática extracción de información directamente procesable por programas de software. Su principal problema reside en la dependencia sobre la estabilidad de la fuente. Cuanto menos sofisticada es la técnica de extracción, más sensible es el Wrapper a cambios, aunque mínimos, en la fuente. Esto repercute mucho en su costo total, sobre todo en la fase de mantenimiento. La experiencia nos muestra que su mantenimiento es tanto o más costoso que su propio desarrollo (Cohen, 99).

En la figura 16 se puede contemplar una arquitectura lógica de un Wrapper en el ámbito de la Web Semántica y más concretamente en el ámbito de este trabajo.

Desde un punto de vista estructural en un Wrapper podemos identificar tres posibles entradas y una salida de datos:

- ***Fuentes*** que son capaces de procesar en la Web actual, las cuales están formadas por:
  - Contenido estático construido manualmente por usuarios.
  - Contenido dinámico construido mediante procesos automáticos de publicación.
  - Contenido multimedia: gráficos, ficheros de video o audio.
  - Contenido de las bases de datos accesibles online.
  - Servicios Web y funcionalidades remotas
- ***Ontologías de dominio:*** El cometido del Wrapper es rellenar con datos un esquema de dominio definido mediante una ontología. Este esquema es una información muy valiosa para el proceso de extracción de información ya que sugiere que información se busca en las fuentes.
- ***Información adicional a la ontología de dominio:*** A veces es posible y deseable ampliar la información que proporciona el esquema de dominio con el fin de facilitar la labor de extracción con información previa. De esta manera si el esquema de dominio prevé la existencia de algún tipo de dato básico para un concepto (número, colección de símbolos, etc.), la información adicional puede proporcionar algunas características que faciliten su identificación en las fuentes (expresiones regulares, restricciones semánticas, relaciones con otros datos, etc.).

Como salida deseada de un Wrapper son las funcionalidades que permitan manejar el contenido original como si fuese estructurado de acuerdo al esquema de dominio

deseado. Estas funcionalidades se han denominado APIs (del inglés: Application Programmer Interface).

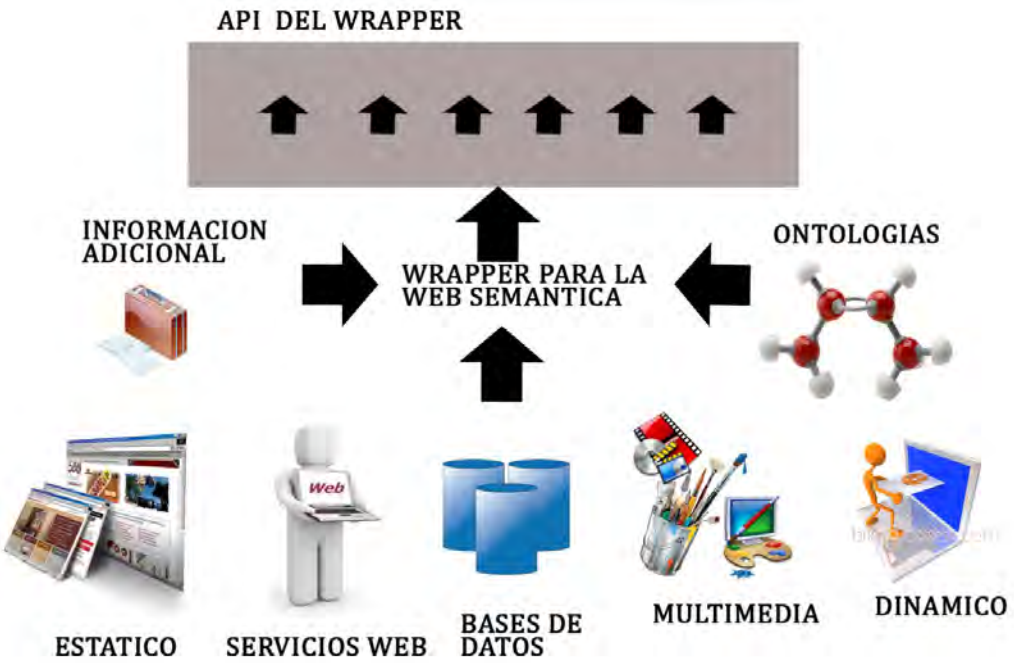


Figura 16 Arquitectura lógica de un Wrapper.

Una de las posibles contribuciones a la solución del reto de la disponibilidad de contenido es establecer marcos de desarrollo de wrappers que permita abaratar los costos de producción. Desde el punto de vista de costos de mantenimiento, será el uso de algunas técnicas de inteligencia artificial que harán que sean más resistentes a cambios en las fuentes que procesan y de esta manera requieran menos mantenimiento. El trabajo esta englobado en estas líneas con el fin de contribuir a superar el reto de disponibilidad de contenido semántico.

2.6. Estado del Arte

En este capítulo se presentan algunas tecnologías y aplicaciones que permiten extraer el conocimiento de documentos online. El estudio se centra en dos parámetros específicos de estos sistemas, que los hacen diferentes a sistemas tradicionales de extracción de información. El primero de ellos es su independencia del dominio. Algunos de los sistemas o aplicaciones resuelven con éxito la tarea de extracción en un dominio concreto (como pueden ser las noticias o las publicaciones científicas) pero son difícilmente adaptables a

otros. El segundo parámetro de interés es el formato final de los datos. Algunas aplicaciones localizan la información relevante dentro del documento procesado y allí acaba su cometido, mientras que otras aplicaciones los insertan en una base de datos o en una ontología.

Existe otra posible clasificación de aproximaciones al problema de extracción de información respecto al conocimiento previo o modelo semántico que se utiliza. Existen líneas de argumentación que, en sintonía con la visión de la Web Semántica, creen en la creación de un repositorio universal de conocimiento que con un número reducido y controlado de estándares puede satisfacer los requisitos de las aplicaciones y agentes inteligentes. Se engloban todas las líneas y aplicaciones que potencian el carácter compartido y de consenso de las ontologías y trabajan en la dirección de generación y adquisición de contenido de acuerdo a ellas. Aquí se incluyen grupos o empresas promotoras de los lenguajes semánticos (RDF, OWL, DAML, etc.), grupos o empresas que aportan herramientas de gestión de ontologías o de su explotación y uso para usuarios finales. El concepto de semántica u ontología no suele estar explicito en muchos de los sistemas estudiados.

En la otra línea de argumentación cuya premisa parte que cada aplicación tiene necesidades particulares especiales que no podrían satisfacerse con una ontología de propósito general. Sería necesario acotar su dominio hasta llegar al nivel de aplicación con lo cual se rompe la condición de modelo compartido y común. Por ello proponen que cada aplicación utilice módulos de extracción de información que trabajen con un modelo propio y específico. El objetivo de esta línea de argumentación se centra en la provisión de métodos rápidos, eficientes y baratos de recuperación de información. En esta línea se agrupan algunos investigadores del área de aprendizaje automático (Kushmerick, 97) y empresas de buscadores tales como Google (Google) o el nuevo buscador Bing de Microsoft (Microsoft).

Una de las principales desventajas de la **tecnología de extracción de información** es la portabilidad de sistemas existentes a nuevos dominios e idiomas. En general, la portabilidad implica reajustar manualmente el conocimiento lingüístico que depende del dominio, por ejemplo: diccionarios, gramáticas, patrones de extracción, entre otros. Desde finales de los 90's a la fecha las investigaciones se enfocan en el uso de métodos empíricos para automatizar y reducir el alto costo de la portabilidad, los esfuerzos se concentran principalmente en el uso de técnicas de **aprendizaje automático** para adquirir de forma automática los **patrones de extracción** útiles para tratar con un lenguaje y dominio particular, y que además es una de las tareas más costosas en el desarrollo del sistema. Las aproximaciones más comúnmente aplicadas son las que utilizan alguna forma

de aprendizaje supervisado. En la tabla 2 se presenta una clasificación de los tipos de **patrones de extracción** adquiridos por utilizar **aprendizaje automático**, cabe destacar que los sistemas incluidos en la tabla son sólo una parte de los sistemas actuales, pero que resultan útiles para ejemplificar el estado actual de la tecnología. A continuación se describen brevemente cada una de las categorías en la tabla.

Aprendizaje de Reglas .Esta aproximación es la más comúnmente usada, la cual se basa en un proceso de aprendizaje inductivo del tipo simbólico (programación lógica inductiva). Algunas de estas aproximaciones trabajan en el contexto de un aprendizaje proposicional, mientras que otras lo hacen en el contexto de aprendizaje relacional. A continuación se detalla cada una.

El aprendizaje proposicional se basa en representar los ejemplos de un concepto en términos de la lógica de proposiciones, algunos sistemas desarrollados bajo este paradigma son AutoSlog y CRYSTAL. El objetivo ha sido aprender los **patrones de extracción** solamente de ejemplos positivos.

La primera columna indica el nombre del sistema. En la columna textos, *NE* representa textos no estructurados, *SE* textos semiestructurados y *E* textos estructurados.

En esta técnica, los ejemplos de un concepto son representados generalmente como un conjunto de atributos. Mientras que los valores a ser extraídos son los núcleos de frases sintácticas que ocurren en los documentos de entrenamiento. Por ejemplo, una de las primeras aproximaciones fue AutoSlog que genera **patrones de extracción** llamados nodos de concepto, donde un nodo es definido como una palabra disparador que puede activar el nodo de concepto que realiza la extracción, además un conjunto de restricciones envuelven al disparador y a los valores extraídos (ver figura 3). En resumen, sistemas que siguen esta aproximación son útiles para extraer información a partir de textos no estructurados (*NE*), sin embargo la extracción del fragmento de texto relevante no es exacta (a pesar de que se detecta la información correcta ésta no se extrae íntegramente).

Sistema	Clase	Modelo	Textos	Fragmento Exacto
AutoSlog	Aprendizaje de Reglas	Aprendizaje Proporcional	NE	No
Crystal				
SRV		Aprendizaje Relacional	SE	Si
RAPIER				
WHISK			NE,SE,E	
Textractor	Separadores Lineales	Clasificadores		
SNoW-IE				
CoA			NE, SE	
LHMM	Aprendizaje Estadístico	Modelo Oculto de Markov	SI	
HMM				
TC				

Tabla 2 Patrones de extracción adquiridos con aprendizaje automático

Figura 17 Un nodo de concepto inducido por AutoSlog

En el aprendizaje relacional los ejemplos de un concepto son representados en términos de lógica de predicados, los patrones de extracción se representan como atributos y relaciones entre elementos textuales. La entrada de este sistema es un conjunto de características relacionadas a los tokens que representan ejemplos positivos y negativos de los patrones, la salida es un conjunto de reglas que ayudan a contestar la pregunta de clasificación (en la figura 18 se presenta un ejemplo de los patrones inducidos por SRV). En general, los sistemas que siguen esta aproximación son útiles para extraer información a partir de textos no estructurados (*NE*), semiestructurados (*SE*) y estructurados (*E*), además de que se tiene una exacta extracción de los fragmentos de texto relevantes.

Separadores Lineales . Esta técnica surgió por la necesidad de crear de una forma más rápida los patrones de extracción requeridos. Es decir, en comparación con los sistemas de aprendizaje de reglas donde en algunos casos se requiere la intervención del experto en el proceso de aprendizaje. En esta metodología se emplean algoritmos de aprendizaje automático para tratar de inducir el conocimiento necesario a partir de un conjunto de documentos de entrenamiento, donde cada entidad a ser extraída está etiquetada. Además, en esta técnica también se incluyen entidades que representan ejemplos negativos a la extracción. La hipótesis que se toma es que las entidades relevantes y las irrelevantes son linealmente separables, el problema de extracción se transforma en un problema de clasificación (ver figura 19).

```
interlocutor.-                          // F es un interlocutor si
  cierto(?A,[], token,*desconocido*)    // F contiene un token (A)
  cada(capitalizar, true)               // cada A en F es capitalizado
  longitud(=,2)                         // F contiene exactamente dos A's
  cierto(?B,[], token,*desconocido*)    // F contiene otro token (B)
  cierto(?B,[precede], token. ";")      // B es precedido por dos puntos
  cierto(?A,[sucede],doble_char,false)  // A no es sucedido por 2 caracteres
  cada(cuadruple_char, false)           // cada A en F no es de 4 caracteres
  cierto(?B,[2_precede], token, "quien") // 2 tokens antes de B está la palabra
                                        // "Quién"
```

Figura 18 Regla inducida por SRV

Generalmente, los **patrones de extracción** son inducidos por generalizar los contextos de las diferentes entidades, por lo tanto los patrones obtenidos están implícitos en el clasificador. Ejemplos de esta aproximación son: Textractor, SNoW-IE y CoA. En resumen, esta técnica se ha empleado principalmente para tratar con textos semi-estructurados (*SE*) y en algunos casos sencillos de textos no estructurados (*NE*), además de que se tiene una exacta extracción de los fragmentos de texto relevantes.

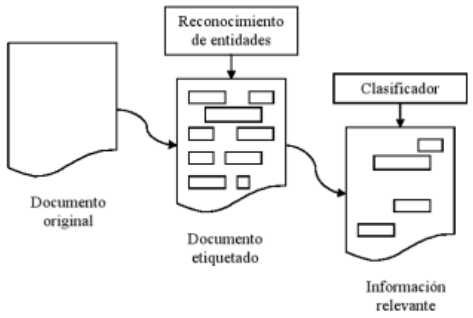


Figura 19 Extracción de información como clasificación de entidades

Aprendizaje Estadístico. Los modelos ocultos de Markov (HMMs, por sus siglas en inglés) son la base de esta aproximación, en esta estructura se representa el conocimiento necesario para extraer los fragmentos relevantes de los textos (los **patrones de extracción** son representados por HMMs). Ejemplos de esta aproximación son presentados en LHMM, HMM y TC. Aquí, generalmente los nodos representan tokens o elementos característicos de éstos, y los enlaces representan sus relaciones, además cada enlace tiene asociada una probabilidad de ocurrencia obtenida de los datos de entrenamiento (ver figura 20 para un ejemplo donde se extrae información desde tarjetas de presentación para llenar un directorio de contactos). En resumen, la relevancia del método es que aprovecha la estructura intrínseca de algunos textos, por lo tanto es adecuado para textos semiestructurados (*SE*), además de que los fragmentos de texto obtenidos son exactos.

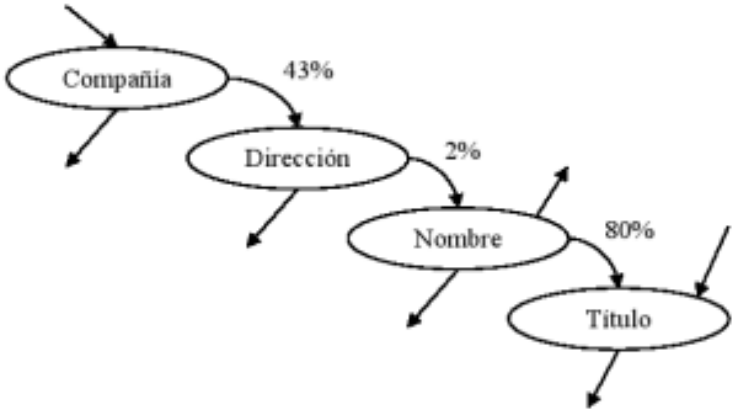


Figura 20 Parte de la estructura introducida con HMMs

2.6.1. Áreas de Investigación en la Extracción de Información

Son varias las áreas que incluyen entre sus líneas de investigación la construcción de software que procesa documentos con el fin de entender su contenido. Entre las principales se incluyen la Recuperación de información (Information Retrieval) y Extracción de información (Information Extraction) que principalmente hacen uso de técnicas de procesamiento de lenguaje natural, métodos estadísticos, procesamiento de cadenas léxicas[16], aprendizaje automático para localizar el contenido útil en fuentes digitales y procesarlo para algún propósito concreto. En muchos casos los sistemas no se limitan a una sola técnica si no que combinan varias de ellas.

2.6.2. Área de Recuperación de Información

Existen muchas definiciones de los que es la recuperación de información. En este trabajo se considera que esta área tiene como objetivo la búsqueda e indexación de documentos útiles dentro de una colección. Desde el punto de vista del usuario, un sistema de recuperación de información atiende una petición de búsqueda y mediante diversas técnicas extrae de una colección los documentos que considera relevantes. Mientras que las interfaces que permiten formular peticiones de búsqueda normalmente solicitan palabras clave, las técnicas más usadas para determinar la relevancia de un documento varían mucho. Las más sencillas buscan cadenas de caracteres idénticas a las dadas por el usuario, técnicas más avanzadas pueden ordenar los documentos por resultado según su popularidad o pueden representar los documentos como un espacio vectorial donde intentan encontrar el documento más relevante a un vector de palabras de búsqueda dado (el modelo vectorial está descrito más adelante).

Cuando los documentos se proveen a un usuario que formula su petición mediante palabras clave y la colección es Internet a estas aplicaciones se les denomina buscadores (Search engines). Desde los años noventa, cuando la WWW se volvió un repositorio universal de documentos ampliamente usado, la recuperación de información está siendo clave para el aprovechamiento de los recursos incorporando técnicas de modelado de documentos, su clasificación y filtrado entre otros para garantizar su buen aprovechamiento (Kobayashi, 00).

Algunos buscadores además ofrecen jerarquías que clasifican los documentos. Este es el caso de buscadores, también llamados directorios como Yahoo! (Yahoo!), Looksmart

[16] Una cadena léxica es una secuencia de palabras, semánticamente relacionadas que aparecen en un texto y que pueden ser adyacentes o encontrarse dispersas a lo largo de un documento. Para encontrar dichas cadenas en un texto genérico es necesario utilizar recursos como WordNet que proporcionan la información necesaria sobre las posibles relaciones entre las distintas palabras.

(Looksmart), etc., que contratan expertos para clasificar documentos nuevos de la red dentro de una jerarquía predefinida de manera manual. Aunque es el método más preciso y más fiable no es escalable a grandes cantidades de documentos que se hospedan en la WWW.

Al final de la década de los noventa se construyeron piezas de software llamados agentes, (crawlers, ants, bots o spiders) que permitían recorrer los documentos de la Web y mediante técnicas automáticas de clasificación e indexación construían una bases de datos donde el usuario puede consultar usando palabras clave. Este es el caso de AltaVista (AltaVista), Lycos (Lycos), Google (Google), etc. Este tipo de software tiene en cuenta tanto el contenido de las fuentes Web como la estructura hipertexto que forman las distintas páginas enlazadas. Existen algunas iniciativas que pretenden aumentar la precisión de las búsquedas añadiendo meta-datos al contenido. El consorcio de la WWW (W3C) ha propuesto un conjunto de meta-datos para páginas Web que describan su contenido y de esta manera faciliten la tarea a los buscadores automáticos. Estos meta-datos se introducen con un conjunto de etiquetas semánticas especiales definidas en el Dublin Core Metadata (Dublin Core) que predefinen algunos atributos de las fuentes como autor, titulo, fecha de creación, idioma, etc.

Ejemplo de etiquetas semánticas de Dublin Core:

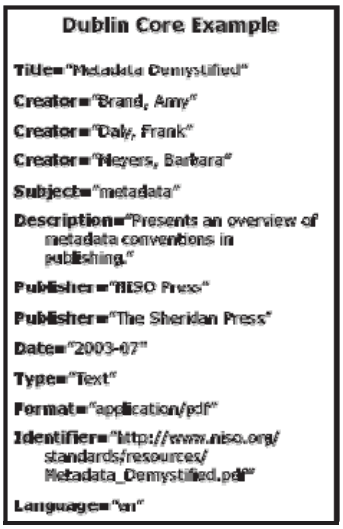


Figura 21 Ejemplo de etiqueta Dublin Core

Aunque los buscadores son aplicaciones de gran aceptación hoy en día presentan algunos problemas:

- ***Calidad de resultados:*** Los buscadores ofrecen como respuesta una lista de documentos que pueden contener la respuesta a la pregunta que intenta formular

el usuario. Con frecuencia las listas de documentos recuperados sobrepasan varios miles de elementos y depende de una buena ordenación del buscador que el resultado buscado se encuentre entre las 10 primeras. Así mismo, al ser un sistema de recuperación de información, nos obliga a procesar los documentos en busca del resultado de manera manual.

- ***Invasores:*** Los agentes entran y salen de los sitios Web como usuarios humanos y generan tráfico ficticio de la red que no corresponde a usuarios reales. Se han propuesto algunos estándares para excluirlos del tráfico en un sitio Web (Koster, 94).
- ***Sobrecarga en el tráfico de la red:*** Dada su actividad generan mucho tráfico en detrimento de consultas de los usuarios humanos.
- ***Sincronización del contenido y los índices:*** La cadencia de refresco que pueden alcanzar los agentes es insuficiente en casos donde el contenido cambia con mucha frecuencia (bolsa, noticias, transporte, etc.).

2.6.2.1. Área de extracción de Información

Entendemos la extracción de información como un proceso más sofisticado que el anteriormente descrito. Mientras que la recuperación ofrece como resultado documentos enteros clasificados como relevantes para la pregunta hecha, en la extracción se presenta solamente la parte de ese documento que satisface la pregunta del usuario. Es necesario aplicar técnicas avanzadas tales como procesamientos de lenguaje natural o estadística para poder identificar la parte del documento que responde a la respuesta de una pregunta dada.

Los textos de la colección se analizan para comprender su contenido y poder ofrecer partes de ellos como respuesta al usuario. Para la mejora de la eficiencia de las técnicas se trabaja sobre dominios acotados lo cual permite reducir el grado de ambigüedad presente en este tipo de análisis. Cuánto más cerrado y acotado es un dominio mejor resultado dan estas técnicas. Dependiendo de la técnica aplicada los modelos de dominios se vuelven explícitos o quedan implícitos en las heurísticas o algoritmos.

En aplicaciones con técnicas estadísticas (modelos de Markov ocultos, autómatas probabilísticos, etc.) los modelos determinan la probabilidad de asignación de significados a palabras o secuencias de ellas, mientras que en aplicaciones de técnicas más simbólicas (por ejemplo: analizadores basados en reglas), el modelo se obtiene en bases de conocimiento con formalismos de reglas, tablas o estructuras semánticas complejas (redes semánticas, ontológicas, etc.).

2.6.2.2. Descripción de Fuentes Disponibles

La aplicación de distintas tecnologías existentes para la extracción de información y el grado de éxito que estas alcanzan depende de muchos parámetros como: tipo de dominio abarcado, idioma, grado de detalles necesario en la extracción, grado de estructuras presentes en las fuentes entre otros. En este apartado se estudia más en detalle el último de ellos que hace referencia a las posibles estructuras presentes en los documentos originales y en qué medida esto influye en la aplicación de distintas tecnologías para la extracción de información.

Como se verá en el sistema propuesto la presencia de estructuras puede tener influencia en el éxito de técnicas aplicadas y puede ser determinante en su selección. Acotando el corpus a documentos textuales de la Web actual podemos distinguir varios tipos de estructuras presentes:

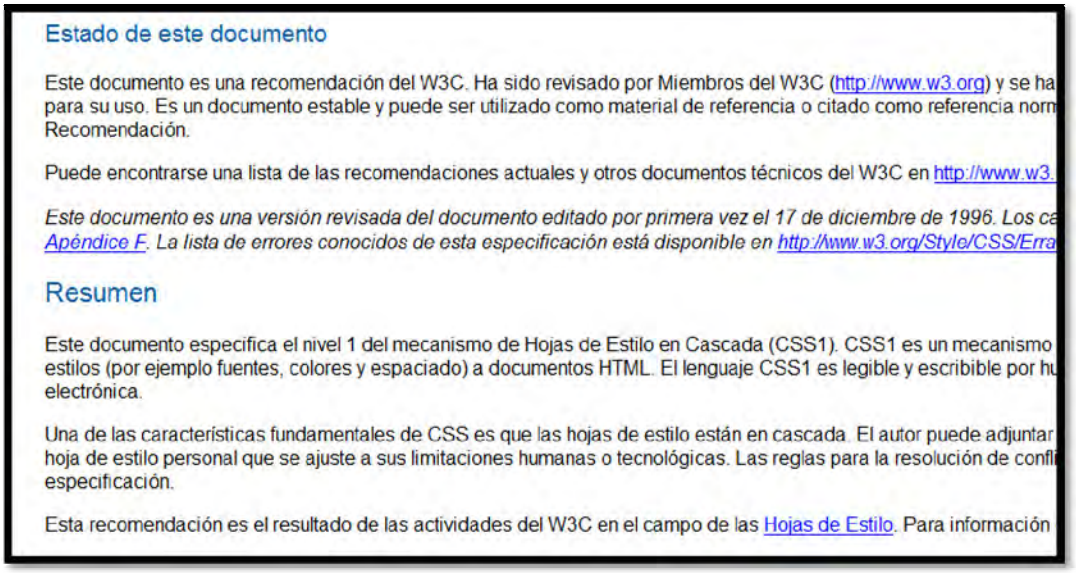
- ***Estructuras internas en la codificación:*** La mayoría de los documentos online en la Web se baja en lenguajes como el HTML (HyperText Markup Language) que siguen una estructura de árbol con enlaces entre documentos y está definida por el consorcio W3C. La presencia de estructuras predefinidas o repetitivas puede elevar el grado de éxito de sistemas de extracción de conocimiento.
- ***Estructuras en el formato del contenido***. A nivel léxico, es decir como cadenas de caracteres es posible identificar estructuras conocidas y aprovecharlo en la extracción. Son muchas las aplicaciones que mediante expresiones regulares, ya sean definidas manualmente o inferidas mediante procesos de inducción, son capaces de identificar y procesar contenido: desde simples direcciones correo electrónico hasta complejas ofertas de trabajo.
- ***Estructuras a nivel visual del contenido.*** El hecho de que el contenido actual de la web vaya dirigido a usuarios humanos hace que el aspecto visual de los documentos sea también portador de significado de los datos contenidos. Estructuras tabulares, relaciones visuales entre distintas piezas de información son relevantes en la extracción e identificación de la información. Estas estructuras guardan cierta relación con las estructuras internas en la codificación aunque a menudo sucede sobre todo en documentos HTML que una misma estructura interna puede producir variaciones en el aspecto visual, según el visor que se use. Así como varias estructuras internas pueden dar lugar a idénticas estructuras visuales.
- ***Estructuras en formas lingüísticas.*** A menudo los documentos de un dominio contienen estructuras propias o típicas que se repiten o cobran un sentido específico. El conocimiento de la presencia y del significado de estas estructuras

permite aumentar considerablemente el poder de las aplicaciones de extracción de información.

2.6.2.3. Contenido no estructurado

Los documentos clasificados como no estructurados, suelen ser páginas escritas directamente por personas en el lenguaje HTML, usando a lo sumo un editor avanzado. La World Wide Web en su origen solamente contaba con este tipo de documentos que albergan las páginas personales de científicos y personal académico. Su elaboración suele ser costosa y el autor no se ve obligado a mantener una estructura determinada en la colaboración del contenido más que en sus propios criterios estéticos.

Este tipo de documentos incluye menos estructuras de colocación de información tales como tablas o enumeraciones. Es frecuente que contenga frases completas en lenguaje natural, en contraposición a meras etiquetas explicativas de los datos, como se muestra en la figura 22.


Estado de este documento

Este documento es una recomendación del W3C. Ha sido revisado por Miembros del W3C (http://www.w3.org) y se ha
para su uso. Es un documento estable y puede ser utilizado como material de referencia o citado como referencia norm
Recomendación.

Puede encontrarse una lista de las recomendaciones actuales y otros documentos técnicos del W3C en http://www.w3.

Este documento es una versión revisada del documento editado por primera vez el 17 de diciembre de 1996. Los ca
Apéndice F. La lista de errores conocidos de esta especificación está disponible en http://www.w3.org/Style/CSS/Erra

Resumen

Este documento especifica el nivel 1 del mecanismo de Hojas de Estilo en Cascada (CSS1). CSS1 es un mecanismo
estilos (por ejemplo fuentes, colores y espaciado) a documentos HTML. El lenguaje CSS1 es legible y escribible por hu
electrónica.

Una de las características fundamentales de CSS es que las hojas de estilo están en cascada. El autor puede adjuntar
hoja de estilo personal que se ajuste a sus limitaciones humanas o tecnológicas. Las reglas para la resolución de confli
especificación.

Esta recomendación es el resultado de las actividades del W3C en el campo de las Hojas de Estilo. Para información

Figura 22 Ejemplo de documento no estructurado

2.6.2.4. Contenido estructurado

La aparición de recursos Web creados dinámicamente desde bases de datos existentes [Lawrence, Giles 99] ha multiplicado por 500 en número de páginas existentes. Tanto los buscadores (Google, AltaVista, Lycos, etc.) como portales de Internet son soluciones temporales y no escalables que pueden paliar este problema hoy en día. De acuerdo con algunas otras publicaciones (Sahuguet 00) más del 80% de la información publicada en la

WWW está generada de manera dinámica a partir de bases de datos. La publicación de manera dinámica permite que el contenido hasta hoy almacenado en sistemas de almacenamiento (bases de datos relacionales, sistemas de gestión de documentos, bases de datos XML, etc.) sea publicado mediante procedimientos automáticos de generación de contenido HTML.

La utilización de estos procedimientos de publicación automática con lleva que las paginas generadas sigan un patrón estructural fijo con el fin de ser rellenado con los datos almacenados. La colocación de los datos en distintos lugares de los documentos HTML está programada en estos sistemas de publicación. La estructura interna de la base de datos (por ejemplo su diagrama Entidad/Relación) que alimenta los documentos suele ser no conocida, aunque el conocimiento previo del esquema de la base de datos subyacente ayudaría a mejorar la precisión del proceso de extracción automática. De esta manera los esquemas XML, definiciones de bases de datos, estructuras de gestores documentales son datos valiosos adicionales para los wrappers, como se puede observar en la figura 23.

Figura 23 Ejemplo de documento altamente estructurado

2.6.3. Tecnologías Usadas

El desarrollo de las ciencias de la información nos lleva a contemplar aplicaciones cada vez más sofisticadas en las áreas de tratamiento de información donde se incluyen aplicaciones como buscadores, sistemas de ayuda a decisión, interfaces avanzados, etc.

Desde el punto de vista de marketing a veces se les denomina 'inteligentes'. Estas aplicaciones suelen usar formalizaciones explicitas del dominio sobre cual trabajan. Las usuales bases de datos que acompañan el software tradicional se convierten en bases de conocimiento. Este conocimiento puede contemplar modelos de usuario, procesos de inferencia, datos complejos o incompletos. Todo esto lo otorga a la aplicación funcionalidades avanzadas que antes eran muy costosas de alcanzar.

Gracias a la estandarización de algunos formatos el desarrollo de aplicaciones se abarató y los servicios que ofrecen se han extendido a costa de tener formalizado y almacenado el conocimiento. El problema se trasladó a la obtención de éste. En la comunidad de sistemas basados en el conocimiento que comenzó a construir este tipo de aplicaciones denomino a este problema: el cuello de botella de la adquisición del conocimiento (Knowledge Acquisition Bottleneck).

El grado de automatización de las herramientas de adquisición es muy variable. Los editores, que son el extremo sin automatización de esta categorización, pueden ir incorporando automatismos hasta llegar al punto de interaccionar con el usuario solamente para la resolución de ambigüedades. En el otro extremo están los Wrappers, herramientas plenamente automáticas que no precisan, aunque no excluyen, una interacción con el usuario.

Figura 24 Generadores de contenido de la Web Semántica según el grado de automatización (Knowledge Parser[17])

[17] Knowledge Parser (KP) es un framework software que permite la extracción automática de datos de fuentes online almacenándolos en un almacenamiento local estructurado. La extracción automática y robusta que KP lleva a cabo está orientada hacia negocios que necesitan datos estructurados originados en fuentes online distribuidas y que no quieren invertir el tiempo y/o el dinero necesarios para construir software de extracción ad-hoc o enfrentarse a cambios frecuentes en los datos origen.

La tarea de extracción de información para propósitos de la Web Semántica se puede dividir en dos procesos de alto nivel:

- ***La adquisición del esquema modelo (ontology learning):*** Está es la tarea que tiene por objetivo la construcción de una ontología como esquema, sin datos particulares. El objetivo es encontrar generalizaciones de conceptos, atributos y relaciones entre los conceptos. Dada la naturaleza de la Web Semántica, esta tarea no es muy frecuente y solo está reservada a unos pocos actores (empresas líderes de cada sector, organismos de estandarización, etc.). En este trabajo no se trata la automatización de esta parte de adquisición de contenido.
- ***La adquisición de los datos o instancias de un modelo (ontology population)***. Una vez definido el esquema el reto de la provisión de contenido consiste en abaratar los procesos de relleno del mismo. La tarea de extracción de datos se puede dividir en los siguientes pasos:
    - ***Extracción de instancias:*** Se trata de extraer referencias de nuevas instancias de la ontología de dominio (nuevas personas, entidades, productos, países, etc.). Esta es la tarea más propicia de ser automatizada, y de hecho muchas herramientas incluyen sistemas de sugerencia de posibles nuevas instancias.
    - ***Extracción de valores de atributos:*** Este problema a primera vista sencillo tiene por objetivo la identificación y extracción de valores de atributos de las instancias. La identificación de los atributos no suele ser enumerada de cadenas, símbolos, etc.). El problema suele sugerir a la hora de designar la instancia a la que pertenecen. Para ello se suelen usar técnicas que calcula la distancia física o semántica dentro de la fuente entre el valor encontrado y las instancias colindantes.
    - ***Extracción de relaciones entre instancias.*** Sin duda el paso más difícil de la adquisición de contenido semántico. Mientras que para los pasos de identificación de instancia y valores suele existir una relación directa entre la fuente y el elemento extraído (es decir: cada nueva instancia o valor de atributo es el resultado de haber encontrado algún elemento en la fuente), en el caso de las relaciones no suele ser tan obvio. Desde el punto de vista del área de procesamiento de lenguaje natural la tarea de detección de relaciones entre entidades o conceptos también es compleja. Se trata de detectar que hechos o instancias del modelo tienen reflejada alguna relación entre ellos. No es usual que exista una correspondencia entre cadenas de caracteres dentro del documento fuente y estas relaciones.

Muchas veces están implícitas o expresadas por varios sintagmas[18] complejos. Pocas herramientas existentes incluyen algún tipo de automatismo para estos casos. Existen líneas de investigación que abordan esta tarea con tecnología de aprendizaje automático (Ciravegna, 01) o técnicas de detección de patrones (Popov et al, 03).

A continuación se hace un repaso de algunas técnicas tanto automáticas como manuales de extracción de datos.

2.6.3.1. Codificación Manual

Para aplicaciones con bases de conocimientos pequeñas o para aplicaciones de dominios académicos o de propósitos demostrativos se suele realizar la tarea de adquisición de conocimiento de manera manual. El desarrollador codifica el conocimiento en el formalismo diseñado para tal propósito en un editor (generalmente de texto plano, XML, etc.) sin más ayuda que las fuentes que proporcionan el conocimiento (documentación, expertos, otras bases de conocimiento, etc.).

En un proyecto llamado HALO (HALO) que tiene por objetivo construir un sistema de búsqueda de respuestas (Questions Answering System) para el dominio de la asignatura de química de educación secundaria de los EEUU, se ha estimado que el costo de la codificación de una página de documentación asciende a 10,000 dólares (HALO final report).

El costo de la codificación depende de la complejidad y del propósito del modelo que se está construyendo. La tarea más común suele ser la codificación de conocimiento correspondiente a instancias de conceptos. Un caso de este proceso es la detección de entidades mediante búsquedas de nombres propios en las fuentes (nombres de personas, elementos químicos, cargos políticos, etc.). Muchas herramientas o editores para la codificación de bases de conocimiento incluyen funcionalidades de localización de nombres propio o cadenas específicas de caracteres que se pueden corresponder con instancias del modelo subyacente.

Una vez construido un modelo con instancias y relaciones entre ellos es posible codificar también reglas o procesos de inferencia que rigen el modelo. En el caso del proyecto

[18] es un tipo de constituyente sintáctico formado por un grupo de palabras que forman otros sub-constituyentes, al menos uno de los cuales es un núcleo sintáctico. Las propiedades combinatorias de un sintagma se derivan de las propiedades de su núcleo sintáctico, este hecho se parafrasea diciendo que "un sintagma se caracteriza por ser la proyección máxima de un núcleo". Por su parte el **núcleo sintáctico** es la palabra que da sus características básicas a un sintagma y es por tanto el constituyente más importante que se encuentra en su interior. Su estructura fundamental es recogida en la llamada teoría de la X' (X-barra)

HALO mencionado anteriormente se debían modelar procesos mentales que proporcionan soluciones a preguntas de examen del dominio de la química. Esta es una de las tareas más costosas y difícil de automatizar en cuanto a codificación de bases de conocimiento.

2.6.3.2. Codificación por formato

En algunas fuentes, sobre todo las estructuradas, existen patrones de formato que permitan asignarle cierta semántica a los datos encontrados. Estos patrones suelen estar formados por elementos de presentación (fuentes subrayadas, tipos de letra, etc.), símbolos gráficos (símbolos de moneda, sobre de correos, etc.) o simplemente por la colocación de los caracteres en estructuras (listas, tablas, etc.). Los extractores de información automáticos hacen uso de estas propiedades y extraen la información de acuerdo al formato que tienen los datos. La dependencia que presenta la semántica de los datos con el formato o aspecto de su presentación es un hecho muy común en los documentos Web, y nos permiten a nosotros los humanos comprender el contenido de manera mucho más eficiente sin necesidad de explicaciones detalladas.

A continuación se detallan tres formas de codificación por el formato:

Codificación por caracteres. Si se considera una fuente textual como una cadena de caracteres se pueden encontrar patrones que se repiten y aportan significado a los datos encontrados. De esta manera podemos suponer con cierta probabilidad de acierto, dependiendo del dominio, que ciertas secuencias de caracteres corresponden a datos buscados: cantidades monetarias, fechas, direcciones de correo electrónico, etc. Es un método muy apropiado para la detección de instancias de conceptos con una particular presentación. En algunos casos este método es suficiente para la detección de valores de atributos de instancias e incluso para la detección de relaciones semánticas entre distintas instancias. Esto sucede si dicha información está codificada enteramente por el formato de los datos y el formato de los datos está unívocamente determinado por los caracteres de la fuente. Un claro ejemplo es un documento CSV (comma separated value), valores separados por comas que representan una tabla con dimensiones conocidas y con significado claro para cada celda. Normalmente, para detección de relaciones semánticas complejas o para asignar valores de atributos a instancias ya existentes debe analizarse más en profundidad el significado del contenido.

El mecanismo más usado para la detección de cadenas que cumplan condiciones deseadas son las expresiones regulares. Una expresión regular permite describir de forma abreviada con conjunto de cadenas con cierta propiedad. Existen variaciones sobre la notación de las expresiones regulares, según la implementación del software de detección que se use.

Ejemplo: Una posible expresión regular para detección de cadenas de correos electrónico.

$$([a-z]|[0-9])+@([a-z]|[0-9])+(.[a-z]|[0-9])$$

Un ejemplo, a veces doloroso, son los programas que buscan y coleccionan direcciones de correos electrónicos para propósitos de publicidad no deseada (spam).

La construcción y depuración de expresiones regulares es una tarea compleja y tediosa. Con el objetivo de reducir este esfuerzo se han utilizado con éxito técnicas de aprendizaje automático que a partir de un conjunto inicial de ejemplos positivos y negativos son capaces de inferir expresiones regulares para la detección de cadenas o grupos de cadenas deseadas. Este es el caso de la empresa WhizBang!, cuyo sistema FlipDog (http://flipdog.monter.com), descrito más adelante, detecta información sobre puestos de trabajo. El proceso de aprendizaje consiste en almacenar las fuentes HTML en una forma tabular y sobre ellas se marcan los ejemplos positivos, los nombres de empresas, puestos ofertados, condiciones salariales, etc., para generar reglas de detección basadas en expresiones regulares.

Codificación por estructura. Algunos sistemas que trabajan sobre un dominio reducido pueden inferir el significado de los datos a partir de la estructura del documento. Si la estructura sigue un patrón estable se puede combinar técnicas de expresiones con algunas reglas semánticas que toman en cuenta estructuras. Este es el caso de sistemas como Citeseer (Citeseer) que es capaz de procesar artículos científicos online. Este tipo de documentos tiene una estructura fija y en ellos se pueden basar las reglas de identificación de títulos, autores, resúmenes, citas a otros artículos, etc. Las reglas de estructura, normalmente construidas ad-hoc para cada sistema, dependen en gran medida del dominio y del formato que usen los documentos (HTML, PDF, LaTeX, etc.).

Codificación Visual. Como los documentos online están orientados para consumo por lectores humanos, la composición visual de éstos es otra fuente que aporta significado a los datos extraídos. Sobre todo en formatos como HTML, donde la forma de visualizar el documento no está unívocamente ligada a una fuente (distintas fuentes producen igual efecto visual resultante), es una ayuda contar con esta información para inferir el significado. Para procesar el contenido visual de un documento es necesario contar con un modelo de dos dimensiones del emplazamiento de los datos. Con este modelo se pueden comprobar las posiciones relativas y absolutas entre elementos del documento. Información sobre si una etiqueta y un dato está en la misma línea virtual, si una frase esta debajo de un gráfico o si unos números están en la misma columna la cual aporta información muy valiosa.

Este es el caso del sistema AIDAS (Hoog et al, 02) del proyecto IMAT (IMAT) descrito más adelante, que procesa manuales técnicos mediante distintas ontologías con el objetivo de construir un documento para propósitos de educación y formación. Los documentos, originalmente en formato PDF, se procesan y se extrae información sobre el aspecto y colocación visual de cada parte, como se indica en el ejemplo:

```
l (text('12.1',  'Times-Bold-12'), area (60,160,34,14), [ ],
    [position=left          /* alineado al margen derecho */
    , column = 3            /* en la tercera columna*/
    ,  fontsize = large     /* fuente grande*/
    , emphasis = bold       /*negrita*/
    ]).
```

Posteriormente reglas de inferencia permiten deducir la semántica de las partes considerando toda la información disponible.

En los documentos HTML está necesidad se hace más notoria. Con un procesamiento simple de las fuentes no es posible determinar si dos datos se encuentran en una misma línea o columna visual o ni siquiera si están cerca uno del otro. Para determinar esta información es necesario realizar el mismo proceso que hacen los navegadores interpretando etiquetas HTML, funciones javascript, hojas de estilo, etc., para visualizar un documento y además construir un modelo espacial de su contenido que se puedan consultar desde programas. Para el formato HTML existen varias librerías de software (IceSoft, WebRenderer) que permiten construir este tipo de modelos donde cada elemento del documento HTML tiene asignado coordenadas (X,Y) para que sean consultadas e interpretadas, como se ilustra en la figura 25.

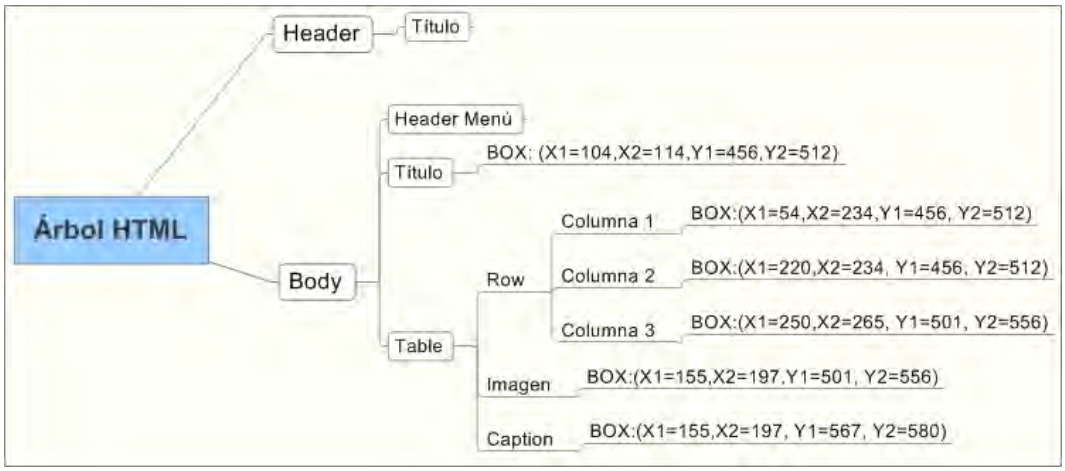


Figura 25 Árbol HTML extendido con coordenadas visuales

A este proceso se le suele denominar renderización[19], del inglés render, que significa retratar.

2.6.3.3. Aproximaciones estadísticas y no simbólicas

Modelos Ocultos de Markov

Una de las técnicas estadísticas frecuentemente usadas en extracción de información son los modelos ocultos de Markov[20] (HMM del inglés Hidden Markov Model). Es un modelo muy eficiente desde el punto de vista computacional y con muy buenos resultados para algunas tareas de procesamiento de lenguaje natural. Se han utilizado con éxito para tareas de asignación de etiquetas gramaticales (POS tagging) (Kupiec, 92), para detección de entidades dentro de textos (Bikel, 97) o para la clasificación de documentos según su tema principal (Yamron, 98). Como muchas otras técnicas estadísticas requiere de la definición de un modelo a priori y de un conjunto de datos de entrenamiento.

Un modelo de primer orden, normalmente usado, es un autómata que consta de un conjunto de estados Q, de los cuales se especifican estado inicial: q_o y final q_f. Entre los estados se definen transiciones $(q \to q')$ y además se define un alfabeto de salida para cada estado. El modelo de entrada al autómata consiste en la asignación de probabilidades de transición entre dos estados $P(q \to q')$ y de la probabilidad de emitir un determinado símbolo de salida en cada estado $P(q \uparrow s)$. Con esto la probabilidad total de emitir una cadena x de salida por el autómata M es de:

$$P(x/M) = \sum_{q_1 \ldots . q_i \in Q} \prod_{k=1}^{l+1} P(q_{k-1} \to q_k) P(q_k \uparrow x_k)$$

Formula: Probabilidad de una cadena de salida en un Modelo Oculto de Markov

[19] La **renderización** es el proceso de generar una imagen desde un modelo. Los medios por los que se puede hacer un renderizado van desde lápiz, pluma, plumones o pastel, hasta medios digitales en dos y tres dimensiones. La palabra *renderización* proviene del inglés *render*, y no existe un verbo con el mismo significado en español, por lo que es frecuente usar las expresiones *renderizar* o *renderear*.

[20] En un modelo *oculto* de Markov, el estado no es visible directamente, sino que sólo lo son las variables influidas por el estado. Cada estado tiene una distribución de probabilidad sobre los posibles símbolos de salida. Consecuentemente, la secuencia de símbolos generada por un HMM proporciona cierta información acerca de la secuencia de estados. Los modelos ocultos de Markov son especialmente aplicados a reconocimiento de formas temporales, como reconocimiento del habla, de escritura manual, de gestos, etiquetado gramatical o en bioinformática.

La asignación de probabilidades a las transiciones y a las salidas requiere de un trabajo manual de entrenamiento y construcción de un modelo probabilístico del dominio.

Modelos Vectoriales

Los modelos vectoriales son muy usados en la clasificación de páginas y documentos Web en clases predefinidas para ser mostrados como resultado de buscadores como altavista [AltaVista] o similares. El propósito es modelar las distintas clases de documentos como vectores de palabras y poder calcular la distancia vectorial de un documento, también representado como un vector hacia la clase.

El espacio vectorial tiene tantas dimensiones como palabras, se consideran en el dominio abordado y el valor de cada dimensión puede ser en número de apariciones de la correspondiente palabra. De esta manera un documento queda modelado como un vector de números enteros que representan el número de ocurrencias de una palabra. Se puede calcular la distancia entre dos documentos como la distancia entre dos vectores de igual dimensión.

Estos sistemas suelen tener predefinidas categorías de documentos cuyo vector central de cada categoría es la suma normalizada de todos los vectores de los documento presentes. Para un documento nuevo se puede determinar la distancia mínima que debe tener hacia el centro de la clase que se considere un miembro de la misma. Cada aplicación en particular adopta este método a sus necesidades, ya sea seleccionando solamente ciertas palabras, normalizando el valor de los vectores o modificando el cálculo de las distancias.

El buscador altavista aprovecha esta aproximación para mostrar resultados de cuantas más clases posibles. Para ellos recorre formando una espiral el espacio vectorial de soluciones a una búsqueda y muestra solamente un representante de cada clase encontrada. Le deja al usuario la decisión si quiere ver más documentos de la misma clase.

2.6.3.4. Aproximaciones Basadas en el Lenguaje Natural

En el campo de la extracción de información una de las técnicas más usadas y que más expectativas han generado y genera es el procesamiento del lenguaje natural (PLN). El procesamiento automático del lenguaje humano data desde la segunda guerra mundial cuando se diseñaron sistemas de traducción automática, que simplemente traducían palabras de un idioma a otro. Estas primeras experiencias demostraron que es necesaria una labor mucho más profunda en teorías lingüísticas, semánticas así como en representación del conocimiento si se quieren abordar este tipo de problemas. Una de las máximas aprendidas fue que el lenguaje natural que podemos procesar con maquinas no es el lenguaje humano en toda sus extensión, sino más bien un subconjunto de él. El

campo del PLN, en los años sesenta, contaba con la teoría de lenguajes de Chomsky (Chomsky, 55) así como en las técnicas necesarias para su procesamiento automático (autómatas).

En un sistema clásico de procesamiento de lenguaje natural se puede distinguir estas fases:

Figura 26 Secuencia clásica de un sistema de procesamiento de lenguaje natural

Nivel fonético

Este primer nivel identificado en el modelo clásico de una aplicación PLN tiene su justificación cuando se trabaja con una entrada de audio. Su objetivo es convertir las señales acústicas en cadenas de caracteres. Es un proceso no exento de ambigüedades y muy sensible a la calidad de entrada. Hoy en día existe una gran variedad de aplicaciones que realizan esta tarea IBM Vía Voice, DragonTalk, etc.

La tecnología más efectiva usada por las aplicaciones comerciales hace uso de técnicas de reconocimiento de patrones de señales obteniendo como resultado la transcripción textual de la entrada.

Nivel Léxico

El objetivo de este nivel es segmentar el texto de entrada en palabras organizadas en frases. Este problema, a primera vista trivial, debe gestionar fenómenos como abreviaturas, palabras partidas al fin de la línea. Palabras desconocidas o con fallos ortográficos. Así mismo dispone de analizadores específicos para reconocer y asignar valores a números, fechas u otros tipos básicos del modelo con el que se trabaje.

El objetivo de esta fase es proporcionar palabras preprocesadas para su posterior análisis en estructuras llamadas *tokens* que albergan la información sobre el tipo de cadena que se está tratando (cadenas alfanuméricas, puramente numérica, puntuación, cadena de control, etc.) así como información adicional sobre la posición de la cadena (número de línea, columna, etc.), documento origen, tipo de letra y similares.

Nivel morfológico

La morfología es la ciencia lingüística que trata con las palabras como unidad de estudio. Su cometido es estudiar las posibles formas que puede adoptar una palabra a partir de una raíz léxica. En algunas lenguas como el inglés las posibles formas flexionadas que pueden adoptar una palabra raramente alcanza diez, sin embargo en otras lenguas que incluyen fenómenos de declinación, tengan morfología aglutinante o simplemente tenga una morfología rica en flexiones, el número de formas puede superar la centena con facilidad.

Los módulos de procesamiento morfológico se pueden usar en dos sentidos, para el análisis de texto se suele tomar como entrada una forma desplegada (forma tomada de la fuente) de una palabra y el analizador debe interpretarla determinando las posibles raíces de la palabra así como sus características morfológicas, propias para cada categoría gramatical. Las categorías gramaticales (en inglés: POS: Part-of-speech) determina como actúa una palabra dentro de una estructura de sintagma[21] o frase.

Ejemplo: posibles resultados de análisis morfológico para la palabra "**lista**".

$$\begin{bmatrix} POS & nombre \\ numero & \text{sin } gular \\ genero & femenino \end{bmatrix} o \begin{bmatrix} POS & verbo \\ \text{mod } o & indicativo \\ numero & \text{sin } gular \\ persona & tercera \\ tiempo & presente \end{bmatrix} o \begin{bmatrix} POS & verbo \\ \text{mod } o & imperativo \\ numero & \text{sin } gular \\ persona & segunda \\ tiempo & presente \end{bmatrix} o \begin{bmatrix} POS & adjectivo \\ numero & \text{sin } gular \\ genero & femenino \end{bmatrix}$$

La ambigüedad de categorías gramaticales y del conjunto de soluciones presentado para una misma palabra aumenta el costo computacional de los sistemas de procesamiento de lenguaje natural. Los sistemas de desambigüación se suelen basar en métodos estadísticos que calculan las probabilidades de secuencias de categorías gramaticales en cadenas de N palabras (también llamados N-gramas). Dado un corpus es posible determinar la probabilidad de una cadena de categorías:

Determinante + Adjetivo + Sustantivo frente a otra cadena de categorías diferentes.

La dirección opuesta a la del análisis se utiliza en sistemas de generación de texto. Dado su lema e información morfológica el módulo debe generar la forma adecuada de la palabra.

[21] es una unidad sintáctica inmediatamente superior al constituyente no-sintagmático, que a su vez es el rango inmediatamente superior a la palabra o núcleo sintáctico, constituida por un conjunto de elementos lingüísticos organizados jerárquicamente en torno a un núcleo y caracterizados por desempeñar la misma función. Se trata, por tanto, de una unidad de función sintáctica.

Estos sistemas tienen su utilidad en la generación de lenguaje natural o en sistemas de ayuda a la investigación lingüística (García-Serrano et al, 98).

Existen varias soluciones tecnológicas para bordar el tema de análisis de palabras para obtener la información necesaria para posteriores procesamientos. A continuación se describen brevemente algunas de ellas.

- ***Morfología tabular***. Para idiomas con un número de formas posibles abordable (el español tiene más de un millón de formas desplegadas) se rellena una tabla cuya clave es la palabra en forma desplegada (en alguna literatura también llamada forma de superficie) y la información asociada contiene todas las estructuras de rasgos morfológicos asociados. Una aproximación computacionalmente muy eficiente aunque asociada con un gran consumo de recursos de memoria.
- ***Morfología inferida.*** Sobre todo apta para idiomas morfológicamente ricos ya que realiza el análisis de una palabra en tiempo real bajo demanda. Se suelen usar reglas de inferencia, expresiones regulares o para sistemas con altos requisitos en su velocidad autómatas finitos llamados transductores. Esta última opción desarrollada por sistemas (PCKIMMO) permiten procesar varios miles de palabras por segundo para lenguas con mucha complejidad morfológica, como son las lenguas semíticas o árabes (Kiraz, 95).

Nivel sintáctico

El procesado morfológico de las palabras nos permite trabajar a nivel de tema de cada palabra, que es la parte que lleva en núcleo del significado. A niveles superiores las palabras se organizan en estructuras llamadas sintagmas y a su a vez estos se organizan en frases que se pueden considerar unidades de discurso. Las combinaciones que forman frases se definen en la gramática de cada lengua y es el procesamiento sintáctico que verifica la validez de estas estructuras. Los diagramas más comunes que permiten conocer una estructura sintáctica se denominan arboles sintácticos, un ejemplo se puede ver en la figura 27.

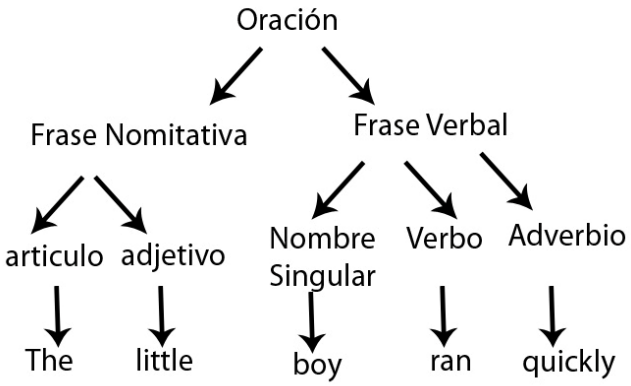


Figura 27 Esquema de un árbol sintáctico

Este un proceso computacionalmente muy exigente y no exento de ambigüedades que solamente el contexto, el procesamiento semántico o el uso de heurísticas o métodos estadísticos pueden intentar resolver. En algunos sistemas se opta por no realizar análisis de frase completos y quedarse a nivel de sintagmas con estructuras más simples. Este tipo de análisis se suele denominar: análisis no profundo (shallow parser). Estos sintagmas (en inglés phrases) pueden ser de varios tipos.

Aunque este tipo de análisis no profundo no es exhaustivo suele ser muy eficiente y suficiente para los propósitos de los sistemas que usan procesamiento de lenguaje natural sobre un dominio limitado. Algunos de estos analizadores denominados analizadores de partes (del inglés: chunk parsers), pueden identificarse relaciones entre los distintos sintagmas. Ciertos tipos de fenómenos como aposiciones[22], algunas coordinadas o subordinadas pueden ser identificados y resueltos enlazando sintagmas con relaciones semánticas básicas. Este es el caso del analizador SCHUG (Declerck, 02) que identifica aposiciones, entre otros fenómenos, como las mostradas en el ejemplo, la cual establece una relación de equivalencias entre ambos sintagmas.

Figura 28 Identificación de algunas relaciones entre sintagmas en Schung.

[22] es una construcción de dos elementos gramaticales unidos, el segundo de los cuales especifica al primero.

Los analizadores sintácticos, tanto los completos como los poco profundos hacen uso de distintos formalismos para expresar las reglas de construcción de las estructuras lingüísticas.

Nivel semántico

El análisis semántico tiene como objetivo estudiar el significado del texto analizado. Suele realizarse después de construir completamente o parcialmente las estructuras sintácticas. Aunque existen muchas definiciones sobre qué y cómo debe entenderse el término '***significado***' para los propósitos de este trabajo se considera el significado como la relación entre las expresiones lingüísticas y los que estas denotan (Woods, 1975). Más concretamente, en el ámbito de la Web Semántica, el espacio de conceptos susceptibles de ser denotados se formaliza en un modelo semántico descrito como una ontología, como se muestra en la figura 28.

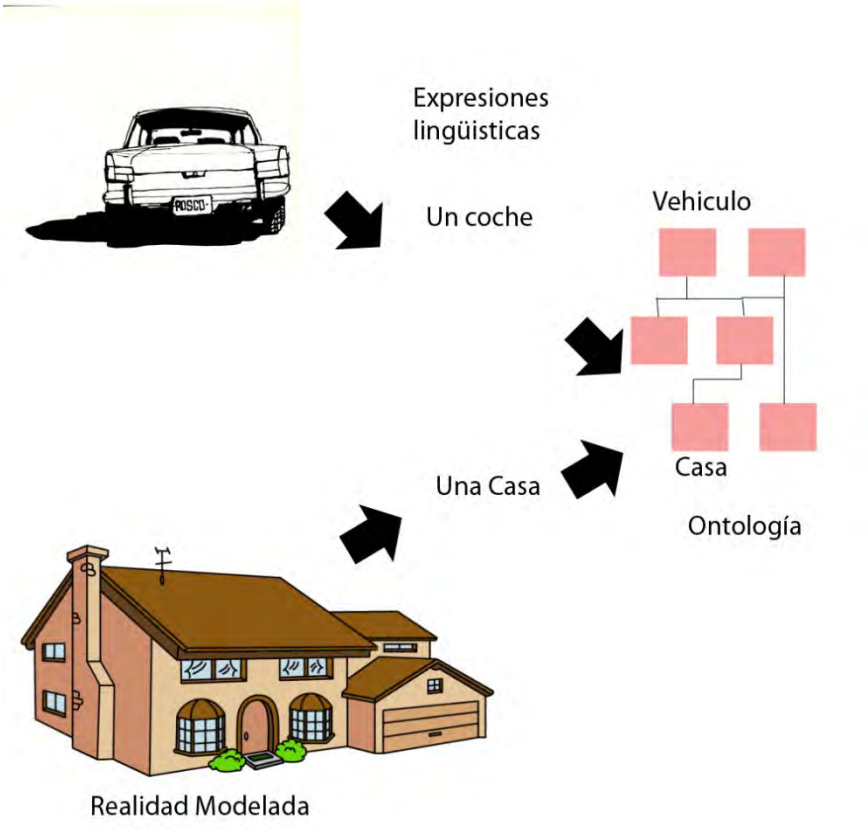


Figura 29 Conceptos reflejados en una ontología.

En las últimas décadas se han tomado varias aproximaciones para que las computadoras realicen análisis semántico del habla humano. La familia de teorías que toma la semántica como una ciencia formal, cercana a las matemáticas (Wittgestein, 53) enumera algunas propiedades deseables en cualquier aproximación tomada (Goñi, 98):

- **Composicionalidad:** Como se enuncia en el trabajo de Frege [Frege, 23] el significado de una frase se calcula en función de sus partes y de la forma que éstas se combinan. Las partes a las que se refiere esta propiedad pueden ser palabras o sintagmas.
- **Basada en modelos:** La utilidad de un modelo de la realidad definido de manera formal. Como se ha comentado anteriormente en este trabajo se tomarán los formalismos de ontologías, que modelan conceptos, atributos y relaciones de un dominio. El propósito del análisis semántico es ligar elementos lingüísticos a este modelo ontológico.
- **Tipado[23] estricto.** Las partes identificadas en el proceso del análisis que sirven para construir el significado están tipadas dentro del modelo conceptual definido. Como ejemplo en algunos sistemas, los verbos suelen denotar relaciones N-arias dentro del modelo, así como sintagmas nominales denotan subconjuntos de conceptos o instancias de los mismos.
- **Formalismo lógico.** El modelo subyacente debe soportar algún formalismo lógico que permita interpretar las expresiones adquiridas por el proceso de análisis. Existen varios sistemas que usan lógica de predicados para esta interpretación, pero la mayoría de ellos no ha demostrado tener suficiente expresividad para capturar la riqueza del lenguaje (Sowa, 93). En formalismos de la Web Semántica la elección de la lógica se centran en formalismos que permitan una asunción de mundo abierto, como la lógica descriptiva definida sobre lenguajes como el OWL-DL.

Para los propósitos de adquisición de contenido para la Web Semántica, el significado de las fuentes textuales debe ser traducido a un formalismo de ontologías. El problema se centra en la traducción de las partes lingüísticas en conceptos o aserciones dentro del modelo. Existen varias iniciativas que proponen un paso intermedio entre los constituyentes sintácticos (frases, sintagmas o palabras) y el modelo ontológico final.

Una manera de descomponer el proceso de traducción (de partes sintácticas, verbales en este caso, al modelo subyacente) para la búsqueda de significado es introducir un formalismo intermedio de primitivas semánticas que modelen posibles significados.

[23] es una caracterización precisa de las propiedades estructurales y de comportamiento que comparten una serie de entidades.

Teoría de Schank (Schank, 75) fue ideada para representar el significado de una frase independientemente del idioma en el que fuese expresada mediante una serie de primitivas semánticas. Tanto conceptos del dominio como sintagmas puede hacer referencias a estas primitivas.

A continuación se presenta una lista de las primitivas semántica propuesta por Schank para verbos y sintagmas verbales:

Acciones físicas de las personas:

- PROPEL: aplicar algo
- MOVE: mover parte del cuerpo
- INGEST: tomar algo adentro de un objeto animado
- EXPEL: expulsar algo de un objeto animado
- GRASP: agarrar algo físicamente

Cambios de estado

- PTRANS: cambiar de sitio físicamente
- ATRANS: cambiar una relación abstracta

Actos Instrumentales

- SPEAK: producir un sonido
- ATTEND: dirigir los sensores hacia un estimulo

Actos mentales

- MTRANS: transferencia de información
- MBUILD: crear y combinar pensamientos

Es en el léxico (diccionario de palabras) donde se definen estas primitivas asociadas a los verbos. De allí vienen sus principales críticas (Goñi, 98):

- Son vagas y ambiguas
- No contemplan otras categorías como los nombres, etc.
- No tiene en cuenta el resultado de la sintaxis, sobre todo para estructuras complejas.

Otra teoría enunciada por Pustejovsky (Pustejovsky, 91) denominada Léxico Generativo atribuye a cada entrada del léxico cuatro niveles de descripción semántica:

- **Estructura Argumental:** Incluye la especificación funcional para relacionarlo con la estructura sintáctica.
- **Estructura de eventos:** Para situar el enunciado en un espacio temporal dentro del modelo.
- **Estructura Qualia.** Atributos de significado en 4 dimensiones:
    - **Constitutivo:** relación del objeto con sus constituyentes o partes (Partes, Peso, Material, etc.).
    - **Formal:** características que permiten distinguir el objeto de otros (color, orientación, forma, etc.).
    - **Télico:** Propósito y función del objeto (propósito del agente que realiza la acción).
    - **Agentivo:** Lo que hizo posible que el objeto exista (creador, cadena de causalidad, etc.)

Otra manera de aliviar el proceso de búsqueda de significado es la introducción de algunas relaciones semánticas generales en el léxico original para que sean confirmadas o validadas en el modelo semántico que debe incluirlas. Estos léxicos denominados léxicos semánticos incluyen algunas de las relaciones clasificadas por el trabajo de Saint-Dizier y Viegas (Saint-Dizier-Viegas, 95).

- Relaciones Jerárquicas
    - Hiponimia/hiperonimia: relación de generalización y especialización de términos.
    - Meronimia/holonimia: relación de partes o composición.
- Relaciones no Jerárquicas
    - Sinonimia
    - Antonimia
    - Complementariedad
    - Oposición

El léxico semántico actualmente más usado para el idioma inglés es WordNet [Miller 95] el cual define un conjunto de significados (*denominados synsets*) con las distintas acepciones posibles. Estos *synsets* se organizan en redes semánticas conectadas por relaciones básicas de hiponimia[24], meronímia[25], sinonimia[26] y antonimia[27].

[24] entre dos palabras se establece una relación de **hiponimia** cuando el significado de una de ellas –hipónimo- está incluido en el significado de la otra

[25] es una relación semántica no-simétrica entre los significados de dos palabras dentro del mismo campo semántico. Se denomina **merónimo** a la palabra cuyo significado constituye una parte del significado total

Existe una versión multilingüe de este diccionario, llamada EuroWordNet (EWN) que incluye información léxico-semántica para la mayoría de las lenguas europeas. La ventaja es que los synsets son comunes a las palabras independientemente del idioma.

2.6.4. Almacenamiento de la Información: Lenguajes de la Web Semántica

Una de las diferencias que existen entre sistemas tradicionales de extracción de información y sistemas de generación de contenido semántico es el formato de los datos que generan. Mientras que en los tradicionales sistemas el paradigma de almacenamiento no era característica principal y más bien se adaptaba a las necesidades concretas de cada uso, en el paradigma de la Web Semántica son las ontologías que imponen el formalismo expresado con la sintaxis derivada del XML.

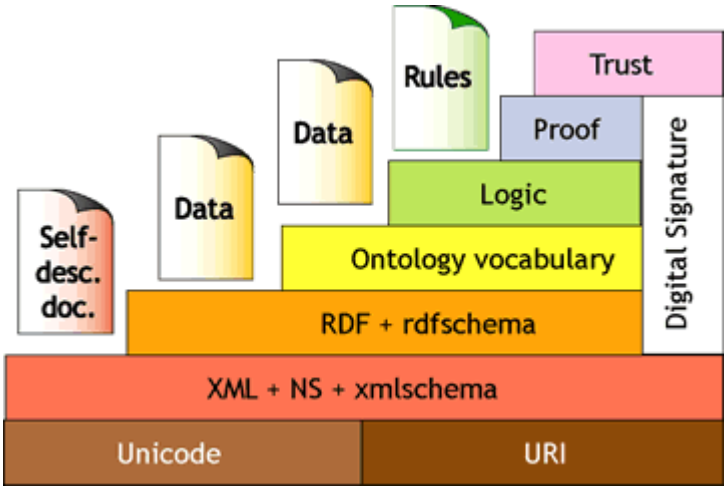


Figura 30 Pirámide de los lenguajes semánticos (W3C)

Como se ilustra en la figura 29 la evolución de los lenguajes semánticos ha seguido esta pirámide y muchos de ellos están construidos en base a otros, más simples. La base de todos ellos es XML, por encima podemos encontrar RDF y RDFS que contempla algunas primitivas semánticas básicas (subclase, propiedad, etc.), por encima de RDF y RDFS podemos encontrar DAML + OIL, por encima de este OWL y por encima de este están los lenguajes específicos de cada dominio, etc. Esta filosofía

de otra palabra, denominada ésta holónimo. Por ejemplo, *dedo* es merónimo de *mano* y *mano* es merónimo de *brazo*; a su vez, *brazo* es holónimo de *mano* y *mano* es holónimo de *dedo*.

[26] dos palabras son **sinónimas** nos estamos refiriendo a que entre ellas existe una relación de igualdad de significado, es decir, que ambas pueden ser empleadas para expresar una misma cosa.

[27] la **antonimia** es un tipo de relación semántica que se establece entre palabras que poseen significados totalmente contrarios, como bueno – malo, frío – calor, o alto – bajo.

inclusiva permite que un agente no entienda un determinado nivel, puede trabajar a un nivel más bajo obteniendo menos información sobre el significado de los datos.

Algunos de los lenguajes más usados en el contenido de la Web Semántica son:

RDF y RDF Schema

Es un estándar propuesto por el consorcio de la WWW (RDF(s)) que consta de dos lenguajes:

- **RDF Schema** para la definición de la ontología, que incluye primitivas semánticas para descripciones de clases, relaciones de herencia entre ellas, atributos de clases y tipos de atributos. En lenguajes semánticos los atributos son entidades de primer nivel, al igual que las clases.
- **RDF** para la definición de los datos, instancias de la ontología. Permite definir nuevas instancias con un identificador único, (URI) así como los valores de sus atributos.

Los recursos se describen en formas de tripletas de (Recurso, Propiedad, Valor), donde el valor puede ser otro concepto de la ontología. Se puede ver conceptualizando como un grafo donde los nodos son recursos (Conceptos, Instancias o Tipos Básicos) y los arcos propiedades (Atributos). Tanto RDF como RDF Schema (RDF(S)) permiten tres formas de serialización los que los hace lenguajes muy cómodos para desarrollos informáticos: XML, XML abreviado y tripletas.

Ejemplo de un recurso descrito en RDF serializado en modo XML abreviado

```
<rdf:RDF
  xmlns: rdf = "http://www.w3.org/2009/09/24-rdf-sytax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
<rdf:Description rdf:about="http://en.wikipedia.org/Gerson_Villa">
    <dc:title>Gerson Villa</dc:title>
    <dc:publisher>Wikipedia</dc:publisher>
</rdf:Description>
</rdf:RDF>
```

OIL

OIL (Horrocks et al, 00) es un lenguaje construido sobre las definiciones de RDF y RDFS. Enriquece la semántica del lenguaje anterior con capacidades de inferencia[28], con el

[28] Una **inferencia** es una evaluación que realiza la mente entre conceptos que, al interactuar, muestran sus propiedades de forma discreta, necesitando utilizar la abstracción para lograr entender las unidades que

formalismo de la lógica descriptiva. Existen varias capas en la arquitectura del lenguaje OIL:

- *Core OIL:* Coincide en gran medida con RDF Schema.
- Standard OIL: añade primitivas semánticas para poder realizar inferencia sobre los datos.
- *Instance OIL:* permite implementar funcionalidades propias de una base de datos.
- *Heavy OIL:* el lenguaje OIL completo que además de las características anteriores incluye nuevas primitivas semánticas.

Esta estructuración en capa permite obtener tres grandes ventajas:

1. La aplicación que la use no está obligada a trabajar con todo el conjunto del lenguaje.
2. Al igual que los agentes en la Web Semántica, si una aplicación entiende solo *Core OIL* sigue siendo capaz de obtener información y
3. Aplicaciones preparadas para versiones completas del lenguaje entienden recursos descritos por capas básicas.

El desarrollo de este lenguaje está parado, y el relevo lo ha tomado la iniciativa conjunta entre Europa y EEUU llamada DAML+OIL.

DAML+OIL

Como se ha comentado anteriormente, DAML+OIL (DAML+OIL) es una iniciativa conjunto de las comunidades científicas de los Estados Unidos, con el lenguaje DAML, originalmente esbozado por Tim Berners Lee, y la comunidad Europea con el lenguaje OIL. Se basa en el paradigma de orientación a objetos e incluye axiomas que permiten generar nuevo contenido y verificar él existente. Su desarrollo paró en Diciembre del 2001.

OWL

OWL, acrónimo de *Ontology Web Language* (OWL) es el lenguaje que en febrero del 2004 el consorcio W3C ha aprobado como propuesta de estándar. En una continuación de las iniciativas OIL y DAML+OIL y al igual que OIL se divide por capas:

componen el problema, creando un punto axiomático o circunstancial, que nos permitirá trazar una línea lógica de causa-efecto, entre los diferentes puntos inferidos en la resolución del problema. Una vez resuelto el problema, nace lo que conocemos como postulado, o una transformada de la original, que al estar enmarcado en un contexto referencial distinto, se obtiene un significado equivalente. Utilizada a menudo en los motores de inferencia de los Sistemas Expertos.

- *OWL-Lite:* El subconjunto más sencillo, incluye la expresividad del RDFS aumentada con algunas facetas (atributos de atributos), como la cardinalidad de los valores. Está diseñado para representar taxonomías con restricciones básicas sobre los valores.
- *OWL-DL:* Es el mínimo lenguaje para asegurar la completitud para propósitos de de inferencia con lógica descriptiva.
- *OWL-Full:* Incluye en vocabulario completo, con el máximo poder de expresividad (extensible). No garantiza una buena eficiencia en los procesos de inferencia.

2.6.5. Aproximaciones Existentes

En este capítulo se presentan algunas aplicaciones que procesan documentos online y extraen la información necesaria. Algunas de ellas incluyen funcionalidades de relleno de ontologías o de otros modelos semánticos mientras que otras hacen uso de la información estructurada para ofrecer algún servicio de valor añadido al usuario. En cada una se esboza la técnica usada para localizar los documentos, criterios usados para determinar su relevancia, formatos admitidos, procesos de extracción de información, la manera de almacenar esta información y procesos de explotación de la misma de cara al usuario.

2.6.5.1. GATE

Las siglas GATE vienen del acrónimo inglés *General Architecture for Text Engineering* (Arquitectura General para Ingeniería Textual) y es una plataforma que permite construir aplicaciones que incorporan de procesamiento de lenguaje humano (Cunningham, 02). Incorpora, entre otros, módulos funcionales de procesamiento y generación de lenguaje natural en varios idiomas (morfológicos, analizadores sintácticos, etc.), procesamiento de cadenas léxicas (expresiones regulares, reconocimiento de entidades a través de diccionarios, etc.) o módulo corrector ortográfico o de identificación de idioma. Su arquitectura lógica comprende tres partes bien diferenciadas:

- GDM *(Gate Document Manager)*: Sistema de gestión de documentos basado en el estándar SGML.
- GGI *(Gate Graphic Interface)*: Interfaz gráfico de usuario para labores de ingeniería de lenguajes, manejo de módulos incluidos, depuración de ejecuciones y herramientas de visualización.
- CREOLE: (*Collection of Reusable Objects for Language Engineering*): Comprende una colección de módulos de Ingeniería textual fácilmente ampliable.

Su gran ventaja sobre otros sistemas menos flexibles es la facilidad de incorporación de nuevos módulos para la construcción de aplicaciones. Los módulos deben seguir un interfaz común dado por CREOLE y con pocas líneas de programación se puede incorporar en la ejecución junto a otros módulos. La secuencia de ejecución se puede determinar gráficamente en un grafo de dependencias.

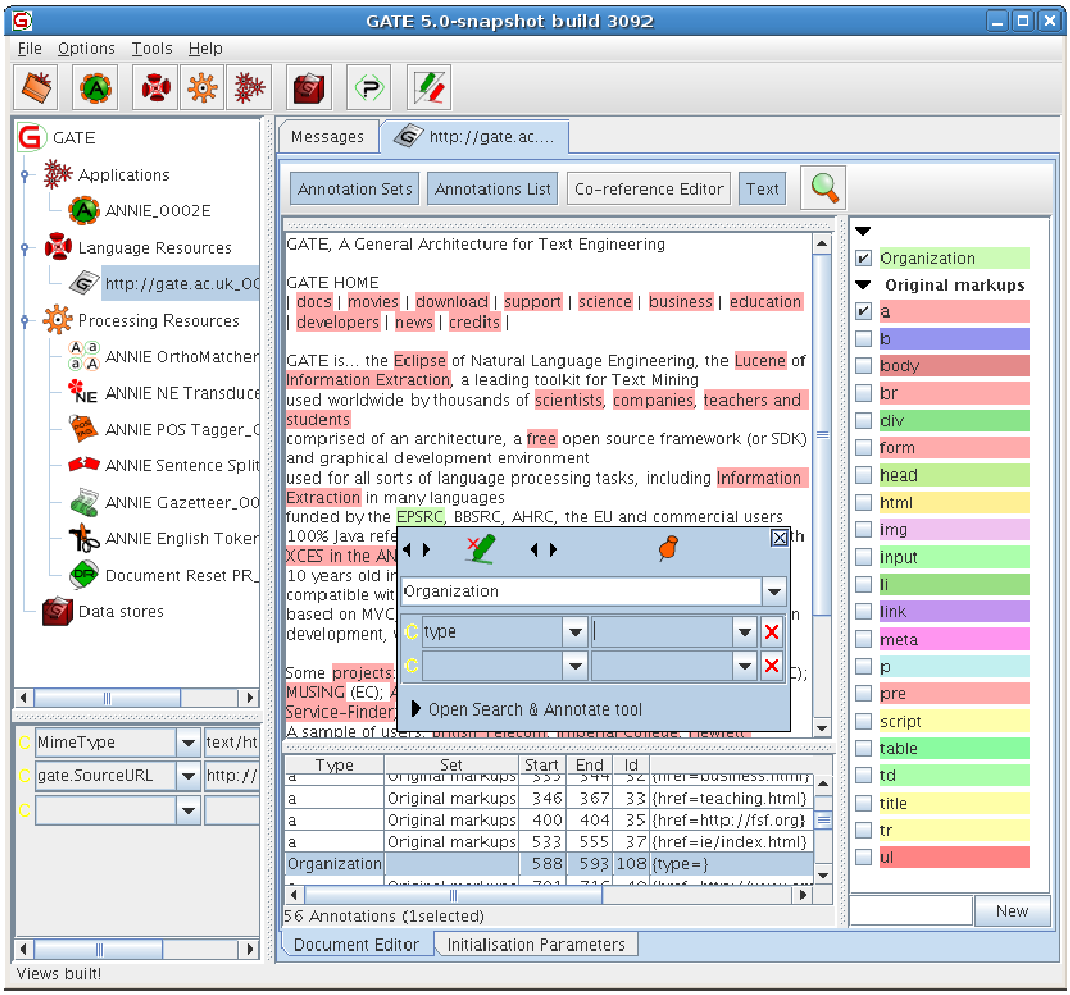


Figura 31 Recuperación de información a través de la aplicación GATE

A continuación se muestran algunos sistemas construidos sobre la plataforma GATE para propósitos de extracción.

2.6.5.2. ANNIE

ANNIE (Cunningham et al, 02b) es un sistema de extracción de información construido de acuerdo a la arquitectura GATE que incluye módulos de procesamiento de lenguaje natural:

- Procesado de fuentes textuales usando la codificación Unicode [Unicode]

- Analizador léxico para la identificación de palabras.
- Diccionario de nombres y entidades llamado *Gazzeetter*
- Anotador de categorías gramaticales (Part-of-speech tagger).
- Segmentador de frases
- Anotador semántico basado en gramáticas y expresiones regulares que asigna valores semánticos predefinidos a las partes del texto.
- OrthoMatcher: Identificador de relaciones entre nombre de identificados por el *Gazzeetter.*

Un complemento muy usado dentro de la arquitectura GATE para ANNIE es el sistema LaSIE (LaSIE) que entre otros ofrece módulos de lematización[29] y de reglas de extracción basadas en lenguaje Prolog[30].

2.6.5.3. Amilcare

Amilcare (Ciravegna, 01b) es un sistema de extracción de información basado en la arquitectura GATE y que incluye el módulo ANNIE (Cunningham et al, 02b). Este sistema ya incorpora la noción de ontología de dominio de manera explícita y ofrece funcionalidades que introducen los datos localizados en las fuentes.

Amilcare implementa el algoritmo (LP)2 (Ciravegna, 01) que mediante el uso de técnicas de aprendizaje automático construye reglas para la extracción de información.

Se maneja una aproximación mixta para las reglas:

- Reglas de formato que trabajan con expresiones regulares y trata el texto como cadenas de caracteres sin significado.
- Reglas inferidas en base a los resultados del procesamiento de lenguaje natural.

Las reglas se inducen a partir de párrafos de texto ya semánticamente anotado con etiquetas semánticas. Las etiquetas no necesariamente corresponden a una ontología previamente definida y el usuario tiene la posibilidad de definirlas libremente.

Algunas aplicaciones que hacen uso del sistema Amilcare:

[29] Lematización (Stemming en Ingles) es una técnica en la recuperación de datos en los sistemas de información (RDSI), esta técnica sirve para reducir variantes morfológicas de la formas de una palabra a raíces comunes o lexemas; con el fin de mejorar la habilidad de los motores de búsqueda para mejorar las consultas en documentos. Básicamente, este consiste en remover el plural, el tiempo, o los atributos finales de la palabra

[30] es un lenguaje de programación lógico e interpretado, bastante conocido en el medio de investigación en Inteligencia Artificial.

- MnM (Vargas-Vera et al, 02): MnM es una herramienta de anotación basada en ontologías que permite anotar páginas Web con contenidos semánticos de forma automática y semi-automática. MnM integra un navegador Web con un editor de ontologías y proporciona unas APIs abiertas para enlazar MnM con servidores de ontologías y para integrar MnM con herramientas de extracción de información. MnM trabaja con diferentes lenguajes de ontologías como RDF, DAML+OIL y OCML.
- OntoMat (OntoMat): Una herramienta interactiva de anotación de páginas Web basada en el sistema SCREAM (Handschuh et al, 02) que incorpora Amilcare. Ofrece funcionalidades para la creación y manteniendo de ontologías escritas en el lenguaje DAML+OIL.

2.6.5.4. KIM

(Knowledge and Information Management). Sistema desarrollado por la empresa OntoText parcialmente sobre la arquitectura GATE incluye funcionalidades de relleno de ontologías. Posee una ontología de propósito general predefinida con conceptos como personas, organizaciones, lugares, etc. Llamada KIMO (KIM Ontology).

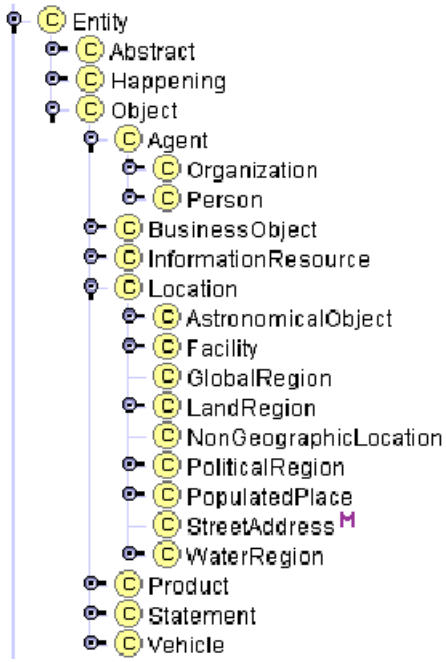


Figura 32 Vista parcial de la Ontología de dominio de KIM (Popov et al, 03)

Esta ontología parcialmente rellena permite localizar estructuras más complejas dentro de textos online y de esta manera completar su contenido. Para la detección de instancias de

entidades y relaciones se usan técnicas de encaje de patrones a nivel sintáctico y semántico.

Desde el punto de vista arquitectónico se apoya en GATE para el procesamiento de texto, Lucene (Lucene) para tarea de indexación y recuperación y en Sesame (Broekstra et al, 01) para la gestión de ontologías.

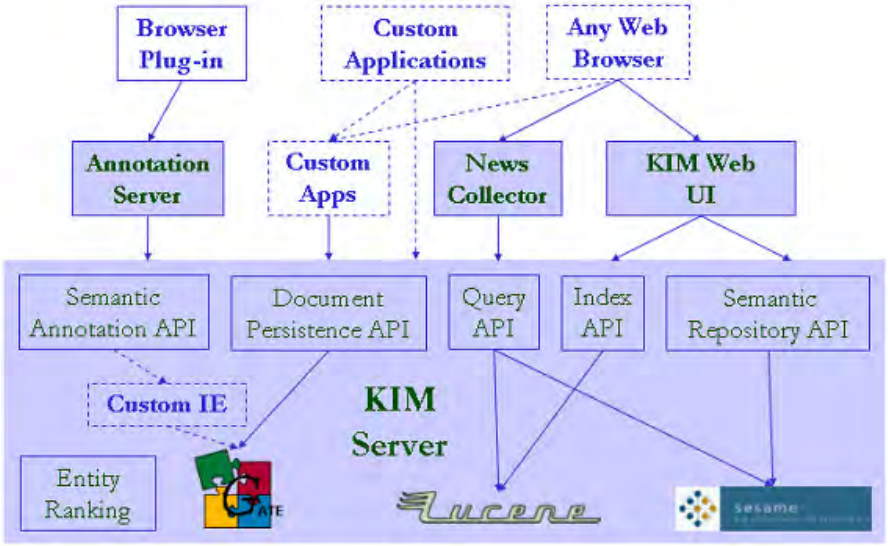


Figura 33 Arquitectura del sistema KIM

2.6.5.5. Citeseer

CiteSeer (CiteSeer) es un sistema de indexación y búsqueda de artículos científicos publicados en formato electrónico. Para localizarlos se sirve de buscadores genéricos de la Web (Altavista, Excite, Hotbot, etc.) junto con algunas heurísticas adicionales para ser convertidos en un formato de texto plano para ser procesado por el sistema de extracción de información. El sistema es capaz de extraer la información sobre el Titulo, Autor, Resumen así como consigue identificar las referencias a otros artículos. De esta forma se consigue construir una red de artículos que se hacen referencia unos a otros muy útil en labores de investigación y documentación.

Figura 34 Interfaz del buscador Citeseer

Citeseer convierte en texto plano de los artículos en datos estructurados con un significado común entre ellos. El modelo semántico que rellena el proceso de extracción está implícito en el modelo de la base de datos del sistema y no existe un modelo en forma de ontología explicito.

2.6.5.6. CREAM

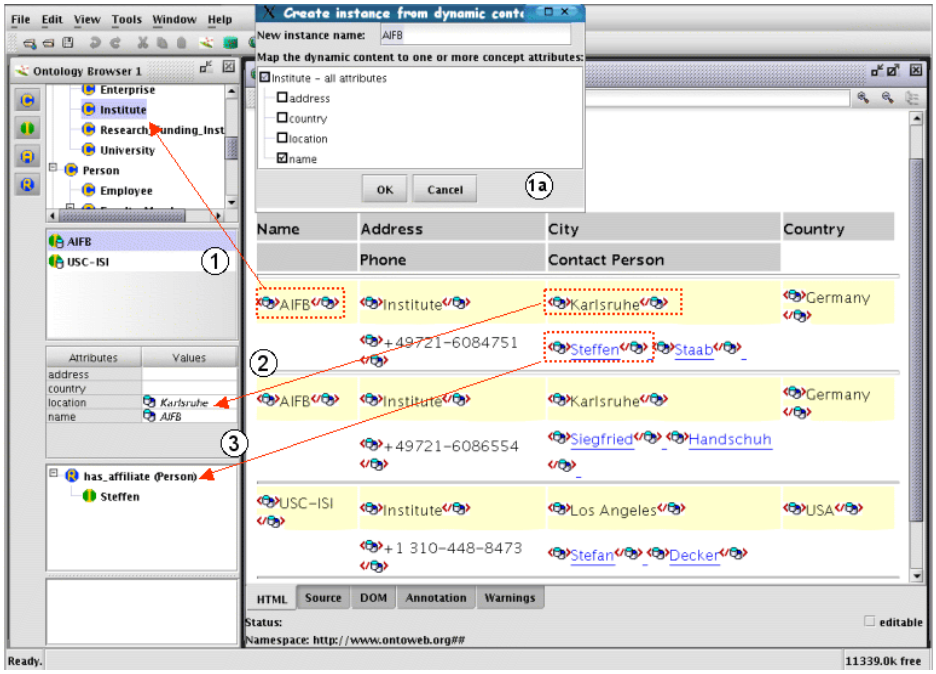


Figura 35 Interfaz de la aplicación CREAM

CREAM (Handschuh et al, 01) es una plataforma para la creación de meta-datos que incluye funcionalidades de inferencia, un navegador y un sistema de gestión documental todos ellos basados en ontologías. La primera versión de OntoMat fue desarrollada sobre esta versión.

2.6.5.7. AIDAS

La aplicación AIDAS (Hoog et al, 02) desarrollada en el marco del proyecto europeo IMAT IMAT) es un Wrapper para documentos de manuales técnicos. El objetivo de la extracción se concentraba en la construcción de material educativo y de formación a partir de manuales técnicos.

Las fuentes en formato PDF se preprocesaban y convertían a través del formato XPDF (XPDF) a un modelo de objetos. Este modelo sirve de entrada para reglas de descubrimiento de estructuras lógicas. Las reglas son de tres tipos:

- Reglas de aspecto (tipo de letra, párrafos, etc.).
- Reglas de geometría (organización visual dentro de la página).
- Reglas léxicas (expresiones regulares).

Las reglas extraen información y la indexan según varias ontologías:

- Ontología de fragmentos: captura aspectos del documento como su tamaño, posición de imágenes, presentaciones, etc.
- Ontología de dominio: captura conceptos relevantes en el texto.
- Ontología de aprendizaje (Instructional Ontology): captura los posibles objetivos en la formación y educación.

2.6.5.8. Ariadne

(Ambite et al, 98) Sistema de extracción e integración de datos de fuentes Web semi-estructuradas. Usa software específico (wrappers) para extraer información de cada fuente e insertaría en un repositorio común. El software de extracción usa gramáticas no sensibles al contexto para localizar valores dentro de las fuentes HTML. Además ofrece una herramienta de creación y gestión de estas gramáticas para facilitar la construcción y mantenimiento de los wrappers. La aplicación que explota este contenido se ha elaborado sobre un dominio de ocio de una ciudad (Restaurantes y Teatros de Cambridge).

La arquitectura de Ariadne comprende los siguientes componentes:

- Modelo de dominio
- Descripción de las fuentes de información.

- Colección de piezas de software de extracción de información. Este software (llamado Wrapper) envuelve los documentos de fuentes ofreciendo acceso como si de una base de datos se tratase.
- Planificador del acceso a las fuentes para una pregunta dada.

Con estos cuatro componentes es posible construir un sistema que le permite al usuario consultar varias fuentes agregadas bajo un mismo interfaz de manera transparente como si fuese una gran base de datos. Al contrario que otros sistemas que almacenan la información recuperada en un repositorio propio, Ariadne indexa el contenido en su ubicación original. Para propósitos de integración y consistencia dispone de axiomas específicos.

Otro modulo interesante de Ariadne es el de descripción de las fuentes de información. En este módulo se guardan las gramáticas que permiten acceder a los distintos datos ofrecidos por el sistema. Estas gramáticas permiten analizar el código fuente del documento e identificar dentro de él el dato para la extracción. Las gramáticas se pueden obtener de tres maneras:

- Mediante codificación manual en formato XML
- Como efecto colateral de la navegación hecha por el usuario
- Mediante inducción a través de un conjunto de entrenamiento

Figura 36 Pantalla del sistema Ariadne

2.6.5.9. Google News

Google News [GoogleNews] usa tecnología de wrappers para agregar más de 4000 fuentes Web. Esta aplicación se centra en la búsqueda de titulares en las distintas fuentes para

luego recabar la información que lo amplia. Muchos detalles de su implementación son secretos ya que se trata de una aplicación comercial.

En su aplicación de buscador genérico Google aplica su tecnología ***PageRank*** para clasificar los sitios por orden de importancia. Aplica como criterio principal para ordenar los resultados el número de enlaces que se dirigen desde los millones de páginas rastreadas hacia otros documentos y no, como el resto de los buscadores automáticos, el número de veces que una palabra clave se repite en una página. El concepto básico de algoritmo ***PageRank*** es que una página es más importante en la medida en que más páginas apuntan hacia ella. El algoritmo extiende este concepto no solo ejecutando la cantidad de enlaces, sino también normalizando de acuerdo a la cantidad de enlaces de una página, y propagando infinitamente de forma tal que la importancia de una página depende de:

- Cuantas o que paginas apunta a ella y la tipología de estas páginas
- La cantidad de enlaces en estas páginas.

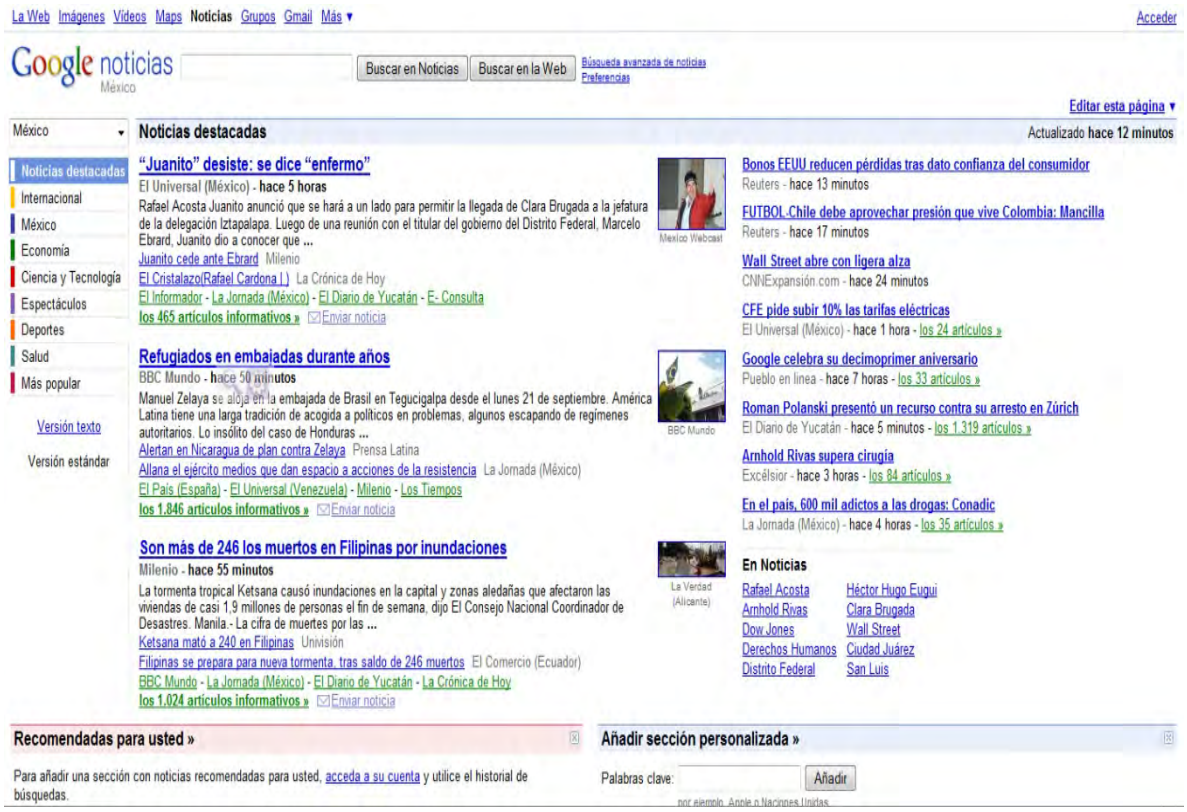


Figura 37 Agregador de noticias: Google News

2.6.5.10. Web Scraper

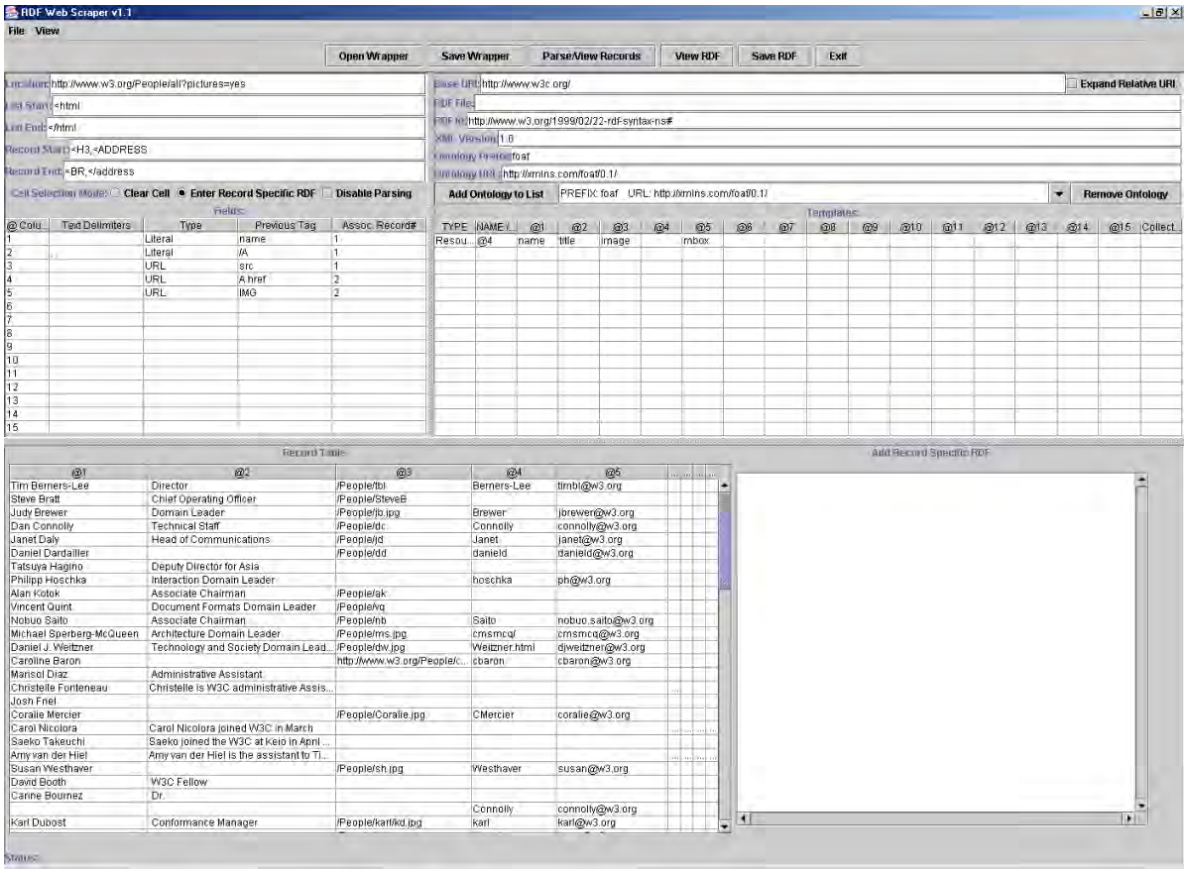


Figura 38 Pantalla principal del Web Scraper

Aplicación desarrollada en el grupo estadounidense mindswap.org permite que el usuario analice una página Web creando un Wrapper que traslade el contenido en un formato tabular. A partir de esta estructura se establecen reglas de conversación de columna en elementos RDF y de esta manera crear instancias por cada registro de la tabla.

2.6.5.11. GETSee

GETSee (GetSee) ofrece servicios de agregación de información financiera y de contabilidad a pequeña escala: Mediante software especifico para cada fuente es capaz de incluir información de más de 40 entidades bancarias, 10 proveedores de servicios de consumo (servicios de telefonía, gas, luz, etc.). Así como proveedores de servicios de correo electrónico. Cada proveedor tiene asociado un Wrapper implementado ad-hoc. Los wrappers trabajan con modelos de documentos para navegar por las fuentes hiper-

textuales y emplean expresiones regulares para la localización de la información. Aunque el sistema dispone de una plataforma de programación de los programas de adquisición que hacen uso de simuladores de navegación y lenguajes específicos para tratamiento de cadenas de texto, el mantenimiento de cada uno de los wrappers es una tarea considerable.

La plataforma GETSee codifica los distintos wrappers en un lenguaje escrito en XML. Estos wrappers ad-hoc para cada fuente se generan mediante un software específico llamado Plugin Engine. La extracción se basa en el modelado de los documentos online en arboles DOM[31] y el encaje de expresiones regulares para la localización de información. Dado lo sensible que son este tipo de extractores, existe una aplicación encargada de monitorizar el correcto funcionamiento de cada extractor y en caso de fallo (por cambio en la estructura de la fuente) se genera una alarma. En este caso es preciso reprogramar los extractores de acuerdo con los cambios producidos.

2.6.5.12. Whizbang

El software de WhizBang permite, mediante técnicas de aprendizaje automático inferir wrappers, Con un corpus de fuentes anotadas semánticamente y módulos de preproceso de lenguaje HTML a estructuras tabulares, el sistema es capaz de inferir reglas de detección de entidades deseadas (Por ejemplo: nombres de empresas, nombres personales, ofertas de trabajo, etc.).

2.6.6. Resumen de las Aplicaciones Vistas

En una tabla se resume las aplicaciones vistas en este recorrido. Los parámetros que se han evaluado son:

- **Fuentes:** Que tipo de fuentes, en cuanto al formato, es capaz de procesar la aplicación.
- **Salida:** Que tipo de salida, en referencia al formato, es capaz de producir el sistema.

[31] El ***Document Object Model*** (una traducción al español no literal, pero apropiada, podría ser ***Modelo en Objetos para la representación de Documentos*** o también ***Modelo de Objetos del Documento***), abreviado DOM, es esencialmente una interfaz de programación de aplicaciones que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos. A través del DOM, los programas pueden acceder y modificar el contenido, estructura y estilo de los documentos HTML y XML, que es para lo que se diseñó principalmente.

- **Tecnología:** Tipo de tecnología usada para propósitos de extracción.
- **Dominio:** Determina si la aplicación está restringida a algún tipo de dominio.

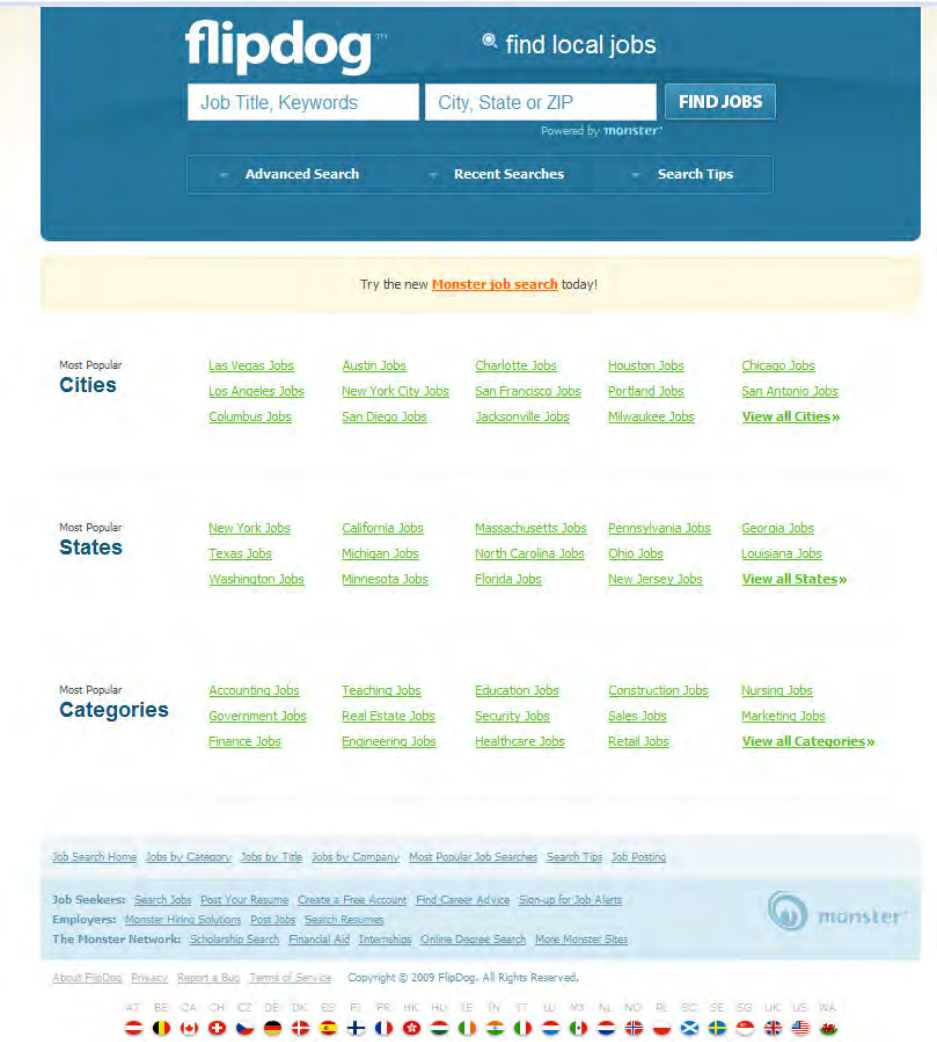


Figura 39 Aplicación del software de WhizBang! Para un portal de empleo

- **Auto Adaptable:** Si la extracción es resistente a cambios en la fuente.
- **Ontología:** Si el modelo de dominio es explícito o implícito.
- **Modo de ejecución:** Indica cómo se ejecuta la extracción, si en el momento de solicitar el dato o en un proceso previo.

Wrapper	Fuentes	Salida	Tecnología	Dominio	Auto Adaptación	Ontología	Modo ejecución
GATE	Texto plano	Cualquiera	PLN	Cualquiera	No	Implícita	Bajo demanda
ANNIE	Texto plano	Cualquiera	PLN	Cualquiera	No	Implícita	Bajo demanda
Amilcare	Texto plano	DAML o similar	PLN, Aprendizaje automático	Cualquiera	Si	Implícita	Batch
KIM	Texto plano	Cualquiera	PLN, patrones	Cualquiera	No	Explicita	Bajo demanda
Citeseer	PDF, PostScript	Base de datos	Expresiones Regulares	Publicaciones científicas	No	Implícita	Batch
CREAM	Texto plano	Cualquiera	Reglas de traducción	Cualquiera	No	Explicita	Batch
AIDAS	PDF	SQL	Procesado de Texto	Manuales de usuario	No	Explicita	Batch
Ariadne	HTML	Base de datos	Expresiones regulares	Cualquiera	No	Explicita	Bajo demanda
Google	HTML	Base de datos	Desconocida	Noticias	Desconocido	Implícita	Batch
Web Scraper	HTML	RDF	Desconocida	Cualquiera	No	Implícita	Bajo demanda
GETSee	HTML	Base de datos	Expresiones Regulares	Financiero	No	Implícita	Bajo demanda
Whizbang!	HTML	Base de datos	Aprendizaje automático y expresiones regulares	Cualquiera	Si	Implícita	Batch

Tabla 3. Resumen de aplicaciones Vistas

3. Propuesta de una arquitectura de adquisición

La contribución a la automatización de la tarea de adquisición del conocimiento y de su relleno en modelos definidos, es un problema ya identificado en la construcción de sistemas basados en el conocimiento. Dentro del contexto de la Web Semántica esta tarea se redefine y adapta para abordar los nuevos problemas que surgen en procesos de adquisición para un repositorio de gran volumen de información no estructurada como puede ser la WWW actual. En este caso son las ontologías el formalismo elegido para el modelado del conocimiento de las aplicaciones (o agentes software). La tarea de adquisición de conocimiento, se puede dividir en dos partes:

- ***Adquisición del esquema de la ontología (ontology learning):*** En un primer momento es necesario diseñar un modelo semántico del dominio que represente los hechos de manera satisfactoria para los propósitos de las aplicaciones que lo exploten. Es una tarea compleja que tiene por objetivo construir un modelo compuesto por conceptos relacionados entre sí y descritos por atributos. Como toda tarea de diseño o modelado, es muy difícil de automatizar plenamente. Existen varias propuestas de aproximaciones metodológicas (Gomez-Perez et al 96) (Staab et al 01) (Gruninger et al 95) ofreciendo herramientas que contribuyen a la automatización de algunos pasos.
- ***Relleno del esquema definido (ontology population):*** La segunda parte de la tarea de adquisición de conocimiento consiste en instanciar el esquema de la ontología de dominio definido. El objetivo de esta tarea es localizar datos en las fuentes disponibles e insertarlos en la ontología creando nuevas instancias, rellenando o modificando las existentes o creando relaciones entre ellas. Existen varias propuestas de automatización de esta tarea en diferentes dominios usando diversas tecnologías, algunas de ellas se han visto en el capitulo anterior.

Para ambas tareas se necesitan ver sistemas automáticos y semi-automáticos cuyo objetivo es localizar información en fuentes online, normalmente textuales. La mayoría de sistemas existentes son fruto de la evolución natural de sistemas de extracción de información.

Como se ha visto en el estado del arte de los sistemas de extracción de información existen varias líneas de trabajo que hacen uso de distintas tecnologías. Desde procesamiento léxico básico basado en expresiones regulares, pasando por sistemas que procesan información de aspecto visual de las fuentes hasta complejos sistemas de procesamiento de lenguaje natural. La adecuación y el éxito de estas aproximaciones

vienen dados entre otras cosas por el tipo de fuente que tratan, el grado de acotación del dominio y las pretensiones de las aplicaciones finales como se muestra en la figura 1.

Figura 1 Adquisición y relleno del contenido para la Web Semántica

Se puede observar que para procesar fuentes estructuradas como podrían ser listados financieros en forma de tablas, manuales con disciplina estricta de presentación o artículos científicos publicados siguiendo formatos impuestos, puede ser suficiente con un procesamiento del formato o aspecto del documento con expresiones regulares. A veces el mero hecho de dos datos estén en una misma columna o fila nos puede aportar información muy valiosa sobre su significado. Algunos sistemas ofrecen funcionalidades de inferencia de expresiones regulares mediante técnicas de aprendizaje automático: reglas de procesamiento a partir de un corpus previamente anotado. En cambio si la aplicación final necesita de una comprensión más profunda del contenido o no es posible determinar una estructura estable que aporte alguna semántica, se hace preciso realizar tratamientos más complejos y profundos. Tal y como se describe en el preámbulo de este trabajo las características de las fuentes tratadas, entre las cuales destaca el grado de estructura presente, condiciona la selección de la tecnología más adecuada para su procesamiento. El término de ***estructura*** no hace referencia solamente a la posible estructura visual que la fuente pueda contener, sino que también incluye posibles estructuras desde el punto de vista de codificación de las mismas, estructuras lingüísticas o estructuras de cadenas repetitivas dentro de la fuente, entre otras.

La propuesta presentada en este trabajo define una arquitectura abierta con distintas tecnologías existentes con el propósito de realizar extracción de información para el relleno de ontologías. En la propuesta se combinan tecnologías de tratamiento léxico procesamiento de lenguaje natural, tratamiento estructural y tratamiento visual sobre las fuentes. Tras una breve enumeración de ideas clave de esta propuesta se presenta una descripción detallada de los distintos módulos propuestos.

La evolución histórica de sistemas de extracción comenzó con la aplicación de diferentes técnicas de tratamiento de información de manera individual (es decir: sistemas que únicamente aplican expresiones regulares, sistemas que realizan un análisis de lenguaje natural sin incluir otras aproximaciones, etc.). Posteriormente algunos desarrolladores optaron por combinar técnicas existentes en sistemas mixtos, y con ellos poder aprovechar las ventajas de cada uno dependiendo de las características de las fuentes: la propuesta presentada en este trabajo se engloba en esta clase de aplicaciones introduciendo además un módulo de control centralizado.

El fin de este módulo es ejercer un control sobre las distintas posibilidades de cada aproximación mediante estrategias de extracción. Para ello se introduce el concepto de operador que se entiende como una implementación de una funcionalidad primitiva de procesamiento de la fuente. Gracias a esta descomposición en operadores de las distintas funcionalidades que nos ofrecen las aproximaciones existentes es posible construir estrategias de control que combinan todas ellas en un esfuerzo de anotar semánticamente una fuente textual.

Otra particularidad de esta propuesta consiste en la importancia del módulo de relleno de ontologías de dominio. A diferencia de algunas arquitecturas que han evolucionado a partir de sistemas de extracción de información, la concepción de la arquitectura del presente sistema incluye desde el primer momento como requisito el relleno final de una ontología de dominio.

Figura 2 Aproximación mixta dirigida por estrategias

Como se explicará en este capítulo la extracción y anotación del contenido viene alimentada con información que describe el contenido susceptible de ser encontrado y anotado. Esta descripción, llamada información de adquisición, hace explicita la información sobre las distintas restricciones y relaciones que rigen sobre los datos y le permite automatizar de su identificación, extracción y posterior inserción en una ontología.

El éxito de este tipo de sistemas se mide en facilidad de la creación de las descripciones del contenido para la automatización de la extracción. El esfuerzo invertido en la formulación de las restricciones y relaciones sobre los datos se tiene que rentabilizar en la facilidad de adquisición de los mismos datos. Es decir, si es más complejo formular una descripción de las fuentes que formular directamente su contenido, entonces el sistema no es rentable. Existen dominios donde las fuentes presentan ciertos patrones repetitivos y por tanto permiten un alto grado de reusabilidad de las descripciones del contenido. No se trata tan solo de patrones sobre el diseño de los documentos, si no que se incluyen: patrones léxicos, patrones de lenguaje o sintácticos, patrones de la estructura visual y combinaciones de estos. Entre los sectores susceptibles de aplicar estas técnicas y donde se prevé una rentabilidad alta en la construcción de las descripciones se pueden incluir dominios con lenguaje controlado (ámbito jurídico, entorno médico, relaciones internacionales, etc.), dominio con esquemas muy bien definidos (documentos bancarios, tiendas online, etc.) o dominios bien acotados y definidos (dominio de las inmobiliarias, música, etc.).

En estos dominios, aunque no existan estándares de publicación y descripción de contenidos, este tipo de sistemas podrán mejorar la adquisición y formalización del conocimiento y de esta manera contribuir a facilitar la construcción de sistemas con funcionalidades avanzadas.

A continuación se presenta brevemente, pero de forma completa, el proceso de adquisición y relleno que se propone, tras la discusión inicial en que se ha situado la propuesta con respecto al estado de arte de las áreas de investigación y tecnologías existentes. Por último y usando el guión del proceso de identificación, extracción y relleno de la información se expondrá la arquitectura de adquisición.

Finalmente se indican algunos aspectos sobre un sistema automático de adquisición y relleno de conocimiento a partir de fuentes textuales ***online*** para la Web Semántica, implementando sobre la arquitectura propuesta.

3.1. Propuesta para Adquisición y Relleno

A continuación se describe el proceso general seguido en la extracción y relleno de la información para posteriormente describir cada una de sus fases. La figura 3 muestra las tres fases de alto nivel que sigue todo el proceso.

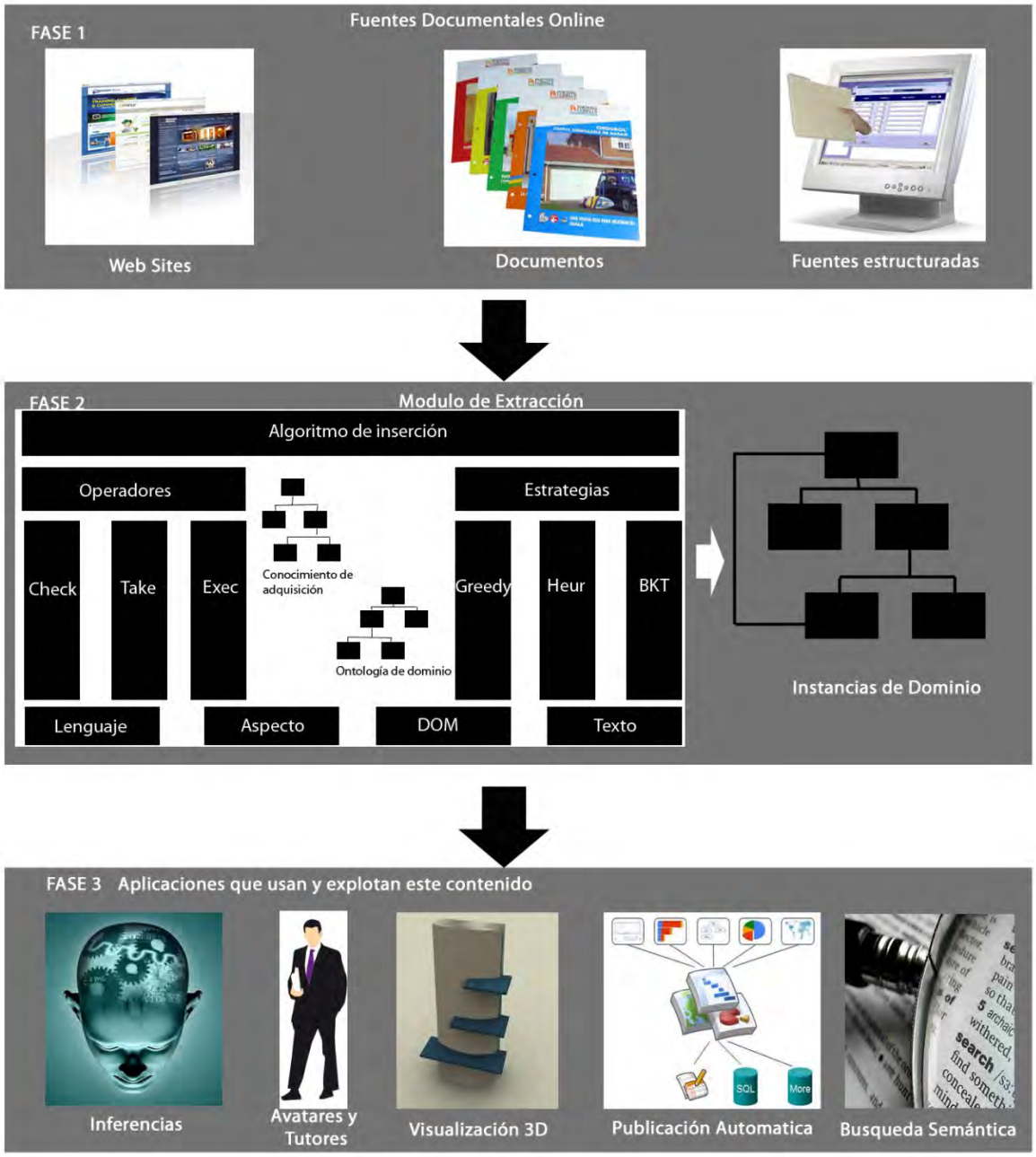


Figura 3 Sistema de adquisición propuesto

Los componentes clave de esta arquitectura se enumeran a continuación y se describen en el presente capitulo:

- **Fuentes documentales online**: situadas en la parte superior de la figura 3 incluye documentos online organizados en sitios Web (Web Sites), documentos sueltos o documentos altamente estructurados como ficheros XML, tablas, etc.
- **Interpretaciones de las fuentes preprocesadas:** según las diferentes aproximaciones que incluye el sistema:
    - Preproceso con técnicas de procesamiento de lenguaje natural: **Lenguaje.**
    - Preproceso con técnicas de visualización: **Aspecto**
    - Preproceso con técnicas de tratamiento de cadenas de texto: **Texto**
- **Modulo de extracción** en la parte central de la figura 3 compuesto por:
    - Un conjunto de **operadores** que realicen tareas de extracción clasificados en tres grupos según su efecto.
    - Un conjunto de **estrategias** capaces de dirigir el proceso de extracción construyendo y ejecutando secuencias de operadores.
    - **Información descriptiva** sobre las fuentes que representa conocimiento específico de adquisición y formalizado en una Ontología de Adquisición.
    - **Conocimiento sobre el dominio:** Ontología de dominio, incluyendo el esquema y las instancias.
- **Modulo de inserción** a cargo de ampliar y/o modificar la ontología de dominio con los datos extraídos: **Algoritmo de inserción**.
- Como el resultado del proceso se incluye la ontología de dominio rellena y ampliada con datos extraídos: **Instancias de Dominio**.
- **Aplicaciones** que usan y explotan este contenido (en la parte inferior de la figura 3).

En la figura se puede observar el flujo de datos en el proceso de recuperación y relleno de datos en una ontología de dominio, resaltando todos los componentes explicados en secciones posteriores.

Para poder extraer la información de las fuentes, estas se deben pre-procesar según las distintas interpretaciones que admite el sistema. El pre-proceso necesario se realiza bajo demanda, es decir, en el momento estrictamente necesario y es responsabilidad del módulo de **Pre-proceso de la fuente** (en la parte izquierda de la figura 4). Como se verá en este capítulo, para extraer información será necesario procesar las fuentes en alguna de

las interpretaciones de documentos conocidos. Para realizar operaciones propias del procesamiento de lenguaje natural, se proveerá la interpretación de lenguaje para operaciones de relaciones visuales, se proveerá la interpretación de aspecto, etc.

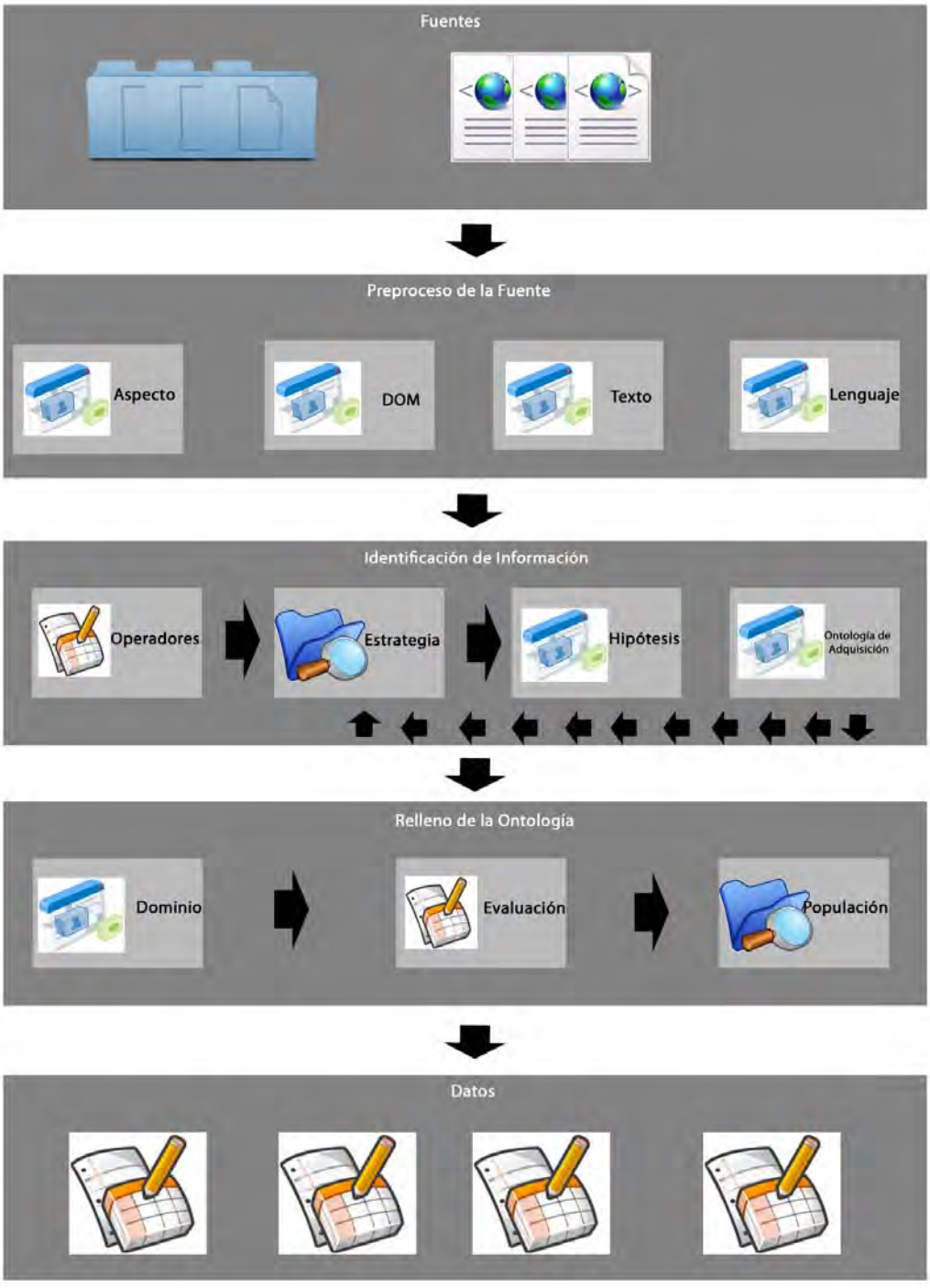


Figura 4 Arquitectura lógica y proceso de adquisición

Es el módulo llamado **Identificación de Información** (en la parte central de la figura 4) el que procesa la información de adquisición y dirige todo el proceso de extracción. El proceso de extracción se realiza aplicando distintos operadores de localización, verificación y manejo de las fuentes. Este módulo permite emplear distintas estrategias de extracción, según la caracterización de las fuentes para dirigir la ejecución de los operadores. El proceso de extracción de información y su posterior relleno en la ontología de dominio está dirigido por la ontología de adquisición. En esta ontología se describen los tipos de piezas de información del dominio que se van a recuperar de las fuentes y cómo éstas se organizan en la estructura de los documentos de la fuente. El resultado del proceso de identificación es un conjunto de hipótesis sobre la correspondencia de datos encontrados en los modelos de las fuentes. Estas hipótesis poseen un valor de plausibilidad que permite ordenarlas para la fase de relleno.

La fase de relleno, implementada en el módulo de **Relleno de la Ontología** (parte derecha de la figura 4), tiene como objetivo realizar cambios en la ontología de dominio según la información encontrada y almacenada en las distintas hipótesis. Este proceso realiza simulaciones de instancias para determinar, teniendo también en cuenta la plausibilidad asignada por el módulo anterior, cuál de las hipótesis debe insertarse. Estas simulaciones tienen en cuenta qué modificaciones deben ejecutarse para insertar el nuevo contenido en la ontología. El costo de estas modificaciones se evalúa y permite discernir entre varias hipótesis. De esta manera el comportamiento del sistema se adapta a la información que contiene en su ontología de dominio.

Cuanta más información esté disponible más precisa se hace la evaluación de las hipótesis y más precisión consigue el sistema.

3.2. Pre-proceso: Abastecimiento de Interpretaciones de Documentos

La arquitectura permite incluir interpretaciones de documentos como resultado de la fase pre-proceso. Cada interpretación de documentos responde a una necesidad de aplicar distintas técnicas en el proceso de identificación y extracción de información.

En la actualidad, la implementación de la arquitectura en el marco de este trabajo incluye cuatro interpretaciones de alto nivel:

- **Interpretaciones de texto plano**: la fuente es procesada como una cadena de caracteres sin tener en cuenta su significado, ubicación o relación con otras cadenas dentro del documento.

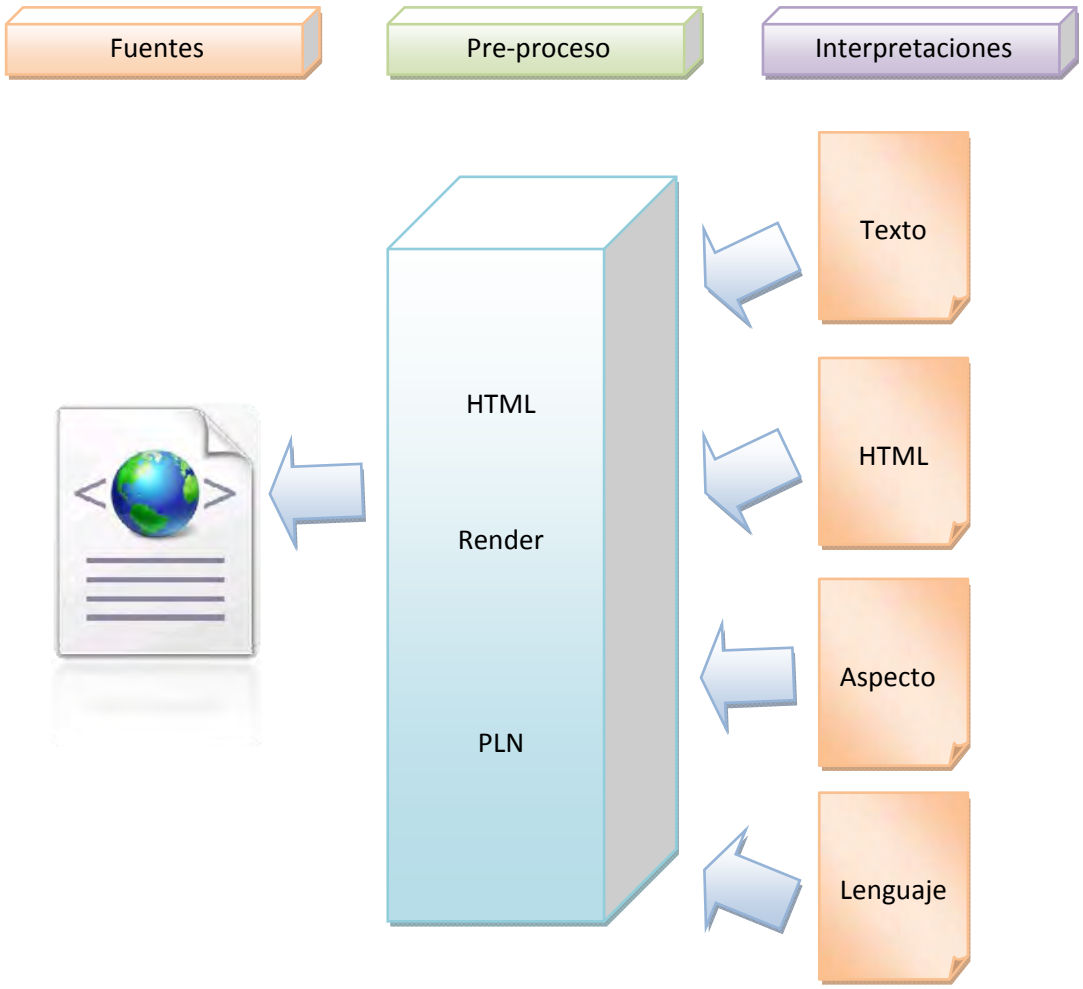


Figura 5 Pre-proceso en varias interpretaciones

- **Interpretaciones de HTML**: Permite modelar las relaciones estructurales entre los distintos datos dentro de un documento hipertexto o entre varios documentos considerados para su análisis. En esta propuesta se ha introducido para representar los estándares HTML interpretando sus distintos elementos: formularios, enlaces, etc.
- **Interpretaciones de Aspecto:** Modela aspectos visuales de los documentos. Sobre todo en aquellos lenguajes de edición dónde la composición visual se hace por los visualizadores (como son los navegadores HTML), es necesario un proceso de generación de aspecto (rendering) para captar las relaciones entre las piezas dadas

por su ubicación. Es un modelo muy útil en lenguajes tipo HTML, Postscript, LaTeX, etc.

- **Interpretaciones de Lenguaje:** Modelo orientado al uso de técnicas de procesamiento de Lenguaje natural (PLN). Permite identificar y recuperar datos de acuerdo a criterios sobre sus estructuras lingüísticas.

Esta arquitectura abierta permite añadir interpretaciones nuevas o especializar interpretaciones ya existentes con el objetivo de ofrecer operadores de más precisión y con más poder de recuperación de información. Algunos ejemplos de posibles ampliaciones o especializaciones de enlistan a continuación:

Interpretaciones de aspecto:

- Interpretación para fuentes HTML con coordenadas a nivel píxel.
- Interpretación para fuentes PDF con coordenadas de línea y columna.
- Interpretación para fuentes HTML con posiciones relativas: encima, derecha, etc.
- Interpretación para fuentes Postscript con coordenadas de línea y columna.

Interpretación de Lenguaje:

- Interpretación con segmentación a nivel de palabra con lemas.
- Interpretación con identificación de sintagmas y su núcleo (shallow parser).
- Interpretación con identificación de sintagmas y dependencias entre ellos (chunk parser).
- Interpretación con identificación de primitivas semánticas

La potencia de la presente arquitectura estriba en la combinación de las interpretaciones en las tareas de identificación de la información. De esta manera un mismo dato puede ofrecer información sobre su posición relativa en coordenadas X e Y en la interpretación de aspecto y sobre sus propiedades lingüísticas en la interpretación de lenguaje. Existen varias maneras de enlazar las interpretaciones existentes entre sí, para poder referenciar una misma parte de la fuente original en más de una interpretación. La inclusión de interpretaciones nuevas en el sistema debe tener en cuenta que granularidad[1] tiene el sistema de enlace entre interpretaciones.

[1] Granularidad consiste en la cantidad de cómputo con relación a la comunicación. En Granularidad Fina o Fine-grained, las tareas individuales son relativamente pequeñas en término de tiempo de ejecución. La comunicación entre los procesadores es frecuente. En cambio en Granularidad gruesa o Coarse-grained, la comunicación entre los procesadores es poco frecuente y se realiza después de largos periodos de ejecución.

Con granularidad se entiende la pieza de la información más pequeña referenciable (una palabra, un lexema, un párrafo, etc.). Por ejemplo: si la interpretación de lenguaje natural ofrece funcionalidades a nivel de palabra, es necesario que todas las demás interpretaciones puedan aportar información sobre la palabra dada: sus coordenadas espaciales, si encaja en una expresión regular o si es parte de un enlace a otro documento. Sin embargo si el nivel mínimo necesario es el de una línea, la granularidad de las interpretaciones decrece. A continuación se muestran dos posibles soluciones para enlazar las distintas interpretaciones entre sí:

- **Enlace de Interpretaciones por posición Absoluta**: Para permitir que el modulo central de identificación de piezas de información pueda consultar las propiedades de las distintas piezas en cualquiera de las interpretaciones ofrecidas, éstas deben de disponer de un sistema de referencias común.

  Suponiendo que la fuente no se altera durante el proceso de extracción se pueden usar como referencias la posición absoluta de cada pieza dentro del texto fuente. Una solución básica de este problema es usar una referencia, URL por ejemplo, al documento tratado (ya que puede haber más de uno), la distancia de la pieza desde el comienzo del documento (offset) y su longitud en número de caracteres. Usando esta técnica cada interpretación resultado del pre-proceso debe ofrecer funcionalidades de consulta de acuerdo al documento tratado, la distancia del origen y la longitud del dato. Una palabra situada en el documento S con distancia D y longitud L, puede ser parte de un enlace a otro documento en la interpretación DOM[2], puede responder a un Nombre Propio en la interpretación de procesamiento de lenguaje natural y estar situada en la misma línea visual que otra pieza en la interpretación de aspecto.

- **Enlace de Interpretaciones Usando un Modelo Común**: En algunos casos no es posible garantizar la estabilidad de las fuentes durante el proceso o hasta puede ser objetivo de la aplicación que cambien (inserción de etiquetas o nuevo texto). Un sistema de referencias como el descrito anteriormente no garantizaría su

[2] El Document Object Model (una traducción al español no literal, pero apropiada, podría ser Modelo en Objetos para la representación de Documentos o también Modelo de Objetos del Documento), abreviado DOM, es esencialmente una interfaz de programación de aplicaciones que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos. A través del DOM, los programas pueden acceder y modificar el contenido, estructura y estilo de los documentos HTML y XML, que es para lo que se diseñó principalmente. El responsable del DOM es el consorcio W3C.

coherencia a lo largo de todas las interpretaciones. En este caso se propone la utilización de un modelo central que sirva de referencia común para las interpretaciones en cuanto a las piezas de información. En este caso hay que prestar mucha atención a la granularidad mínima exigida entre todas las interpretaciones. En uso del estándar DOM (de las siglas inglesas Document Object Model) (DOM) puede ser adecuado para ser utilizado como nexo común entre las interpretaciones para la referencia de piezas de información.

3.2.1. Interpretación de Texto Plano

La interpretación de texto plano, en la más sencilla de las interpretaciones presentadas y permite almacenar el documento fuente tal y como fue escrito. Se trata la fuente como una cadena de caracteres continua. Los operadores que necesitan de esta interpretación son los que trabajan con expresiones regulares.

La complejidad de las expresiones regulares viene dada por el usuario que diseña la ontología de adquisición. El sistema dispone de algunas expresiones predefinidas para la detección de algunos tipos básicos de datos.

- Cadenas que representan números
- Cadenas que representan direcciones de correo.
- Cadenas que representan direcciones de internet

Algunas estructuras más complejas, como las fechas o nombres propios están situadas más en el ámbito del modelo de procesamiento de lenguaje natural, ya que una expresión regular puede ser un formalismo no suficiente o puede resultar demasiado laborioso para su detección.

A cambio de sus limitadas capacidades de detección, la eficiencia es la mejor de las interpretaciones disponibles. No exige grandes procesamientos previos y permite de manera más sencilla identificar o recabar propiedades sobre algunas piezas básicas de información.

3.2.2. Interpretación en HTML

La interpretación de HTML derivada del modelo de documento-objeto (DOM) es un interfaz de programa para el acceso o documentos. DOM, en su primera versión, recomendada por el organismo W3C en octubre del 1998, define la estructura lógica de documentos XML y HTML. Esta estructura lógica, extendida en posteriores publicaciones

del W3C, representa un documento como un conjunto de árboles almacenado en una estructura de objetos (llamados nodos, raíces, hojas, etc.)

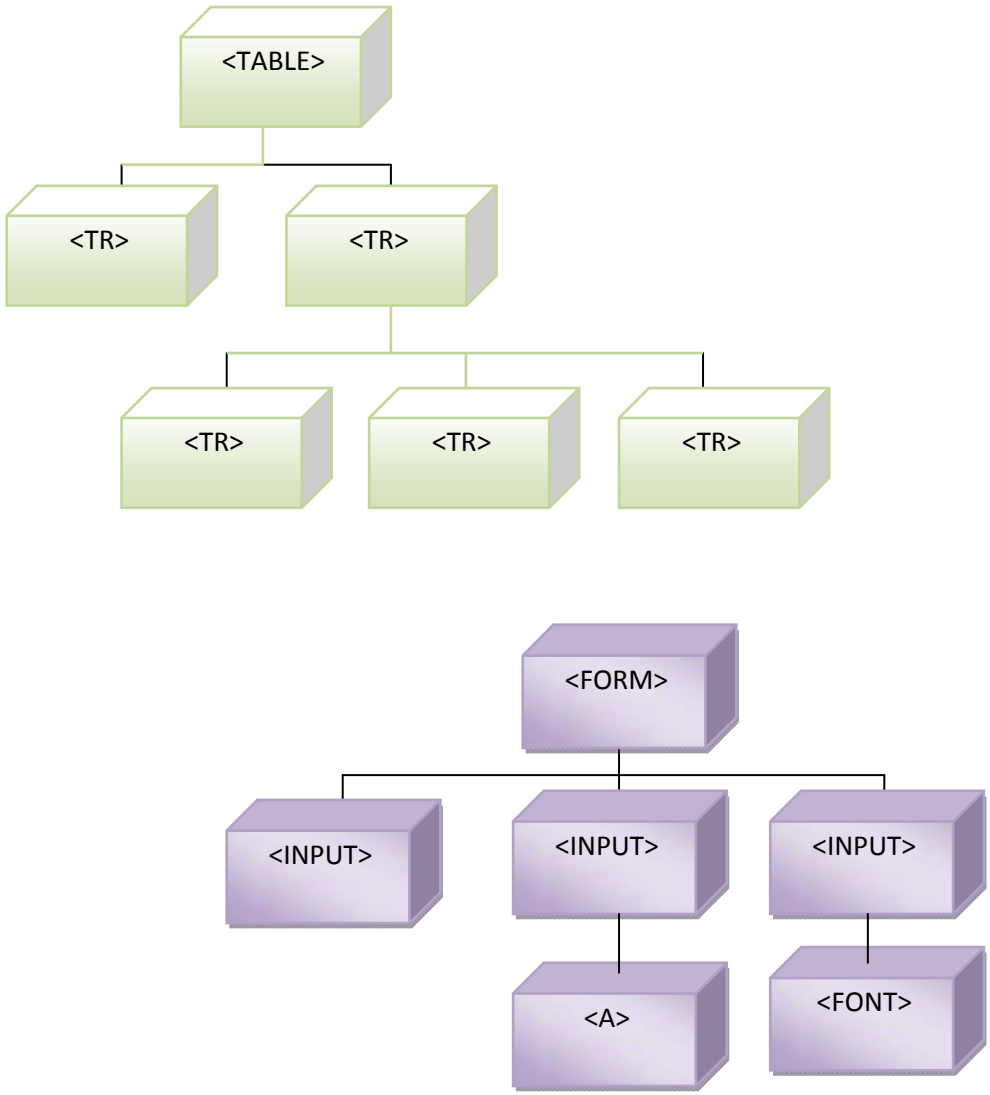


Figura 6 Estructuras arbóreas de DOM para HTML

Este modelo es válido para documentos derivados del XML, como pueden ser los documentos online HTML. En el sistema propuesto esta interpretación se usa para modelar documentos dentro de un sitio Web y para modelar objetos propios del HTML dentro de cada documento.

La interpretación del modelo documento-objeto es esencial para documentos HTML ya que permite la ejecución de operadores que manejan directamente elementos propios de este formato. Así mismo existe mucha semántica codificada en los objetos HTML. Un enlace nos lleva a otro documento, un formulario nos permite invocar funcionalidades contra el servidor Web y obtener resultados nuevos HTML.

3.2.3. Interpretación de Aspecto

El resultado de procesamiento de aspecto (*Layout*) tiene como objetivo formar una representación visual del documento tratado. Esta representación debe reflejar el documento tal y como lo vería el usuario humano. En documentos HTML esta labor la realizan los navegadores Web cuando procesan las fuentes de los documentos y los traducen a una presentación en pantalla. En caso de un sistema de extracción de información la presentación final no es para el usuario humano, si no es para un software que debe conocer y tratar computacionalmente esta representación.

El objetivo de esta interpretación es ofrecer una representación espacial, en dos dimensiones, del contenido presente en los documentos tratados. Para ello debe establecer un sistema relativo de coordenadas y a cada dato de interés en el documento asignarle el par de valores en las coordenadas que permita ubicarlo. Este sistema de coordenadas permitirá que sea posible comprobar, por ejemplo, si dos datos, o piezas, están en una misma línea o columna visual.

En el formato de fuentes HTML, a este proceso se le conoce comúnmente por el nombre de inglés *rendering* (traducción a formato visual) de una fuente. Existen varias implementaciones software para esta tarea, en su mayoría fabricadas por empresas propietarias de navegadores (Microsoft con Internet Explorer, Mozilla, etc.). El problema es complejo, ya que el contenido de los documentos HTML es muy variado y en muy pocos casos se limita al contenido puro especificado por la recomendación del W3C. Hoy en día no es extraño encontrar un documento online que contenga algunas partes en *javascript*, animaciones *flash*, piezas de software descargadas en un *applet* o *java* que no tenga alguna parte de su contenido en formato multimedia.

La interpretación de aspecto debe proporcionar un sistema de ubicación de distintos componentes de documentos HTML en unas coordenadas relativas. En el presente sistema se pretende implementar dos derivaciones del modelo de aspecto.

3.2.3.1. Interpretación de Aspecto con Coordenadas Lógicas

Dado que la mayoría de documentos online basados en el lenguaje HTML usan solamente una pocas estructuras para ubicar su contenido en una página, es posible obtener una

posición relativa evitando tener que calcular la apariencia visual final del documento en detalle.

Si los operadores especificados en la información de adquisición solamente necesitan verificar ubicaciones relativas de dos piezas (por ejemplo: un enlace está por encima de una etiqueta) no es preciso realizar el *rendering* completo del documento para obtener las coordenadas físicas de cada elemento. En este caso es suficiente con procesar aquellas etiquetas del HTML que desplazan los elementos en posición vertical (por ejemplo: nueva línea
, nuevo párrafo, <P>, nueva línea en tabla <TR>, lista <LI>, etc.) para determinar que pieza está ubicada por encima de la otra. Esta interpretación ofrece un número limitado de operadores de ordenación espacial que hace referencia aposiciones relativas como:

- ABOVE (encima): en la dimensión vertical del documento, la pieza A está situada por encima de la pieza B (no necesariamente es la misma columna).
- UNDER (debajo): pieza A situada debajo de la pieza B.
- RIGHT (derecha): pieza A situada a la derecha de la pieza B (no necesariamente en la misma línea).
- LEFT (izquierda): pieza A situada a la izquierda de la pieza B.

La ventaja que ofrece es una mayor eficiencia y menor costo computacional, que podría ser determinante en algunos dominios con fuertes requisitos sobre el tiempo de proceso.

3.2.3.2. Interpretación de Aspecto con Coordenadas Físicas

Esta interpretación realiza la tarea de *rendering* completa, tal y como lo haría un navegador Web. La mayoría de los algoritmos de construcción de la representación visual de un documento HTML se basa en procesadores independientes de cada etiqueta de HTML (*Tag Processors*) que construyen de manera incremental el aspecto final de la página. En este proceso se tienen en cuenta atributos de las etiquetas, especialmente aquellos referentes a las dimensiones de los objetos HTML (*width, height, size, etc.).*

El objetivo de esta interpretación es que cada pieza de información disponga de unas coordenadas asignadas con precisión de un pixel. Esto permite invocar operadores que comprueben posicionamiento mucho más precisos:

- IN ROW (en línea): dos piezas tienen la misma (o muy similar) la coordenada Y
- IN COLUMN (en columna): dos piezas tienen la misma (o muy similar) la coordenada X.

Es un hecho conocido que el mantenimiento de software de extracción de información (*wrappers*) a partir de páginas Web es una tarea muy costosa, comparable, incluso con el costo de la construcción del propio software. Este software es muy sensible a pequeños cambios en la estructura de la fuente. La manera de inclusión de una nueva línea o columna en una página HTML puede hacer fracasar la ejecución de la extracción y obligar a reprogramar el proceso. Es muy habitual que esto suceda con sitios Web que incluyen publicidad online, esto cambia en cada acceso y podría afectar a la estructura de la página. En algunos dominios, los proveedores de contenidos adoptan incluso estrategias de anti-agregación. Es decir, hacen lo posible para que los distintos *wrappers* fracasen en su propósito de extraer el contenido de manera automática y ofrecerlo en otro sitio. Para ello introducen cambios en los documentos online que afectan la propia estructura del documento pero no afecta, o afecta limitadamente, su apariencia final.

En estos casos el uso de interpretaciones de aspecto que trabajan con los datos desde el punto de vista del usuario puede introducir mayor robustez en el software de extracción haciéndolo más resistente a cambios.

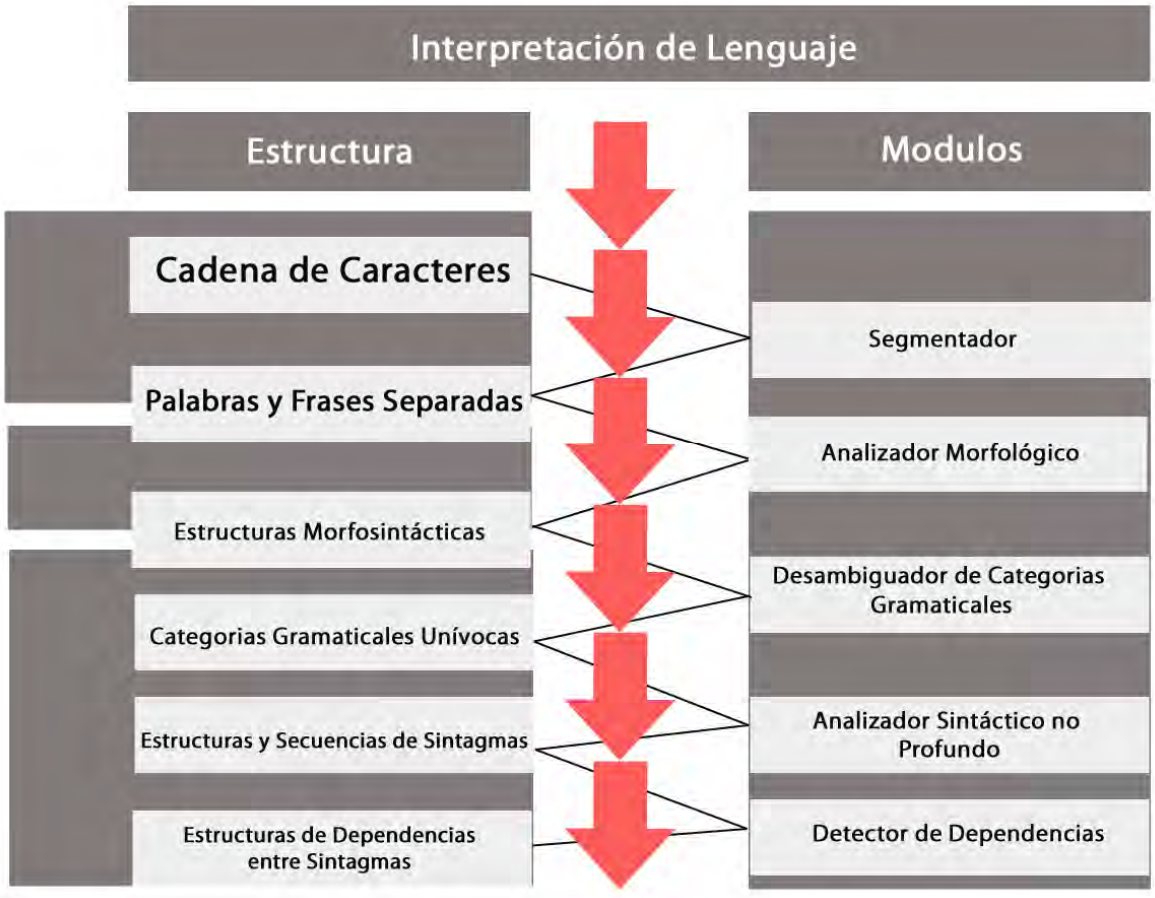


Figura 7 Estructura interna del proceso de interpretación de lenguaje

3.2.4. Interpretación de Lenguaje

Otra de las interpretaciones de procesamiento de la fuente es la de lenguaje que permite interrogar sobre propiedades lingüísticas propias de los sistemas de procesamiento de lenguaje natural. El objetivo de esta interpretación es ofrecer funcionalidades que permitan recuperar datos de acuerdo a criterios lingüísticos a nivel de palabra, grupo de palabra, sintagma, frase o incluso párrafo.

La estructura interna del proceso que produce una interpretación de lenguaje el cual se basa en un modelo clásico de aplicaciones de procesamiento de lenguaje natural que procesan fuentes textuales pero que se detiene en la obtención de información sintáctica a nivel de sintagmas[3] con algunas relaciones semánticas. Lo que tradicionalmente se entiende por procesamiento semántico, que sería el siguiente paso, se obvia para ser implementado con la estrategia del modulo de identificación de información, para que se combine la información, no solo de interpretación de lenguaje, sino también de las interpretaciones de aspecto, HTML o texto plano.

A continuación se describen los propósitos y las funcionalidades de los módulos involucrados en la obtención de la interpretación del lenguaje.

3.2.4.1. Segmentación

Tiene por objetivo construir una secuencia de palabras separadas a partir de la fuente textual. Esto no siempre es una tarea trivial ya que debe separar los caracteres de control de la fuente y reconstruir el texto original. Dependiendo de la opción tomada en la solución para el enlace de los distintos modelos el Segmentador deberá trabajar o con fuentes codificadas (HTML, LaTeX, etc.) en el caso de referencia por offset o con el modelo DOM.

3.2.4.2. Análisis Morfológico

[3] El **sintagma** (del griego σύνταγμα *syntagma* 'arreglo, coordinación, agrupación -ordenada-') es un tipo de constituyente sintáctico formado por un grupo de palabras que forman otros sub-constituyentes, al menos uno de los cuales es un núcleo sintáctico. Las propiedades combinatorias de un sintagma se derivan de las propiedades de su núcleo sintáctico, este hecho se parafrasea diciendo que "un sintagma se caracteriza por ser la proyección máxima de un núcleo". Por su parte el **núcleo sintáctico** es la palabra que da sus características básicas a un sintagma y es por tanto el constituyente más importante o de mayor jerarquía que se encuentra en su interior. Su estructura fundamental es recogida en la llamada teoría de la X' (X-barra)

El análisis morfológico, como se ha comentado en el estado del arte, nos permite obtener información morfo-sintáctica sobre cada palabra identificada por el Segmentador de texto. Entre esta información se incluyen los lemas posibles de la palabra, con su categoría gramatical *(POS o Part of speech)* correspondiente. Para cada par lema y categoría se enumeran los valores morfo-sintácticos.

Muchas de las palabras pueden producir más de un resultado y generar así una ambigüedad en el procesamiento posterior que debe ser tratada. Existen varias estrategias en su resolución:

- La utilización de desambiguadores de categorías gramaticales, pueden seguir las siguientes aproximaciones entre otras:
    - ***Desambigüación por frecuencia de aparición***: Algunos analizadores morfológicos llevan asociada una frecuencia de uso de las distintas categorías gramaticales para cada lema. De esta manera se podría discernir que la palabra lista se usa en su categoría de nombre un tanto por ciento de las veces. Este método es muy dependiente del corpus que se utilice para el cálculo de las frecuencias y no garantiza resultados correctos.
    - ***Desambigüación por contexto***. Módulos especiales de desambigüación que generalmente usan técnicas estadísticas para calcular la probabilidad de que una palabra actúe con una determinada categoría gramatical según las palabras que la rodean.
- Mantenimiento de todas las posibilidades para análisis posteriores. En algunos sistemas, como la propuesta de implementación en este trabajo, la desambigüación de la categoría gramatical se deja para el análisis sintáctico no profundo. Esto aumenta el costo computacional del sistema pero ofrece más garantía de la corrección de los resultados.

3.2.4.3. Análisis Sintáctico

En análisis sintáctico propuesto en este sistema es un análisis no profundo que identifica los diferentes sintagmas dentro de cada frase del texto de entrada. La selección de un análisis no profundo frente a uno completo se debe a varias razones:

- No podemos asumir que la mayoría de los textos online estén escritos con frases completas. Sobre todo en fuentes generadas de manera dinámica es frecuente el uso de palabras sueltas o construcciones cortas, que no alcanzan la forma de una sentencia gramatical completa.

- Desde el punto de vista computacional, el análisis no profundo es menos costoso que los análisis completos que construyen árboles sintácticos para frases bien formadas.

Gracias por el procesamiento morfológico realizado en el paso anterior, es posible aplicar distintos paradigmas de análisis para construir los posibles sintagmas presentes en el texto.

Las operaciones que proporcionan un analizador sintáctico no profundo trabajan a nivel de sintagma. Algunas de ellas se enumeran a continuación:

- Recuperación de todos los sintagmas nominales cuyo nombre principal se a "X".
- Recuperar todos los sintagmas verbales cuyo verbo principal sea sinónimo de "Y".
- Recuperar todos los sintagmas adverbiales de lugar, tiempo o cantidad.

El uso de analizadores no profundos permite analizar frases incompletas, pero no ofrecen información completa sobre algunas relaciones y dependencias entre los sintagmas que podrían ser muy útiles en la interpretación semántica. Es por eso que en el diseño propuesto de este sistema se recomienda la inclusión de un analizador de dependencias. Este analizador, a veces denominado *chunk parser* (analizador de partes) realiza un análisis sintáctico no profundo y es capaz de identificar algunas relaciones semánticas entre los distintos sintagmas.

3.2.4.4. Análisis Semántico

En el sistema propuesto se entiende el análisis semántico como el proceso de vincular partes del texto de entrada con el modelo semántico expresado en la ontología de dominio. Este proceso se realiza fuera de la interpretación de lenguaje, en el módulo de relleno de ontologías que procesa las hipótesis generadas por el modulo de identificación de información. Las hipótesis no solo tienen en cuenta resultados del procesamiento de lenguaje natural, si no que aglutinan información de todos las posibles interpretaciones presentes en el sistema.

Sin embargo, existen partes del análisis semántico que pueden ayudar al sistema en su tarea de vinculación con la ontología de dominio. Estas partes se implementan en la interpretación de lenguaje al final del procesamiento de dependencias. Se incluyen palabras vinculadas mediante alguna relación semántica simple, como pueden ser los sinónimos, antónimos, merónimos o hiperónimos. La inclusión de estas relaciones semánticas mejora la cobertura del sistema pero introduce a su vez más complejidad en el mismo ya que son relaciones relativas a las acepciones de cada palabra que no están unívocamente determinadas.

3.3. Modulo de identificación de información

El modulo de identificación de información (situado en el centro de la figura 4) es el modulo de control del sistema propuesto que dirige todo el proceso de extracción y relleno de datos. El objetivo de este módulo es recuperar y validar las distintas piezas de información buscadas y construir un conjunto de hipótesis de acuerdo a las restricciones impuestas por la información descriptiva de las fuentes. Las fuentes, las posibles fuentes de información a extraer y algunas restricciones sobre ellas vienen descritas en una ontología de adquisición que sirve de base de conocimiento de este módulo.

Este proceso se ejecuta siguiendo estrategias de adquisición que consisten en llamadas a diferentes operadores capaces de trabajar sobre las interpretaciones de documentos proporcionados por el modulo anterior. Existen varias secuencias de ejecución de operadores que se encuentran, obtienen y validan las piezas de información en las fuentes. La composición exacta de las secuencias viene dada por la estrategia que el módulo de identificación este ejecutando. El objetivo de toda estrategia es construir un conjunto de hipótesis sobre la asignación de distintas partes de las fuentes a piezas de información buscadas.

Ejemplo: una estrategia cuyo principal objetivo sea la rapidez de la extracción prioriza la adquisición de información frente a la validación de relaciones entre piezas y empleara el uso de heurísticos que reduzcan la carga computacional del tratamiento de ambigüedades. Por otra parte, una estrategia orientada hacia la calidad de los resultados, priorizara las validaciones de relaciones entre los datos recién adquiridos para poder tomar decisiones de adquisiciones alternativas si es necesario.

En este capítulo se hace una descripción e la base de conocimientos del módulo expresado de acuerdo a una ontología de adquisición, se elabora una clasificación de los operadores de acuerdo a su efecto sobre los datos de las fuentes y se proponen varias estrategias de secuenciación de estos para la creación de conjuntos de hipótesis.

3.3.1. Ontología de Adquisición

El sistema dispone a priori de la descripción de tipo de información que podrá encontrar en las fuentes. La información descriptiva añade características sobre cada elemento de la ontología de dominio susceptible de ser localizado en los documentos fuentes, así como información sobre la estructura de los documentos disponibles.

La información contenida en la ontología de adquisición viene principalmente de dos fuentes:

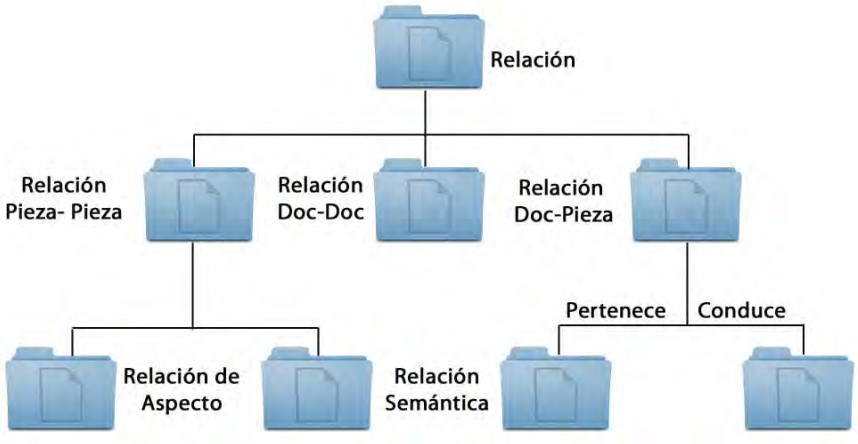
- **Esquema del modelo de dominio**: Se conocen los conceptos, atributos, relaciones y axiomas del modelo de dominio que se rellenará con el sistema. Esto permite dirigir la búsqueda y extracción de información por piezas de información que se deseen encontrar.
- **Descripción del dominio**: Además del propio esquema del dominio de partida se cuenta con información adicional. Esta información no suele formar parte del dominio, pero está implícita (en algunas aéreas se le denomina: *background Knowledge*). Se incluyen las descripciones de los tipos de datos básicos (como identificar valores numéricos, cuales son los dominios acotados de algunos atributos, que forma tiene los valores, etc.), restricciones sobre los valores de los atributos (números enteros positivos entre 0 y 110, etc.), información sobre relaciones entre los distintos conceptos, etc.

La ontología de adquisición es una extensión de la ontología de dominio donde se añade información necesaria para el proceso de extracción y relleno. De esta manera la ontología de dominio no sufre modificaciones y es reutilizable para otros propósitos. De la misma manera la ontología de adquisición se puede aplicar sobre otras ontologías del mismo dominio con modificaciones mínimas.

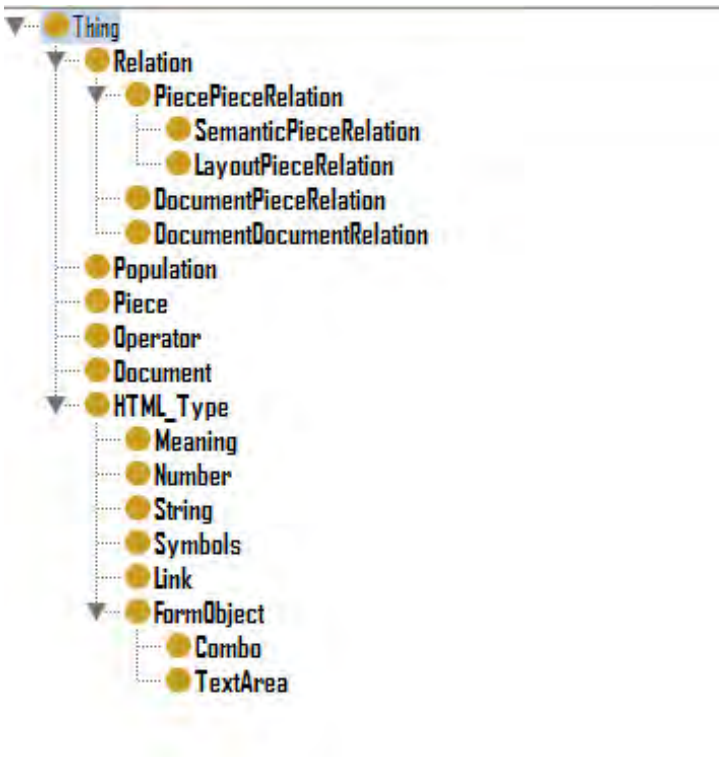
El reto de este tipo de sistemas que cuenten con un conocimiento previo descriptivo de fuentes, es poder utilizar estas descripciones con facilidad en otras fuentes de mismo dominio. La construcción de las descripciones es costosa y debe rentabilizarse en su aplicación a más de una fuente. No todo dominio es propenso de una descripción fácil y reutilizable. Depende de lo acotado que esta, tipos de estructuras (visuales, de lenguaje, de estructura, etc.) que contiene. El reto es identificarlos y obtener descripciones sectoriales del dominio para aplicar con facilidad las técnicas de extracción.

Así mismo en la presente propuesta de arquitectura se ha previsto el uso de las instancias del modelo de dominio como información muy útil para futuras identificaciones o extracciones. El sistema presentado como proposición permite aprovechar tanto las instancias existentes previas al comienzo de la tarea de extracción, como las instancias adquiridas durante la misma para reforzar o rechazar la hipótesis que se generan en el modulo central. Esto permite que a medida que el sistema vaya extrayendo información, vaya mejorando la eficiencia del mismo proceso gracias a la información adquirida.

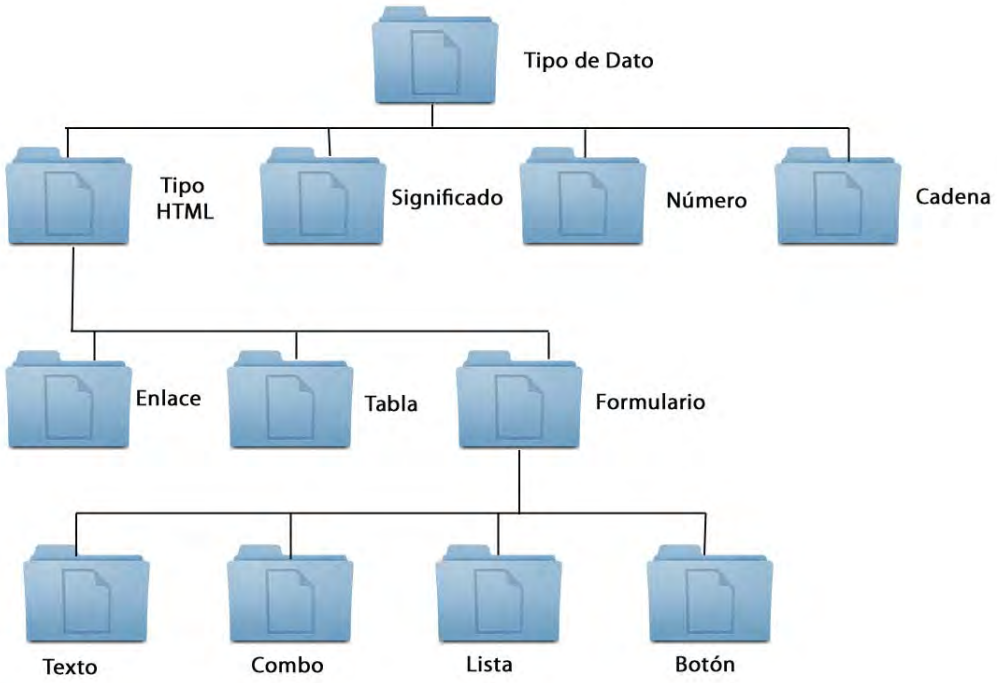
A continuación se describe la información de adquisición en forma de ontología que sirve de base de conocimiento para el módulo central de identificación de información.



a. Información de Adquisición



b. Información de adquisición



c. Información de adquisición

[Figura 8] Información de conocimiento en forma de Ontología

3.3.1.1. Pieza de información

El elemento central de la ontología de adquisición es la pieza de información. Cada atributo de la ontología de dominio que es susceptible de ser recuperado en las fuentes, debe tener definida su pieza correspondiente en la ontología de adquisición.

Cada pieza se describe por lo siguientes atributos:

- **nombre:** Nombre de la pieza.
- **valor:** Valor de la pieza en la fuente. Es la información que debe asignar el proceso de identificación de información.
- **tipo:** Define el tipo de dato que la pieza admite para sus valores. El tipo de dato es otro concepto de la ontología de adquisición que se describe a continuación.
- **inserción:** Alberga información necesaria para instanciar la pieza en la ontología de dominio usada por el modulo de relleno. A su vez es otro concepto de la ontología de adquisición descrito más adelante.
- **operador:** Contiene información sobre los operadores que pueden actuar sobre la pieza con distintos efectos (ejecución, comprobación o recuperación). El operador a su vez es un concepto de la ontología.

- **fuente externa:** Permite apuntar a una URL externa donde encontrar los posibles valores de la pieza en un fichero formateado para tal propósito. Evita tener que enumerar grandes dominios para ciertas piezas.

3.3.1.2. Documentos

Un documento se entiende como una entidad procesable por algunos de las interpretaciones definidas en el sistema y al mismo tiempo que como un contenedor de piezas de información. En una fuente hipertexto, como las tratadas en esta propuesta, los documentos son equivalentes a las páginas online. Las piezas se localizan dentro de un documento usando operadores de recuperación sobre la interpretación de documentos que proceda. Para cambiar de documentos es necesario invocar operadores de ejecución, en concreto: navegación. Un documento se caracteriza por los siguientes atributos:

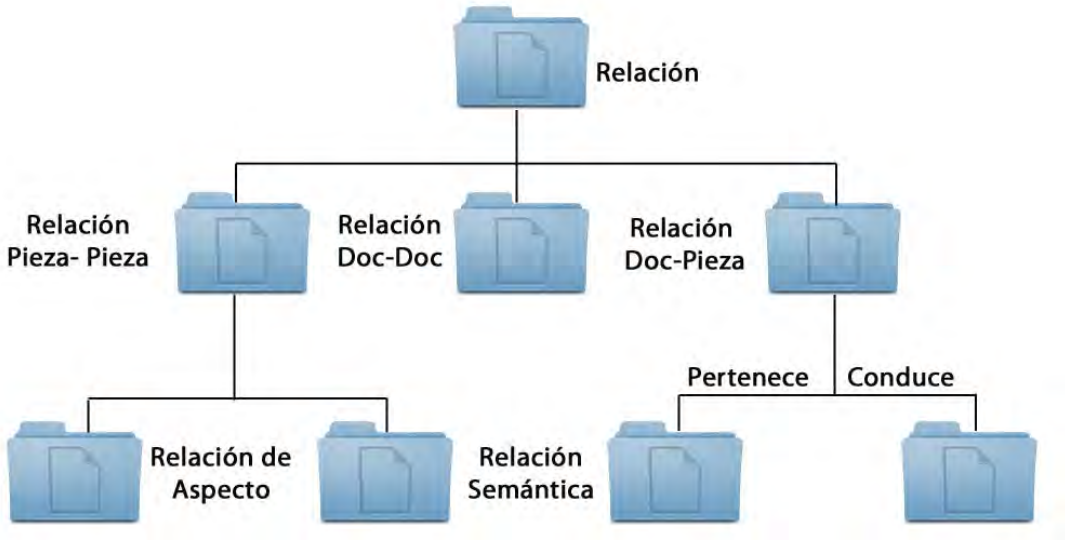
- **nombre:** Nombre del documento.
- **URL:** Ubicación del documento.

Ejemplo:

Documento descriptivo de un país en la Web de información geopolítica. Se indica su nombre, la URL suele ser desconocida y mediante relaciones de pertenencia (descritas más adelante) se vinculan a él las piezas de información que puede contener: número de habitantes, PIB, edad media y muchas más.

3.3.1.3. Relaciones

En la ontología de adquisición existen conceptos específicos para modelar relaciones entre las piezas y los documentos existentes. Se describen algunas de ellas:



[Figura 9] Jerarquía de relaciones

Relaciones entre piezas

Las relaciones entre piezas son conceptos que modelan las dependencias entre dos piezas en un documento. Las relaciones posibles, relaciones contempladas se dividen en dos subcategorias:

- **Relaciones de aspecto:** indican a nivel visual que posiciones relativas pueden adoptar las piezas.
- **Relaciones semánticas:** indican a nivel de significado las relaciones entre dos piezas.

Cada una de ellas se describe a continuación:

Relaciones de aspecto

Una relación de aspecto entre dos piezas define su posición relativa en la visualización del documento. Especialmente en documentos de la WWW el archivo fuente no determina la aparición final del documento directamente y es necesario procesar las marcas HTML para poder ubicar las distintas partes de la página. Algunos sistemas de extracción de información desde fuentes online tienen en cuenta solamente el código fuente con lo cual los cambios pequeños en la definición de la página, aun sin repercusión en su visualización final imposibilitan la extracción automática. Sin embargo, si el sistema realiza la misma tarea que el visualizador final de la página como un navegador, y toma referencias visuales, estos pequeños cambios no le afectan.

Una relación de aspecto se caracteriza con los siguientes atributos:

- **nombre:** Nombre de la relación de aspecto.
- **pieza origen:** Una de las piezas que participa en la relación. En caso de relaciones unidireccionales esta es la pieza origen.
- **pieza destino:** Pieza que participa en la relación.
- **documento:** Documento en el cual ocurre la relación.
- **tipo:** Determina la relación de aspecto concreta. De momento se contemplan las siguientes posibilidades:
    - **en línea:** dos piezas están visualmente en la misma línea (coordenadas de altura similares).
    - **en columna:** dos piezas están visualmente en la misma columna.
    - **encima:** una pieza está encima de otra
    - **derecha:** una pieza está a la derecha de otra.
    - **etc.**
- **grado:** El grado determina la obligatoriedad de la ocurrencia de la relación entre dos piezas determinadas. Admite dos posibles valores:
    - **opcional:** La relación no tiene por qué darse entre las os piezas. Su apariencia o no tiene efecto sobre la plausibilidad de la hipótesis creada por el modulo de identificación.
    - **obligatoria:** Significa que la relación debe ocurrir. En caso contrario se invalida la hipótesis en construcción.
- **operador:** Lista de operadores que puede comprobar la existencia o no de la relación. Normalmente estos operadores tienen como pre-requisito la disponibilidad de la interpretación de documentos de aspecto (*layout*) para conocer las coordenadas bidimensionales de las piezas.
- **inserción:** La existencia o no de una relación puede tener efectos sobre la ontología de dominio. Si dos piezas tienen relación de aspecto es probable que estén relacionadas en la ontología de dominio, con lo cual se debe de añadir una instancia o relacionar dos ya existentes.

[Figura 10] Ejemplo de una relación visual: EN LINEA, entre dos piezas.

Relaciones semánticas

Las relaciones semánticas permiten relacionar piezas de información a través del significado del contenido. Las relaciones semánticas se verifican usando técnicas de procesamiento de lenguaje natural para determinar qué relación existe entre dos piezas. Todos los operadores de comprobación tienen como pre-requisito la disponibilidad de interpretación de documento lenguaje que contiene el resultado de un análisis sintáctico no profundo de la fuente (*chunk parsing*) identificando y relacionando sintagmas presentes.

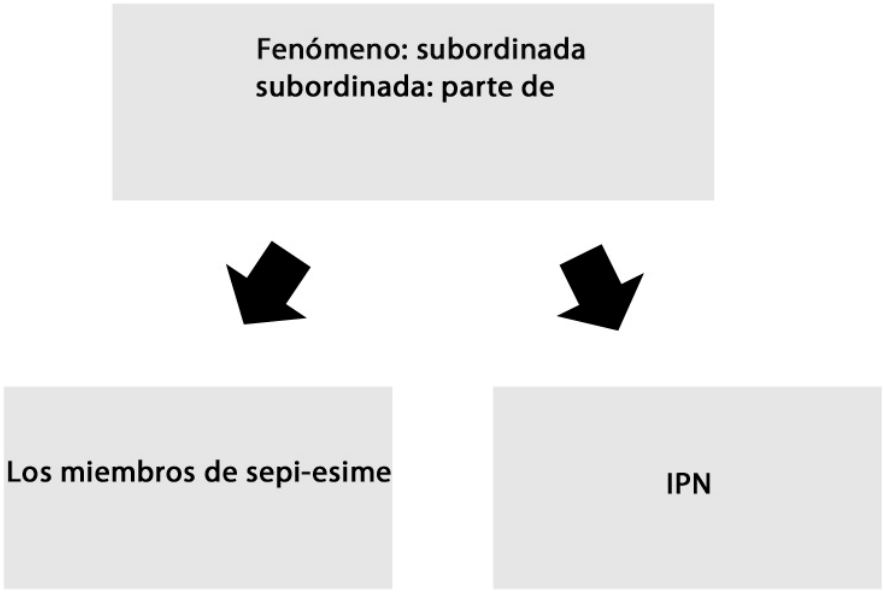
Una relación semántica se caracteriza con los siguientes atributos:

- **nombre:** Nombre de la relación de aspecto.
- **pieza origen.** Una de las piezas que participa en la relación.
- **pieza destino:** Pieza que participa en la relación.
- **documento:** Documento en el cual ocurre la relación.
- **tipo de relación:** Permite identificar el tipo de relación semántica que existe entre dos piezas. En un principio se proponen las siguientes, que a su vez son soportadas por el analizador sintáctico: aposición, subordinada, coordinada, enlace mediante el sintagma.
- **significado:** En el caso de una relación de enlace mediante sintagma determina que sintagma debe existir entre las dos piezas.
- **grado:** Al igual que en la relación de aspecto, el grado determina la obligatoriedad de la ocurrencia de la relación entre dos piezas determinadas.

Admite dos posibles valores:

- **opcional:** la relación puede darse entre las dos piezas.
- **obligatoria:** Significa que la relación debe ocurrir.

- **operador:** Lista de operadores que puede comprobar la existencia o no de la relación.
- **inserción:** Acciones de relleno.



[Figura 11] Ejemplo de una relación semántica de “parte_de”

Relaciones entre Documentos y Piezas

Existen diversas posibilidades en las que las piezas de información se relacionen con los documentos. En esta ontología se han modelado dos tipos: que una pieza pertenezca a un documento o que, una pieza permita transitar entre dos documentos.

Relación de pertenencia

Sus atributos se describen a continuación:

- **documento:** Representa el documento que alberga la relación.
- **pieza:** Pieza que pertenece al documento.
- **cardinalidad:** Define cuantas ocurrencias de la pieza puede haber en el documento. Se indican tanto la cota inferior como superior.
- **operador:** Lista de operadores que puede comprobar la existencia o no de la relación.

- **Inserción:** Acciones de relleno.

Relación de transición

Es una relación propia de fuentes hipertextuales. Una relación de transición indica que una pieza de información enlaza con otro documento. Esto se ha modelado así para poder recoger que enlaces HTML, formularios, listas desplegables, etc. Son, a menudo, puerta a otros documentos de la misma fuente. Este tipo de enlaces son generalizables en ciertos dominios (el enlace sobre el número de cuenta en un banco nos lleva al documento de movimientos, el enlace sobre un nombre de país nos lleva a la información detallada sobre, etc.).

Sus atributos se describen a continuación:

- **documento anfitrión:** Representa el documento que alberga la relación.
- **documento destino:** Documento resultado de transitar por la pieza**.**
- **pieza:** Pieza que pertenece al documento anfitrión.
- **operador:** Lista de operadores que puede comprobar la existencia o no de la relación.
- **inserción:** Acciones de relleno.

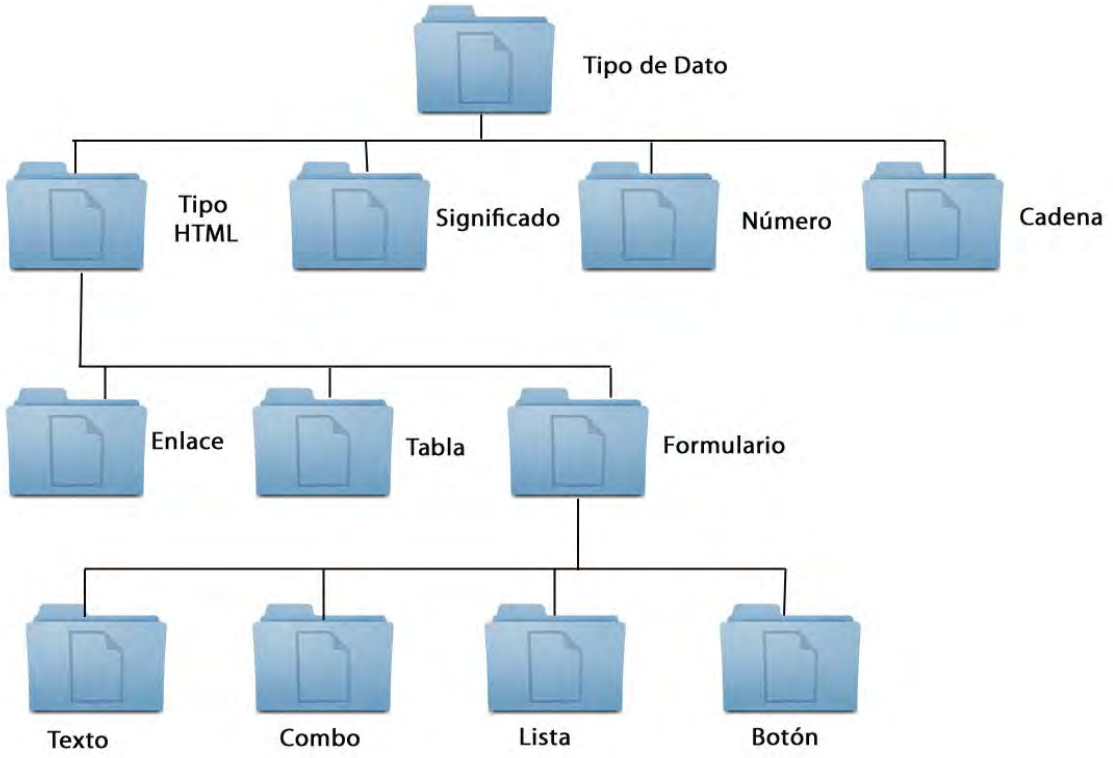
Relaciones entre documentos

Este tipo de relaciones definen la precedencia de documentos en una fuente. Esto es útil en algunas fuentes donde no se conoce la ubicación concreta de los documentos y es necesario realizar una secuencia de saltos previos para alcanzarlos. Solo tiene dos atributos:

- **documento origen:** Documento que precede.
- **documento destino:** Documento precedido.

3.3.1.4. Tipos de datos

La jerarquía de los tipos de datos abarca tipos básicos como cadenas, números o elementos que se puedan encontrar en documentos HTML.



[Figura 12] Jerarquía de los tipos de datos

Los valores de las piezas pueden ser de los siguientes tipos:

Cadena

Representa una sucesión de caracteres alfanuméricos. Tiene la siguiente lista de atributos:

- **nombre:** nombre del tipo de datos.
- **expresión regular:** Expresión regular que define el lenguaje de los valores.

Número

Representa un valor numérico. Tiene los siguientes atributos:

- **nombre:** nombre del tipo de datos.
- **expresión regular:** Expresión regular que define el lenguaje de los valores
- **operadores:** operadores de restricciones, positivo, entero, etc.)

Significado

Expresa en valor semántico de un dato. Dada una interpretación de lenguaje natural es posible determinar a nivel de sintagma o nivel de palabra qué significado tiene asociado.

De acuerdo diccionarios o léxicos usados es posible restringir su dominio en unos pocos significados. Por ejemplo si usamos WordNet [Miller 95] podemos determinar el *synset* que esperamos obtener, pudiendo además describir si admitimos los sinónimos del mismo. Los atributos que tienen son:

- **nombre:** nombre del tipo de datos.
- **expresión regular:** Expresión regular que define el lenguaje de los valores.
- **Significado:** significado semántico (*synset)*.
- **operadores:** operadores de restricciones.

3.3.1.5. Operadores

Las distintas estrategias propuestas para la implementación en el módulo de identificación de información modificaran el orden de ejecución de los operadores sobre los modelos de documentos. Los operadores son piezas de software que realizan las tareas de recuperación, navegación o comprobación sobre las interpretaciones de documentos disponibles y son llamadas por el módulo de identificación de información de acuerdo a la estrategia de información.

Los operadores se clasifican de la siguiente manera:

- **Operadores de alto nivel**: encargados de ejecutar una estrategia en particular (ordenando la ejecución de operadores de bajo nivel), provisión de nuevos documentos en modelos procesables o construcción de nuevas hipótesis.
- **Operadores de bajo nivel:** encargados de operaciones primitivas sobre las fuentes y la información extraída. Cada uno de estos operadores actúa sobre una determinada interpretación de documentos. De esta manera la provisión de un determinado modelo se convierte en precondición del operador (Ejemplo: no es posible ejecutar el operador de identificación de nombres propios si no se ha proveído el modelo de lenguaje sobre una fuente). Estos operadores a su vez se clasifican según su propósito:
    - **Operadores de recuperación de información** (*Retrieve):* Piezas de software capaces de localizar información dentro de una interpretación determinada.
    - **Operadores de validación** *(Check)*: Operadores capaces de verificar una propiedad impuesta (que la información sea del tipo número entero positivo, que tenga un significado X, etc.).

- **Operadores de ejecución y navegación:** *(Execute*). Operadores de manejo de fuentes, tales como la navegación por un documento hipertextual, localización de documentos por su URL, etc.

La ejecución de los operadores de recuperación tiene por objeto localizar una cierta información esperada en el documento fuente. Aunque esta pieza tiene impuestas condiciones de validez (por ejemplo: para posibles edades de personas: número entero positivo, comprendido en el rango de 0 a 120), son varias las posibilidades que se encuentran dentro de un mismo documento. Es por ello que el presente sistema que se pretende proponer construya una hipótesis que forman las distintas posibilidades de asignación de partes de la fuente a piezas de información esperadas.

Se clasifican según su efecto en las siguientes clases:

Operadores de Recuperación

Esta clase de operadores toma el modelo del documento fuente y obtiene un conjunto de piezas de información que se ajusta a la descripción dada por la ontología de adquisición.

Ejemplos:

- **Listar Nombres Propios:** Operador que trabaja sobre el modelo de lenguaje y recupera todas aquellas piezas de información que pudiesen constituir un nombre propio. Necesita del modelo de lenguaje para poder ejecutarse.
- **Lista Enlaces HTML:** Operador que devuelve una lista de enlaces que que pertenecen a un documento hipertexto.
- **Lista por Expresión Regular:** Operador que devuelve todas las piezas que encajan con una expresión regular dada.

Operadores de Ejecución

Los operadores de ejecución permiten realizar tareas de manipulación de documentos online tales como navegación entre documentos, interpretación de código embebido o carga de un documento para ser procesado por algunos de los modelos. La mayoría de ellos tienen como precondición la disponibilidad de la interpretación de la fuente DOM para tener formalizado los distintos objetos HTML.

Ejemplos:

- **Ejecutar Enlace HTML:** Encontrada una pieza del tipo: Enlace HTML (*HTML Link*) el operador navega hasta la página apuntada.
- **Proveer Página:** Descargar una página para su procesamiento por los distintos modelos.
- **Ejecutar Formulario:** Encontrado un formulario, se ejecuta.

Operadores de Comprobación

Operadores de comprobación toman piezas proporcionadas por los operadores de recuperación y comprueban si cumplen las restricciones impuestas. Estas restricciones pueden ser sobre su valor, sobre su tipo o sobre las relaciones que existen entre varias piezas.

Ejemplos:

- **Es un número entero:** Determina si un número hallado es entero.
- **Pertenece número rango:** Determina si el valor del número pertenece a un rango dado.
- **Existe Pieza en dominio:** Determina si una pieza está en la ontología de dominio.
- **Es Enlace HTML:** Devuelve *cierto* si la pieza es un enlace HTML.
- **Es Nombre Propio:** Devuelve *cierto* si es un nombre propio.
- **Están la Pieza A en Línea con B:** Comprueba las coordenadas de dos piezas para ver si están en la misma línea visual.
- **Está A a la derecha de B:** Comprueba las coordenadas de dos piezas para ver si una está a la derecha de la otra.

3.3.2. Estrategias

Como se ha comentado anteriormente son las estrategias que construyen dinámicamente las secuencias de los operadores con el objetivo de elaborar las hipótesis para el relleno. El algoritmo principal de ejecución de estrategias se realiza en un bucle de tres pasos:

Algoritmo principal del módulo de identificación:

```
MIENTRAS queden decisiones por tomar REPETIR:

   1. Tomar una decisión estratégica.
   2. Ejecutar la decisión tomada.
   3. Aplicar los resultados de la ejecución
```

```
FIN de Bucle
```

El objetivo de este algoritmo es construir un conjunto de hipótesis que relaciones piezas de información con partes del texto fuente para éstas pueden pasar a la fase final de relleno. Todo el espacio de búsqueda definido por la ontología de adquisición dará lugar a varios conjuntos de hipótesis. En cada conjunto las hipótesis compiten entre sí y forman alternativas para el relleno de piezas. Normalmente cada conjunto de hipótesis corresponde a un documento dentro de la red de documentos, aunque no es una restricción del sistema. Como se verá una hipótesis consiste en una asignación de candidatos formados por partes de fuentes extraídas de las fuentes a piezas de información definidas en la ontología de adquisición.

Dependiendo de la estrategia ejecutada el sistema generará una sola o varias hipótesis. Las estrategias actúan sobre el paso primero de toma de decisiones. Las decisiones que se pueden tomar son:

- **Recuperar una pieza:** Decisión de recuperar una determinada pieza mediante operadores de recuperación. Los operadores tomo como entrada la descripción de la pieza de la ontología de adquisición y recuperan todos aquellos fragmentos de la fuente que cumplan con las condiciones impuestas por la pieza, es decir: tipo de dato, restricción sobre valores, propiedades lingüísticas, etc. El resultado de esta operación es un conjunto de candidatos para ocupar la pieza descrita. Esta operación genera la parición de nuevas hipótesis para una parte del espacio de búsqueda, ya que muchas veces en número de candidatos para una pieza supera la cardinalidad máxima permitida de la misma.
- **Recuperar un documento:** Decisión de recuperar un documento completo de la fuente. El operador carga el documento en memoria, pero sin aplicarle ninguna interpretación al documento en concreto. Las interpretaciones de fuentes se irán generando según surjan las necesidades de los operadores de recuperación.
- **Recuperar una relación:** Decisión de comprobar una relación enunciada en la ontología de adquisición. Las relaciones, según se definen en la ontología de adquisición pueden ser de dos clases: de aspecto (posiciones espaciales relativas entre piezas) o de significado (relaciones semánticas identificadas como dependencias en el modelo del lenguaje).
- **Insertar hipótesis:** En cuanto se construye un conjunto de hipótesis para un subconjunto de piezas estas se pasan a la fase de relleno. Las distintas alternativas son evaluadas y solamente se selecciona la más plausible de todas ellas.

La manera de integrar una estrategia en módulo de identificación de información consiste en generar partes disjuntas de la ontología de adquisición y usar la estrategia para generar

un conjunto de hipótesis para cada una de estas partes. El conjunto de hipótesis correspondiente a cada parte se evalúa y ordena de acuerdo a su plausibilidad y aportación de información y se entrega al siguiente módulo que rellena la ontología de dominio.

3.3.2.1. Estrategia de Fuerza Bruta

La estrategia de fuerza bruta (*Greedy)* tiene por objetivo tomar decisiones encaminadas hacia una instanciación de datos lo más rápida posible. La restricción impuesta en esta estrategia en el mantenimiento de una única hipótesis para cada parte del espacio de búsqueda. La toma de decisiones sigue la siguiente secuencia:

- Recuperación de todos los documentos con URL conocida.
- Para cada documento se recuperan todas las piezas que le pertenecen y para cada pieza se recuperan todos los posibles candidatos.
- Para el documento se recuperan todas las posibles relaciones entre las piezas.
- Se comprueban las relaciones recuperadas y se eliminan los candidatos que lo incumplen.
- Si la hipótesis es válida (las cardinalidades se cumplen) se pasa a la inserción, en caso contrario se procede con el siguiente documento recuperado.

La disponibilidad de una única hipótesis impone la utilización de una heurística en la resolución de las ambigüedades causadas por excesivo número de candidatos para una pieza. Existen tres posibilidades en la toma de decisiones en la recuperación de una pieza P con una cardinalidad (N, M) para sus posibles valores:

- $(\mathrm{K} < \mathrm{N})$ El número de candidatos recuperados K es menor que la cardinalidad mímica (0)[4] de la pieza: Se descarta la hipótesis y el sistema termina anunciando la imposibilidad de cumplir con los requisitos de la ontología de adquisición.
- $(N \leq K \leq M)$ El número de candidatos recuperados K encaja en la cardinalidad de la pieza permitida: El sistema asigna los candidatos a la pieza.
- $(M < K)$ El número de candidatos recuperados K es mayor que la cardinalidad máxima de la pieza: se toman M piezas según una heurística que base la selección únicamente en la lista de candidatos.

[4] si cada instancia de la entidad no está obligada a participar en la relación

La estrategia de fuerza bruta es adecuada para sistemas con altos requisitos en tiempo de respuesta. Su complejidad es lineal dependiente del número de documentos y piezas descritas y halladas en las fuentes.

3.3.2.2. Estrategia de Búsqueda con retroceso (Backtraking)

En entornos donde el tiempo proceso de adquisición no es crítico o donde los requisitos sobre la calidad son elevados es conveniente usar otro tipo de estrategia que tenga en cuenta varias posibilidades y sea capaz de resolver ambigüedades comparando varias alternativas. Este es el caso de la estrategia de búsqueda con retroceso (*backtraking*). Esta estrategia despliega todo el posible árbol de búsqueda que consiste en la toma de decisión de las hipótesis, explotando todas las combinaciones. Cuando se encuentran más candidatos para una pieza de lo que su cardinalidad permite o si la cardinalidad es mayor de uno, es preciso bifurcar la construcción de las hipótesis ofreciendo todas las posibles combinaciones.

El proceso genérico de un algoritmo de búsqueda con retroceso sigue el siguiente seudo código:

```
Vector Backtraking (H)
{
  IF esFinal (H_i)   THEN
    {
      RETURN new Vector (H_i);
     }
    ELSE
    {
     Vector nuevas_hipotesis = ampliar (H_i);
     Vector  buenas hipotesis = filtrar_imposibles (nuevas_hipótesis);
     Vector resultado = new Vector ( );
     FOR (int = 0; i<buenas_hipotesis; i++)
     {
       Hipotesis h = (Hipotesis)  buenas_hipotesis.elementAt (i);
       Vector resultado_parcial = Backtraking (H);
       resultado = union (resultado, resultado_parcial);
     } // END FOR
     } // END IF
   } // END BCKTR
```

Donde las funciones que dirigen el proceso se describen así:

- *es final*: Determina si una hipótesis está finalizada. Esto ocurre si todos los documentos, piezas y relaciones han pasado por el proceso de asignación de candidatos a partir de los distintos modelos fuentes.
- *ampliar:* función que dada una hipótesis parcialmente construida genera un conjunto nuevo de hipótesis avanzadas, tomando todas las posibles decisiones. La explosión de ramas en el árbol de búsqueda sucede en una cardinalidad múltiple de una pieza de información. Suponiendo que una pieza tiene cardinalidad(N, M)[5] , siendo $M > 1$ y se localizan K candidatos para ser asignados a esta pieza:
    - o $(K < N)$: se descarta la hipótesis en curso y no se generan más hipótesis. La rama muere.
    - o $(N \leq K \leq M)$: se generan todas las posibles combinaciones de asignación de los K candidatos a la pieza. Finalmente habrá $K + 1$ nuevas hipótesis, una sin tomar ningún candidato en cuenta, una con uno solo, una con dos candidatos, así sucesivamente hasta generar la hipótesis que toma los K candidatos.
    - o $(M < K)$: El número de candidatos recuperados K es mayor que la cardinalidad máxima de la pieza. Para mantener exhaustivo el algoritmo se deben generar todas las posibles combinaciones de asignación de parte de los K candidatos en los $(M - N)$ puestos.
- *filtrar imposibles*: función que elimina las hipótesis parciales que incumplen alguna de las restricciones criticas impuestas por la ontología de adquisición.

El resultado de la función de búsqueda con retroceso es un conjunto de hipótesis que cubre parte de la ontología de adquisición. Es responsabilidad del proceso principal del modulo de identificación generar las distintas partes del espacio total de búsqueda, delimitado por la ontología de adquisición, para luego servir los conjuntos de hipótesis de las distintas partes a módulo de relleno de la ontología de dominio.

La complejidad de esta estrategia es exponencial en función del número de piezas descritas y encontradas. Para dominios con elevado grado de ambigüedad tienen unos altos requisitos en recursos.

[5] **"N", "M", ó "*"** si cada instancia de la entidad no está obligada a participar en la relación y puede hacerlo cualquier número de veces.

3.3.2.3. Estrategia de Búsqueda con retroceso Optimizada

La exhaustividad de un proceso como el descrito con la estrategia de búsqueda con retroceso garantiza soluciones de calidad pero lleva asociado un alto costo computacional. Existen muchas variedades del original algoritmo para paliar este consumo computacional. En el sistema propuesto se ha optado por combinar dos de ellas:

- ***Poda:*** Poda del árbol de búsqueda de acuerdo a una función de costo. Las hipótesis, aún en un estado de construcción parcial, deben contar con una función de evaluación de su costo, tanto real como estimado. Gracias a esta función, la estrategia puede priorizar la consecución de las soluciones más prometedoras en cuanto a un costo mínimo. Este costo se usa como umbral en la construcción de las ramas de búsqueda que se podan si superan el alcanzado hasta ese momento.
- ***Heurística:*** Introducción de heurísticas que no reducen el número de ramas en el árbol de búsqueda pero que persiguen retrasar las ampliaciones de las hipótesis que mayor ambigüedad generan lo máximo posible.
    - Priorizar las decisiones de recuperación de candidatos para piezas que participan en relaciones críticas.
    - Priorizar las decisiones de recuperación de candidatos que no introducen mucha ambigüedad, es decir, cuyo número total no excede la cardinalidad máxima de la pieza.

El proceso genérico de un algoritmo de búsqueda con retroceso optimizado sigue el siguiente seudo código, como se muestra en la figura 12.

Donde las funciones que dirigen el proceso son una variante de las funciones de la estrategia genérica de búsqueda:

- ***ampliar***: función similar al proceso de *backtraking* pero con heurística programada para priorizar las decisiones que o bien pueden descartar hipótesis o bien no introducen mucha ambigüedad.
- ***podar:*** la existencia de una variable que alberga el costo mínimo alcanzado hasta ese momento, permite detener el algoritmo en aquellos nodos cuyo costo, aún con hipótesis parcialmente construidas, supera el almacenado. Estas hipótesis borran el conjunto de trabajo.
- ***ordenar:*** permite ordenar las hipótesis para situar a las más prometedoras (menor costo alcanzado y estimado) como las primeras para ser procesadas. Esto permite alcanzar costos mínimos con más rapidez y así aprovechar más la poda de los nodos.

```
Vector Backtraking (H_i, funcion_costo)
{
  IF  esFinal (H_i)  THEN
    {
   costo_minimo = MIN (costo_minimo, funcion_costo(H_i));
   RETURN new Vector(H_i);
    }
   ELSE
    {
    Vector nuevas_hipótesis = ampliar (H);
    Vector buenas_hipotesis = filtrar_imposibles (nuevas_hipótesis);
    Vector hipotesis_podadas = podar (buenas_hipótesis, funcion_costo);
    Vector resultado = new Vector ( );
    FOR  (int = 0; i<hipótesis_ordenadas; i++)
   {
       Hipotesis h = (Hipotesis) hipotesis_ordenadas.elementAt(i);
       Vector resultado_parcial = Backtraking (h, funcion_costo);
        resultado = union(resultado, resultado_parcial);
    } // END FOR
}//END IF
}// END BCKTR_Opt
```

[Figura 13] Pseudocódigo de un algoritmo de búsqueda con retroceso optimizado

El orden computacional de este algoritmo no es distinto, que el orden del algoritmo genérico de búsqueda con retroceso pero el uso de podas y heurísticas permite disminuir ambigüedades tratadas y de esta manera reducir el tiempo y el consumo de memoria mismo.

3.3.3. Hipótesis

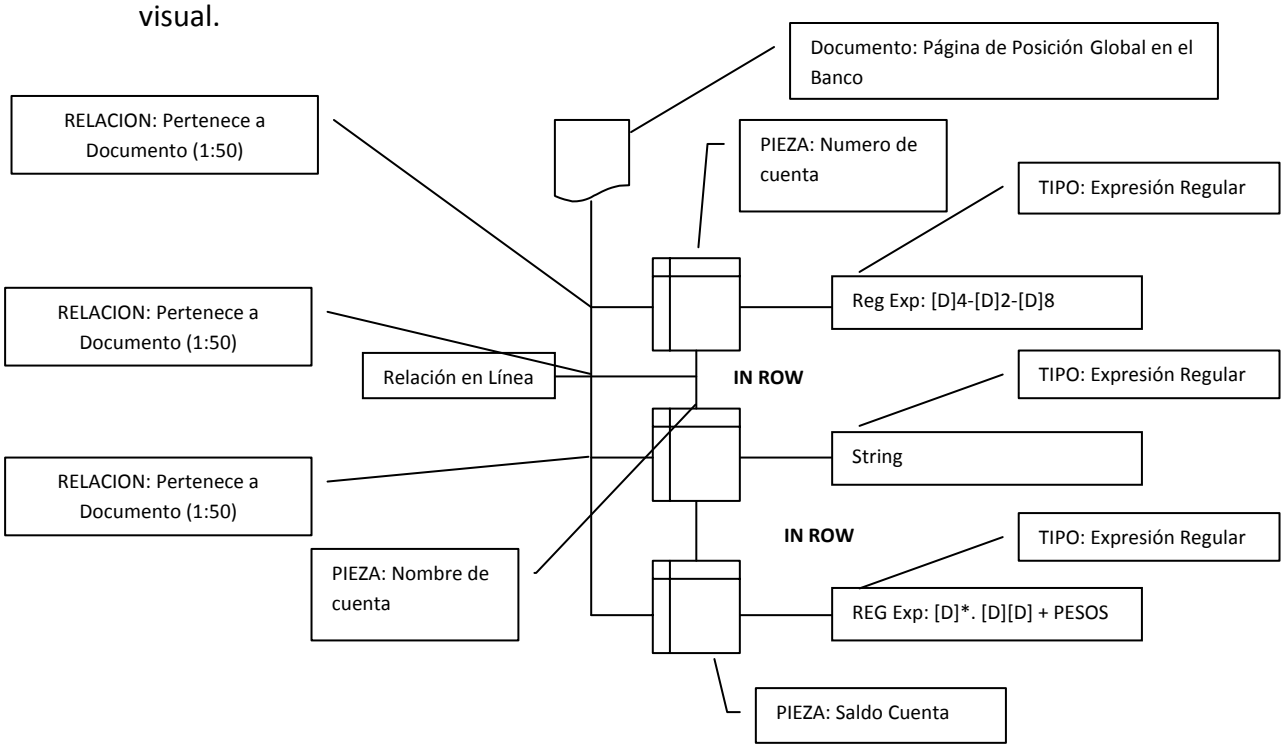
Las hipótesis son estructuras de datos resultantes de la fase de identificación que albergan la información sobre la asignación de distintas partes de los documentos fuentes con piezas de información esperadas y descritas en la ontología de adquisición. Como efecto de la existencia de restricciones sobre las piezas de información esperadas (su tipo de

datos, relaciones entre ellas, cardinalidad, etc.) se puedan considerar varias configuraciones de asignación de partes de las fuentes a piezas de información. Estas configuraciones se reflejan en las hipótesis resultantes del modulo de identificación.

Ejemplo:

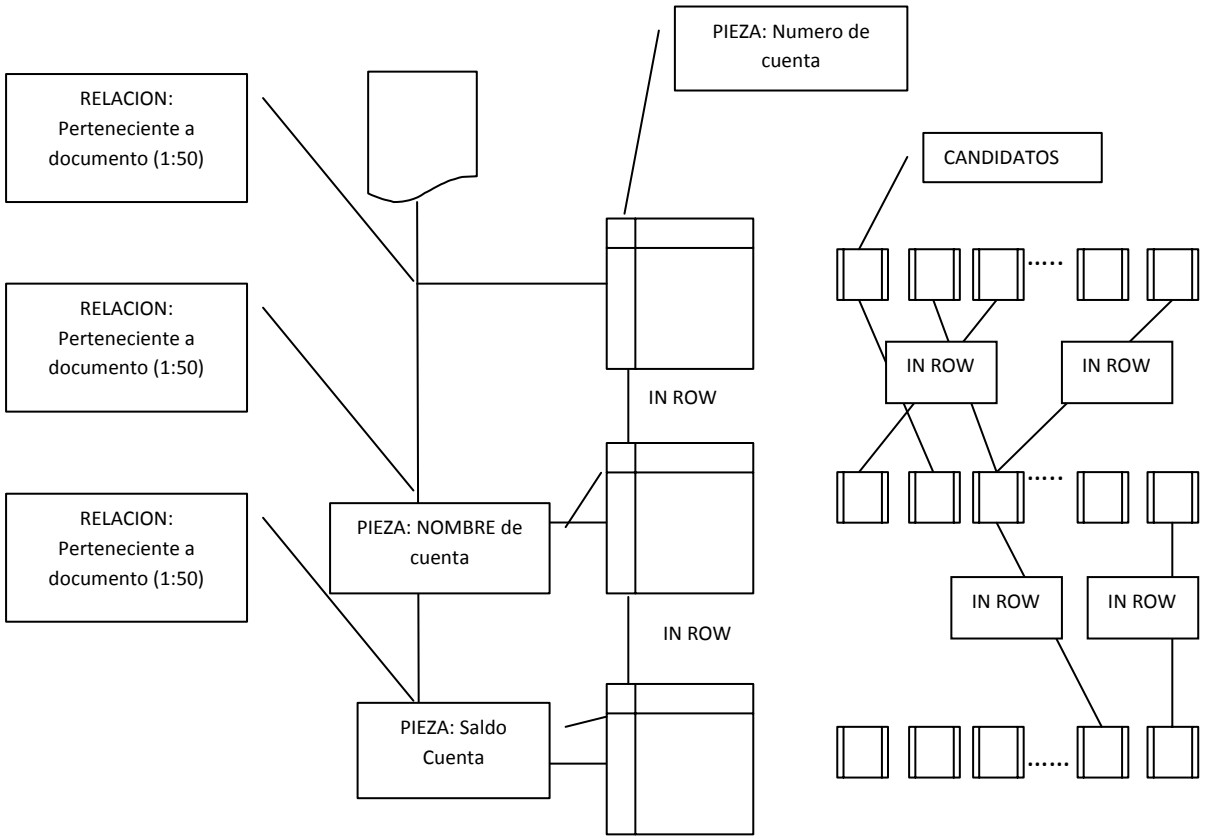
La ontología de adquisición describe un documento típico de los Portales de los bancos. La página de posición global agrega las cuentas de usuario, mostrando el balance de cada una de ellas. En un modo simplificado y dado que se trata de un documento altamente estructurado, la posible formalización de este documento constaría de los siguientes elementos:

- Documento que alberga las distintas piezas.
- Número de cuenta con una estructura muy bien definida en una expresión regular.
- Descripción de la cuenta considerada una cadena alfanumérica.
- Saldo de la cuenta en formato de cantidad monetaria.
- Dos relaciones de aspecto (IN ROW) que sitúan las tres piezas en una misma línea visual.



[Figura 14] Ejemplo de uso de la ontología de adquisición para un documento de posición global en la Web de un banco (simplificado).

Las distintas estrategias invocarán operadores para asignar candidatos a las tres piezas presentes en el modelo. Según qué estrategia, primero se invocaran operadores de recuperación de información ó operadores de comprobación de las relaciones. Las estrategias generan más de una hipótesis si la cardinalidad de una pieza es mayor que uno. Si se recuperan N candidatos para una pieza con cardinalidad M hay que poder generar hipótesis sobre que candidatos tomar en cuenta. Ya se ha visto anteriormente que alguna hipótesis se puede desechar si alguna condición crítica no se cumple (por ejemplo: la relación EN LINEA es obligatoria).



[Figura 15] Una hipótesis como propuesta de asignación de candidatos a piezas de información

El número de hipótesis depende de la estrategia ejecutada. De esta manera, la estrategia fuerza bruta aplica heurísticas para generar a lo suma una única hipótesis, mientras una estrategia de búsqueda con retroceso pura (sin ninguna heurística) puede generar millones de hipótesis.

3.3.3.1. Evaluación de las Hipótesis

Con el fin de obtener un relleno correcto de los datos en la ontología de dominio es necesario poder seleccionar una única hipótesis. Se requiere el establecimiento de un mecanismo de evaluación con el fin de poder establecer un orden entre ellas. Algunas

estrategias como la búsqueda con retroceso óptima, hacen uso de las evaluaciones para realizar podas en el árbol de búsqueda de la solución. Esto es posible solamente si el mecanismo de evaluación incluye una función de estimación que permita calcular el posible futuro valor de la hipótesis aunque ésta no esté completamente construida. La función de evaluación está concebida conceptualmente como una función de costo, cuanto más alto valor devuelve, menos plausible es la hipótesis que evalúa.

$$Evaluar(H) = Evaluar(H_1, \psi, \phi) + Estimar(H_2, \psi, \phi)$$

La formula expresa de manera conceptual como se calcula el valor de una hipótesis H, construida parcialmente y que consta de dos partes:

- H_1: Parte construida de la hipótesis:
  - Piezas de información con candidatos
  - Relaciones comprobadas
  - Documentos recuperados
- H_2: Parte que queda por construir de la hipótesis
  - Piezas sin candidatos
  - Relaciones sin comprobar
  - Documentos que puedan surgir

Las funciones ψ y ϕ se definen como:

ψ: Función que expresa en grado de incumplimiento de la hipótesis con las restricciones opcionales en la ontología de adquisición. La ontología de adquisición contiene relaciones marcadas como opcionales (documento puede contener una pieza, dos piezas pueden estar en línea, etc.) y su incumplimiento aumenta el valor devuelto por esta función.

ϕ: Función que expresa el beneficio de la hipótesis. Cuantas más piezas, relaciones o documentos aporta la hipótesis mayor es su beneficio y disminuye su costo total. Esta función se usa para evitar que las hipótesis vacías o con poca información se sitúen como las más plausibles por no incumplir ninguna directiva de la ontología de adquisición.

Existen varias maneras de combinar los valores de las dos funciones de evaluación, aunque las más inmediatas y fáciles de implementar son:

- $\psi - \phi$: Suma de la función de incumplimiento con el valor negativo de la función de beneficio.
- ψ / ϕ: Expresa en grado de incumplimiento en proporción a la información aportada.

La función de evaluación tiene un doble objetivo: permitir a las estrategias implementar heurísticas basadas en la función de costo y estimación (podas, priorización, etc.) y por otra parte permite ordenar las hipótesis para ser rellenadas en la ontología de dominio. La mejor hipótesis será aquella que mejor evaluada salga del módulo de identificación y menos inconsistencias genere en el modelo. La evolución de las posibles inconsistencias introducidas se explica en la sección siguiente.

3.4. Relleno de Ontologías

El relleno de ontologías es la última fase de ejecución del sistema propuesto. Comprende la inserción de valores recuperados en lugares de un modelo de dominio lo puede parecer un problema a primera vista trivial. En el estado de arte de sistemas de adquisición de información y su posterior inserción en una ontología, no existe un estudio amplio de los problemas que la inserción pueda presentar. La aproximación más frecuente es dotar a un sistema tradicional de extracción de información con un módulo de inserción de los datos en lugares preestablecidos en la ontología de dominio. Cada pieza identificada en las fuentes tiene asociada una información sobre su lugar en el modelo de dominio: a que concepto corresponde y que atributo se ha de rellenar. La mayoría de las veces estos sistemas únicamente identifican nuevas instancias de conceptos de dominio. Cada nueva pieza encontrada en las fuentes da lugar a nuevas instancias de un concepto predeterminado, lo que implica una creación de una instancia y el relleno de su atributo nombre o etiqueta, dependiendo del formalismo del modelo de dominio.

3.4.1. Operaciones en la instanciación de Ontología de Dominio

El relleno de una ontología con datos extraídos de fuentes no estructuradas puede implicar alguna de las siguientes operaciones:

- **Creación de nuevas instancias para el relleno de un atributo:** Es la operación más básica en el relleno de las ontologías y es la operación más frecuentemente cubierta por los sistemas existentes. Implica la creación de una nueva instancia en la ontología de dominio con el dato extraído figurando como atributo de un valor. Sucede frecuentemente con nombre propios hallados en las fuentes, ya que suelen dar lugar a instancias nuevas de personas, organizaciones, lugares, etc.
- **Relleno de atributos de instancias existentes con valores extraídos del texto:** En otras ocasiones el dato extraído de las fuentes puede completar alguna instancia ya existente sin tener la necesidad de crear una instancia nueva. Surge el problema de identificar si la instancia que se complementará existe y en caso afirmativo cual es.

- **Modificación de valores de instancias existentes:** En esta operación el dato recuperado de las fuentes, modifica un dato anteriormente insertado. El uso de esta operación no es frecuente y requiere una clara definición del cálculo de la plausibilidad y la reputación de los datos extraídos así como la posible definición de grados de consistencia de los datos en la ontología de dominio.
- **Relleno de atributos para relacionar instancias existentes o de nueva creación:** La siguiente operación es relativa al manejo de relaciones en la ontología de dominio. En lugar de insertar nuevos datos o crear nuevas instancias esta operación añade información semántica respecto a las relaciones de los datos existentes. En la ontología de dominio esto se traduce al relleno de atributos cuyo rango son otras instancias, en vez de ser datos extraídos de las fuentes. Este tipo de operaciones normalmente viene desencadenando por la localización de relaciones entre las piezas de información definidas en la ontología de adquisición. La relación visual que sitúa en una misma línea a un autor y a su obra nos aporta la información sobre la existencia de una relación en la ontología de dominio entre la instancia del autor y la de la obra, aunque ya existan ambas.
- **Modificación de relaciones entre instancias existentes:** Al igual que la modificación de valores de atributo existentes es una operación compleja, la operación capaz de borrar relaciones requiere una clara definición de las políticas de evaluación y actuación sobre los datos existentes.

3.4.2. Proceso de Instanciación

El problema que se presenta es debido a que aunque las operaciones de inserción y modificación de los datos extraídos en la ontología estén definidas, es muy complejo definir cuando aplicar cada una de ellas. Desde el punto de vista del usuario no es factible prever todas las aplicaciones combinaciones de valores ni tampoco es posible definir reglas sencillas que decidan si la aplicación o no de las distintas operaciones. En el sistema propuesto se solicita al usuario información sobre donde insertar el valor obtenido (concepto y atributo).

3.4.2.1. Información para la Inserción

En la ontología de adquisición para la definición de una pieza de información se ha previsto un atributo que indica el lugar donde debe de insertarse el valor recuperado. Cada candidato de una pieza de información almacenado en la hipótesis tratada puede tener relleno este valor indicando el concepto y el atributo que rellenan.

Ejemplo:

Pieza: Nombre de la Persona

- Tipo: Nombre Propio; Enlace HTML
- Inserción:
  - Ontología del Dominio: People
  - Concepto (URI): http://owl.man.ac.uk/2006/07/sssw/people.owl
  - Instancia (URI):
  - Atributo (Etiqueta): Nombre
  - Clave (Booleano):

La información de relleno contiene información sobre a qué ontología pertenece el concepto que se rellenará, cual es el atributo afectado y también es posible indicar la instancia concreta que se rellena. El valor de 'Clave' indica si el atributo rellenado identifica a la instancia. Esto se usa para decidir si se crea una instancia nueva o se rellenan instancias existentes.

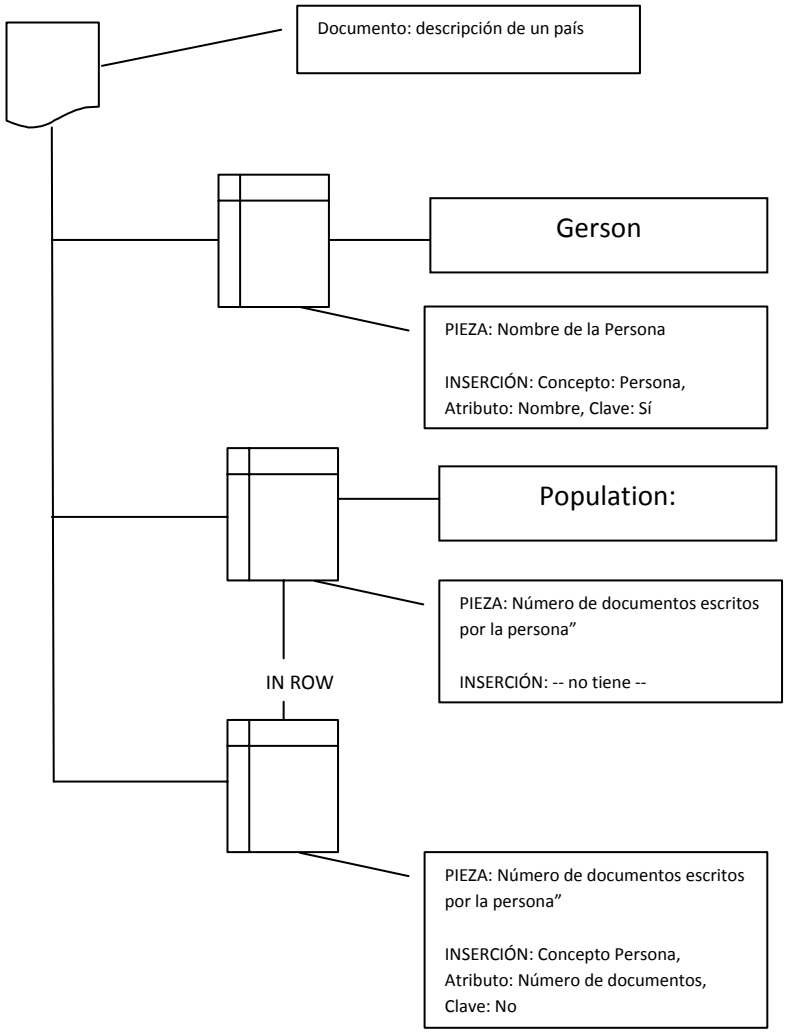
3.4.2.2. Memoria Temporal de Contexto para la Resolución de Ambigüedades

La entrada al propio proceso de inserción es una hipótesis resultante del proceso de identificación de información. Esta hipótesis está formada por documentos, relaciones y piezas asociadas a datos encontrados en las fuentes. La tarea del módulo de relleno es tomar todos los candidatos de la hipótesis y rellenarlos según la información contenida en los atributos de 'Inserción'. La cardinalidad de este atributo en la ontología de adquisiciones es múltiple, con lo cual cada pieza puede tener más de una acción de inserción asociada. La mayoría de los casos el atributo 'Instancia' no contiene datos sobre la instancia que debe rellenarse y es el propio proceso de relleno que debe identificar de que instancia existente se trata o tomar la decisión de crear una instancia nueva.

Para la resolución de ambigüedades sobre que instancia debe rellenar el sistema contara con una memoria cache, que realiza las funciones de contexto en el proceso de relleno. Cuando una instancia se ve afectada por una acción de relleno (creación o modificación), se almacenará una marca temporal que indica cuanto de reciente fue esta modificación. Las ambigüedades que pueden surgir a decidir que instancia debe rellenarse se resuelven tomando la instancia más reciente de la memoria caché que no implique una modificación de algún valor existente.

Ejemplo:

Una hipótesis simplificada que contiene datos sobre el nombre de una persona y un valor del número de documentos que ha escrito.



[Figura 16] Hipótesis para el relleno de datos de population de documentos de Gerson

El proceso de relleno toma los atributos mercados como claves y realiza las operaciones de inserción, que en este caso comprueba si existe alguna instancia del concepto "Persona" con nombre igual a "Gerson". Si no existe se crea una nueva instancia y se marca en la memoria cache como recientemente modificada.

Seguidamente el proceso de relleno toma aquellos candidatos que tiene información de inserción rellena y marcada como clave para realizar el relleno complementario. En este paso se procede a rellenar el atributo de "Número de documentos" del concepto "Persona". Para ellos se buscan las instancias de concepto "Persona" que no tenga el atributo de Número de documentos relleno a un valor distinto y se ordena según la memoria caché, recientemente el más antiguo. Se rellena el atributo del primero de ellos.

Si no hubiese ningún candidato, se ejecuta la operación de creación de una nueva instancia.

3.4.3. Simulación

El módulo de identificación de información proporciona una serie de conjuntos de hipótesis. De esta manera si una estrategia construye un conjunto de hipótesis por cada documento identificado, el módulo de relleno de la ontología recibe más de una hipótesis para el relleno de los datos de ese documento. El módulo de identificación evalúa las hipótesis de acuerdo a su plausibilidad e información que aportan, con lo cual el módulo de relleno puede tomar la mejor de ellas. La evaluación del modulo de identificación se realiza de acuerdo a las restricciones sobre los datos y a la cantidad de información disponible, sin tener en cuenta el impacto que un inserción podría tener en la ontología de dominio. Podría darse el caso que la hipótesis mejor valorada en cuanto a restricciones impuestas introduzca una gran cantidad de inconsistencia en la ontología obligando a realizar operaciones de borrado o modificación de valores existentes.

El módulo de relleno realizara simulaciones de las inserciones de las hipótesis para completar la evaluación realizada por el módulo anterior. En esta simulación se mide:

- Número de modificaciones realizadas en la ontología del dominio.
- Número de creaciones de instancias
- Franja temporal máxima usada en la memoria caché
- Tiempo global de cada simulación

El objetivo de las simulaciones es calcular lo costoso que resulta insertar instancias en la ontología de dominio en términos de creación de nuevas instancias o modificación de las existentes. Partimos del supuesto que la información encontrada no tiende a contradecir la información ya almacenada y se premia a aquellas hipótesis que no introducen inconsistencias. Esta medida permite enriquecer la información de evaluación proporcionada por el módulo anterior y reordenar las hipótesis.

4. Propuesta de Implementación de Arquitectura

En el paradigma de la Web Semántica el contenido se anota de acuerdo a ontologías, obteniendo o bien documentos extendidos con marcas semánticas o bien contenido semántico, ambos ligados a instancias de ontologías. Estas instancias de las ontologías, ya están almacenadas en bases de datos, ficheros o estén distribuidos junto con el contenido existente, aportan poco valor añadido si no se explota su aspecto formal y consensuado en beneficio de alguna aplicación concreta. De acuerdo a la evolución de los modelos de uso apuntada por la comunidad de la Web Semántica y comentada en este trabajo, los sistemas de gestión documental constituyen uno de los primeros hitos[1] a corto plazo.

En este capítulo se presenta un posible uso del contenido semántico y se divide en dos partes, la primera titulada Gestión y Búsqueda de Información Semántica que describirá un sistema genérico de publicación de datos semánticos a través de un portal Web que permite visualizar los datos adquiridos por el sistema propuesto. En la segunda parte titulada Portal Semántico se aplica el sistema de publicación a cualquier dominio. Se mostraran algunos ejemplos de uso de ambos sistemas, el de adquisición y el de publicación, completando así el posible ciclo de vida de los datos semánticos, desde su creación hasta su uso.

4.1. Gestión y Búsqueda de Información Semántica

La gestión y recuperación de información es el hito más cercano de la tecnología de la Web Semántica. La información estructurada ofrece grandes posibilidades respecto a las búsquedas tradicionales basadas en palabras clave.

4.1.1. Contexto de Sistemas de Gestión Actuales

Cuando se busca información en sistemas de ayuda existentes, sistemas de gestión documental o en Internet, no se obtienen respuestas como esperaríamos, sino documentos. Los sistemas actuales recuperan los documentos que pueden o no contener las respuestas a las preguntas formuladas, basándose en apariciones de palabras clave, sus frecuencias u otras heurísticas propias del área de recuperación de información. Actualmente, en una sociedad cada vez más basada en la información y en el conocimiento, estos sistemas tienden a cambiar hacia soluciones que encuentren y formulen respuestas concretas, que aprendan de los usuarios u ofrezcan la información solicitada de manera visual y fácil de entender. Existe una gran demanda, sobre todo por pate de profesionales no informáticos, de sistemas que les permitan localizar y manipular

[1] Punto de control de objetivo intermedio antes de que el proyecto finalice.

información concisa, sin tener que expresar sus preguntas y criterios de búsquedas en lenguajes crípticos.

Esta revolución en el campo de la recuperación o extracción de la información viene alimentada por las mejoras en el hardware informático, capaz de manejar grandes cantidades de datos de manera efectiva, por la aparición de tecnologías de gestión de conocimiento que nos permiten comprender la manera en la que los humanos manejamos y buscamos la información así como por tecnologías que nos permitan que los ordenadores interpreten nuestro lenguaje y a su vez aprendan a adaptarse a nuestras necesidades de manera dinámica.

La tecnología actual empleada en sistemas de gestión documental permite almacenar grandes cantidades de documentos que se clasifican por unas pocas características predefinidas. Estos sistemas pueden incluir un servicio de publicación del contenido en formato de la WWW. La funcionalidad principal ofrecida a los usuarios finales es el listado de los documentos de acuerdo a los criterios de clasificación y su recuperación por palabras clave. La calidad de este servicio depende en gran medida en el esfuerzo invertido en la codificación o clasificación previa y manual del documento. Queda como responsabilidad del usuario codificar el contenido del documento de acuerdo a las meta-datos y palabras clave disponibles.

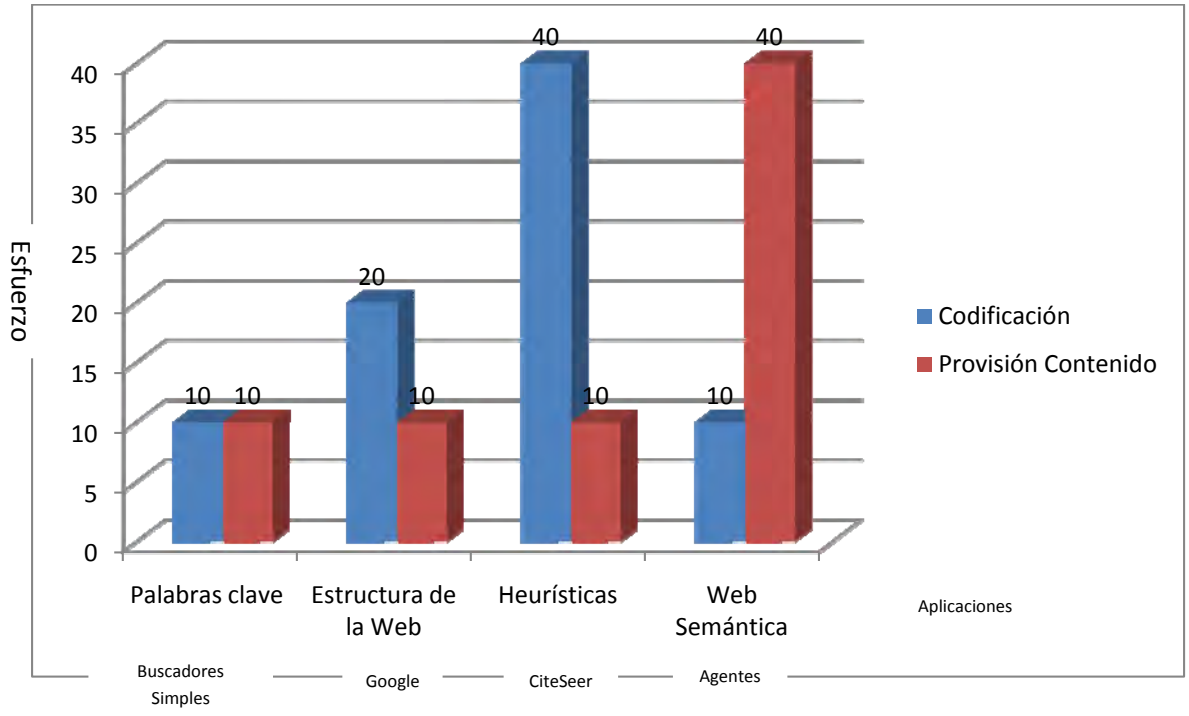


Figura 1 Relación ilustrativa en el esfuerzo de codificación y complejidad de las aplicaciones

Los buscadores simples son ya un estándar y no requieren grandes esfuerzos ni en codificación ni tampoco en adecuación y provisión del contenido. Algunos buscadores avanzados, como Google (Google), tienen también en cuenta la estructura hipertextual de la red y la usan para ordenar los resultados de acuerdo a una reputación calculada para cada documento. En la figura 1

Se pueden observar cual es la relación entre el esfuerzo invertido en la preparación de los contenidos (su codificación) y el esfuerzo en la codificación de las propias aplicaciones. Las aplicaciones a su vez ordenadas por el grado de sofisticación en sus funcionalidades. Se pueden observar por ejemplo, los buscadores de palabras clave, apenas necesitan gastar esfuerzo en la preparación del contenido ofreciendo un nivel bajo en funcionalidades. De la misma manera Google que ofrece mejores resultados de búsqueda tiene aumentada la necesidad de esfuerzo en codificación de la aplicación ya que el contenido se tiene que procesar (algoritmo de reputación de páginas: *PageRank*^{TM} [2]). El caso de la aplicación bibliográfica en el ámbito científico CiteSeer (CiteSeer) es más pronunciado. Procesa publicaciones científicas en diversos formatos y extrae información sobre los autores, bibliografía, palabras clave. Etc. El esfuerzo en implementación de una heurística de análisis de documentos es grande.

Sin embargo en el marco de la Web Semántica, con todo el contenido generado de acuerdo a modelos semánticos consensuados, el esfuerzo requerido se centrará en la lógica de negocio de la aplicación en vez de su provisión.

Los nuevos sistemas de gestores documentales semánticos que están emergiendo, constituyen soluciones construidas sobre sistemas de análisis y comprensión de lenguaje humano muy cercano al humano, y que permiten extraer el significado de las preguntas formuladas, contrastándolas contra un modelo lógico que delimita el dominio con información semántica lingüística.

A continuación se presenta la propuesta de implementación de una plataforma de gestión de información que se basa en el uso de ontologías para el modelado de conocimiento y ofrece funcionalidades mejoradas de publicación de contenido en un portal Web así como un buscador semántico. Su objetivo es permitir la publicación de documentos online accesible y recuperable usando una ontología construida en colaboración con los expertos del dominio.

[2] es una marca registrada y patentada por Google el 9 de enero de 1999 que ampara una familia de algoritmos utilizados para asignar de forma numérica la relevancia de los documentos (o páginas web) indexados por un motor de búsqueda. Sus propiedades son muy discutidas por los expertos en optimización de motores de búsqueda. El sistema PageRank es utilizado por el popular motor de búsqueda Google para ayudarle a determinar la importancia o relevancia de una página. Fue desarrollado por los fundadores de Google, Larry Page y Sergey Brin, en la Universidad de Stanford.

4.1.2. Portal Semántico

A pesar de las ventajas que los modelos semánticos presentan a la hora de modelar información, uno de los problemas de éstos es su presentación legible y comprensible a un usuario humano. A pesar de que la intención original de los modelos semánticos en el paradigma de la Web Semántica fue servir de base de conocimiento para software o agentes inteligentes, nos encontramos con el problema que los usuarios humanos dejamos de entenderlos. Sobre todo en las primeras fases de evolución de la Web Semántica, donde no existe una población amplia de software capaz de consumir este contenido, necesitamos mecanismos que hagan visible el contenido a los humanos.

Así mismo la disponibilidad de un modelado de información semántico, formal y consensuado, ofrece un gran abanico de posibilidades en la construcción de sistemas de gestión de información. Aquí incluiremos gestores documentales, portales Web y buscadores online. Con la constante evolución de este tipo de aplicaciones se puede observar que sus fabricantes tienden a modelar la información con el objetivo de mejorar las funcionalidades de gestión y abaratar los costos de producción. La aproximación aquí presentada propone el uso de formalismos de ontologías para el modelado de la información, que son formalismos igual de formales que los modelos propietarios de los fabricantes.

4.1.2.1. Arquitectura Lógica de un Portal Semántico

Al auge de los portales está fomentando la aparición de plataformas de desarrollo, las cuales se puede construir un portal con muy poco esfuerzo y con unos resultados aceptables. Esto se debe a que las distintas partes funcionales y de contenido de los portales, ya constituyen un estándar ampliamente aceptado por usuarios de la WWW. Un portal semántico, como se propone en este trabajo, permite usar ontologías para modelar la información, y permite navegar por las instancias de los conceptos. Estas instancias constituyen la pieza básica del portal, siendo la información publicada ligada a los documentos relacionados, y siendo también la información encontrada en los buscadores. Los buscadores evolucionan de esta manera y en vez de devolver documentos relevantes, como se hace en la mayoría de los portales actualmente, los buscadores de los portales semánticos devuelven instancias de conceptos como respuestas a las preguntas.

El portal semántico consta de tres módulos diferenciados como se muestra en la figura 2:

- **Módulo de interpretación de búsquedas:** encargado de interpretar las búsquedas hechas por el usuario dentro del modelo semántico definido. El interfaz sencillo, consiste en una serie de formularios que se corresponden con conceptos definidos

en la ontología, donde el navegante puede definir criterios de búsqueda rellenando parcialmente los valores de los atributos.

- **Módulo de adquisición:** recupera información de fuentes online y los inserta en la ontología de dominio para que forme parte del portal.
- **Modulo de presentación:** encargado de presentar las instancias de la ontología de dominio y permitir una navegación entre ellos.

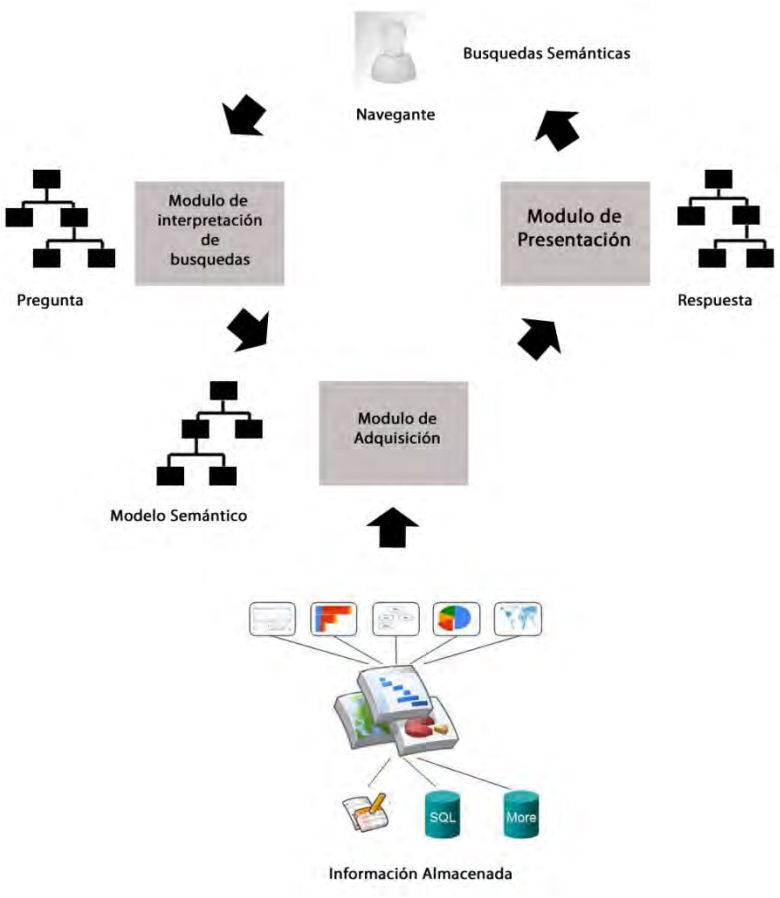


Figura 2 Esquema de un Portal Semántico

4.1.2.2. Modelo de Conocimiento Publicable

En la fase de diseño y construcción de una ontología se valora su poder expresivo y no se toman en cuentan los aspectos visuales o estéticos del mismo. La modelación de los conceptos y relaciones de las ontologías de dominio no se debe ver restringida por criterios de publicación de su contenido como podrían ser el número de atributos de un concepto, numero de instancias o existencia de conceptos auxiliares para la representación de relaciones. Si se publica directamente una ontología construida por

expertos con el objetivo de modelar un dominio particular, es probable que nos encontremos conceptos que tienen demasiados atributos, relaciones modeladas por conceptos o conceptos abstractos con poca información útil. Todos estos fenómenos son irrelevantes en el modelo semántico desde el punto de vista de la expresividad de una ontología pero pueden ser perjudiciales desde el punto estético cuando la ontología es usada por usuarios humanos.

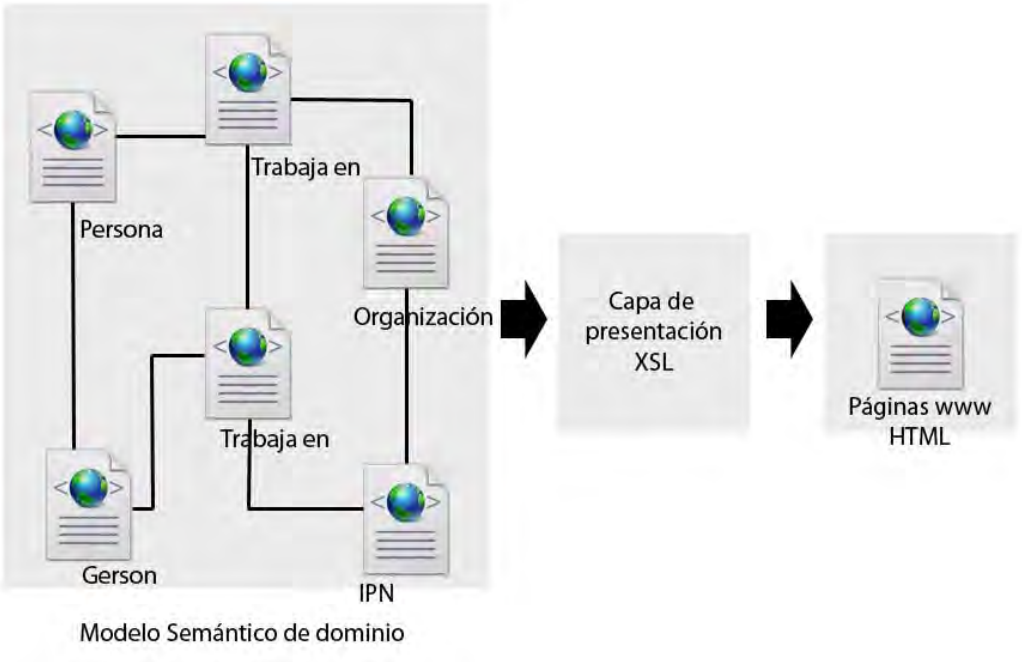


Figura 3 Método usual de publicación de ontologías

Los lenguajes de la Web Semántica están basados en el estándar sintáctico XML y son fácilmente publicables con transformaciones XSL (XSL), un formalismo de reglas de transformación, que generan páginas HTML. Estas transformaciones hacen muy costoso los cambios profundos sobre la estructura de la información y solamente se suelen usar para definir aspectos estéticos del contenido HTML.

El proceso de publicación propuesto en este trabajo se basa en la existencia de una ontología auxiliar, llamada ontología de visualización o modelo de publicación, que permite definir vistas sobre la ontología de dominio. Estas vistas se definen de acuerdo con criterios de usabilidad y estéticos con el fin de presentar una presentación legible del modelo semántico.

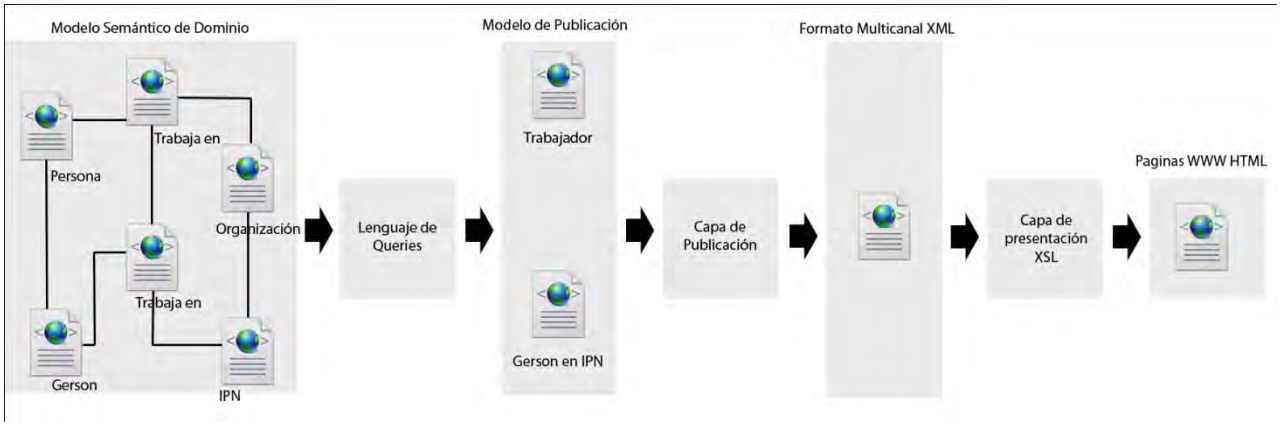


Figura 4 Fases de publicación con un modelo de publicación

El modelo de publicación sigue el formalismo de ontología y actúa como un contenedor de entidades y atributos publicables que extraen los valores mediante lenguajes de consultas sobre la ontología de dominio. Estas entidades publicables, contienen aquellos atributos de la ontología de dominio que se van a presentar al usuario final. El contenido de una instancia publicable puede agrupar varios conceptos del modelo semántico original, o al contrario, pueden dividir un concepto de dominio complejo en varias instancias publicables.

Más concretamente, la ontología de visualización incluye dos conceptos predefinidos, que realizan la función de meta-modelo:

- **Entidad de publicación:** concepto que encapsula objetos tal, como se verán publicados. Todo concepto definido en la ontología de publicación heredará de él y, deberá definir los siguientes atributos:
  - **Hoja de estilo** asociada al concepto que traduce sus instancias publicables.
  - **Consulta** que obtiene todos los valores de los atributos de la instancia correspondiente en la ontología de dominio.
- **Atributo de publicación:** todos los atributos que se muestren en la aplicación final deben heredar de este concepto. La forma que en el atributo se muestre en la página Web se define mediante las siguientes propiedades:
  - **Etiqueta:** La posible etiqueta que aparecerá con el valor del atributo.
  - **Consulta:** se ejecuta para obtener el valor del atributo.
  - **Enlace:** si el valor publicado debe realizar algún acción al pulsar sobre él (enlace Web, mail, botón, etc.), la acción se describe aquí:

Las componentes de una página del portal para visualizar instancias de la ontología se describen como subclases de Entidad publicación, y sus instancias se definen de acuerdo con el canal de publicación (HTML, WAP, Voice-VML, XML) a generar, a través de

transformaciones XSL. La administración de portales donde se ha empleado el uso de esta tecnología se divide así de manera natural en dos partes:

- Administración del contenido: gestiones sobre la ontología de dominio.
- Administración de la presentación: gestiones sobre la ontología de visualización que permiten modificar la agrupación de la información presentada así como su estética.

La separación entre la ontología para la representación del conocimiento del dominio y la ontología de visualización facilita la independencia de estas dos labores. Otra ventaja de la aproximación seguida es que para ambas se pueden utilizar las mismas herramientas de manejo de lenguajes de ontologías, como Protégé (Protege).

4.1.2.3. Búsquedas en un Portal Semántico

Las búsquedas en un portal semántico se basan en el uso de la ontología de dominio. A diferencia de los buscadores tradicionales basados en palabras clave, donde la respuesta es una lista de documentos que contienen la cadena buscada, los buscadores semánticos tienen en cuenta el significado de la palabra buscada y devuelven instancias de conceptos en vez de documentos completos. Este tipo de búsquedas permite recuperar instancias de conceptos, acotando los valores de los atributos que los caracterizan. El usuario podrá asignar valores o restricciones sobre ellos en los parámetros.

Búsqueda de conceptos

Si en un portal de cualquier índole o tema en particular se busca la fecha "1980", no queda claro si es una fecha de nacimiento, cantidad aleatoria, o la fecha de algún evento. En un buscador semántico, al trabajar con un modelo de dominio se pueden especificar que acotamos la búsqueda a eventos que comenzaron en el año de 1980 obteniendo como resultados instancias de estos eventos con enlaces a documentos referenciados.

El mero hecho de poder restringir la búsqueda a "Eventos", constituye un gran avance, puesto que un buscador tradicional recuperaría todos los documentos con esta palabra, sin tener en cuenta si es una fecha, un código o un precio. Algunos buscadores permiten estos tipos de restricciones, pero en estos casos están codificadas en el programa del interfaz. Mediante las ontologías se representa este conocimiento de manera explícita.

Búsqueda por relaciones

Entre los conceptos existen varias relaciones que pueden admitir criterios sobre sus atributos o servir de enlace entre criterios de búsqueda. De esta manera, se puede buscar

un concepto que esté relacionado con otro mediante una relación con una determinada característica.

¿Qué profesores trabajan en el departamento de Matemáticas en Esime Zacatenco?

Otro tipo de búsquedas que permiten estos tipos de formalismos son la recuperación de relaciones fijando los criterios sobre los conceptos que las componen. De esta manera, es posible preguntar sobre todas las relaciones entre dos entidades, como por ejemplo:

Que tipos de planes de estudio tienen suscritos el IPN y la UNAM

Axiomas

Una ontología también incluye reglas de inferencia –axiomas- que permiten inferir conocimiento nuevo, sin estar explícitamente escrito. De esta manera, la información almacenada constituye solamente una base de hechos adecuada para inferir mucha más información de la presente.

Por ejemplo, si tenemos una relación transitiva definida entre escuelas como el IPN y la UNAM, como la homologación del bachillerato. En este caso, cuando se añade una escuela nueva en la ontología, solo se es necesario crear una instancia de relación de homologación de bachillerato y mediante axiomas enunciar la transitividad de la relación:

R = homologación del bachillerato

$$aRb \wedge bRc \Rightarrow aRc$$

(Si la escuela 'a' tiene una homologación de bachillerato con escuela 'b' y país 'b' la tiene con país 'c' entonces también existe una homologación de bachillerato entre las escuelas 'a' y 'c')

Para el usuario será absolutamente transparente el tipo de búsqueda que esté realizando, pudiendo además combinar varias con un único interfaz. Así, por ejemplo, se puede buscar por una relación no explicita, restringiendo algunos valores de los atributos de los conceptos colindantes y añadiendo a su vez algunas restricciones sobre el propio concepto buscado.

4.2. Portal Semántico

Como caso de estudio de la aplicación del sistema de adquisición y relleno se presenta un portal semántico sobre el dominio de cualquier tema en general. Este caso muestra el uso del sistema propuesto para adquirir contenido semántico para un portal de cualquier tema que desee el usuario. Como se ha comentado anteriormente, es un ejemplo que se

corresponde al primer tipo de aplicaciones dentro de la Web Semántica que permiten mostrar el contenido con funcionalidades avanzadas de búsqueda. El sistema de adquisición propuesto se encarga de agregar y convertir la información de varias fuentes online para almacenarla en la ontología de dominio.

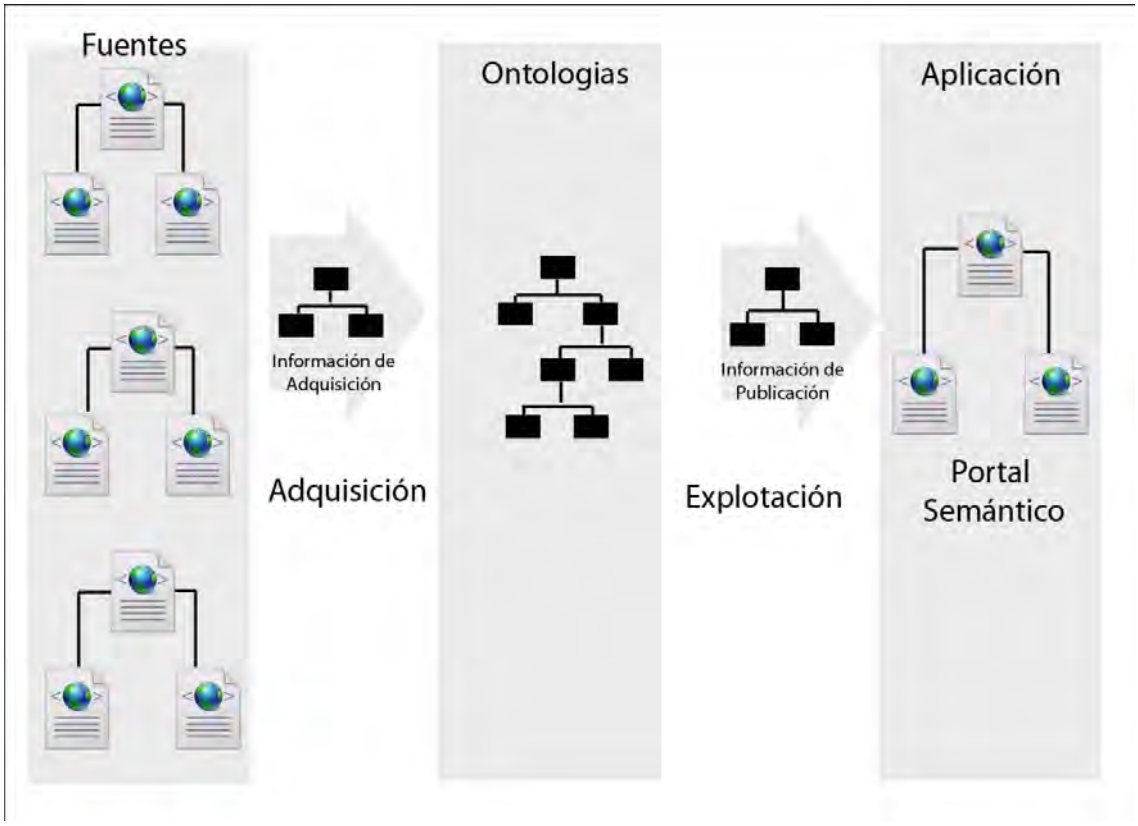


Figura 5 Arquitectura conceptual de un portal semántico alimentado automáticamente

A continuación se mostrarán con más detalle las distintas partes del sistema que se propondrá, comenzando por la parte central formada por la ontología de dominio del tema que se desee abordar y las fuentes online. Se mostrarán ejemplos de la ontología de adquisición que describe los datos en las fuentes online junto con algunos bosquejos del funcionamiento que se pretende del sistema que la alimenta (flecha izquierda de la figura 5). Por último se mostraran algunos ejemplos del resultado esperado de la publicación y búsquedas semánticas (flecha derecha de la figura 5).

4.2.1. Propuesta de Construcción de la Ontología de Dominio

En labores de documentación y análisis y más concretamente en el ámbito de cualquier área, se manejan grandes cantidades de información en forma de texto libre procedentes de diversas fuentes y con distintos formatos. Para su uso y explotación es esencial que la información sea de calidad y no introduzca ruido innecesario en forma de datos no

relevantes. Como son las fuentes de consulta utilizadas sin estructura definida, paginas de la WWW o bases de datos documentales. A menudo surge la necesidad de disponer de tesauros que agrupen información sobre un dominio determinado y que permitan la consulta rápida que proporcione información relevante.

En la sociedad de la información, con la tecnología informática disponible, se están ofreciendo soluciones que podemos clasificar en dos grupos, según la estrategia de codificación de los datos útiles.

- Por una parte se almacenan los textos en formato electrónico, en la misma forma que lo realizan los gestores documentales. Estos gestores indexan el contenido permitiendo una búsqueda rápida por palabras, basada en el encaje de patrones léxicos. Las ventajas que ofrece son que, una vez encontrado el texto deseado, lo ofrecen en la forma original de la fuente. Sin embargo, las búsquedas se realizan por encaje de patrones léxicos, es decir, por aparición de palabras, sin tener en cuenta su significado, por lo que suelen ofrecer resultados que no tienen por qué corresponder con lo deseado por el usuario.
- Una segunda alternativa la ofrecen las bases de datos relacionales que modelan la información en tablas y permiten indexar el contenido de los campos. De esta forma el usuario dispone de datos estructurados almacenados en tablas de rápido acceso. El éxito de este formalismo reside en la correcta formalización del modelo, que no siempre es consensuada ni refleja la realidad, y en una eficiente y usable interfaz que refleje los deseos de funcionalidad del usuario final. Este conocimiento suele estar codificado en la propia aplicación que permite el acceso sin estar n ser objeto de mantenimiento.

El uso de la ontologías para mejorar la recuperación de contenido de humanidades, como podrían ser en cualquier otra área, está empezando a dar algunos frutos en estos momentos, como han demostrado algunos experimentos en material fotográfico (Schreiber et al, 01), y mobiliario de época (Wielinga et al, 01), trabajos basados en tesauro AAT (Art and Architecture Thesaurus) (AAT).

La construcción del dominio en particular que se desee es una tarea muy costosa. Existen varios tesauros escritos y publicados que facilitarán este trabajo y servirán de base para su construcción. Los tesauros deben contener una estricta jerarquía de clases, deben estar basados en conceptos únicos, en vez de términos de lenguaje natural y deben poder ser representados mediante lenguajes consensuados de descripción semántica (como parte de la Web Semántica). La construcción de las ontologías se puede basar en la estructuración de tesauros, así como en fuentes externas (WordNet, CyC, SUO, etc.) para la creación del modelo final de dominio que se desee atacar.

A continuación se muestran alguna de las ontologías que se podrían tomar en cuenta en la construcción de algún dominio que se desee atacar.

Modelos de propósito general de alto nivel

- **WordNet:** (Miller, 95): Un léxico de la lengua inglesa, que organiza las palabras según su función gramatical, ligándolas entre sí relaciones semánticas. Se enumeran a continuación algunas básicas:
  - Sinonimia: Términos con significado equivalente.
  - Antonimia: Términos con significado opuesto
  - Hiponimia/Hiperonimia: Relación de generalización/ especialización: Un árbol es hiperónimo de pino y el pino es hipónimo de árbol.
- **EuroWordNet** (EWN): Una base de datos para algunos idiomas europeos. EuroWordNet amplia en número de relaciones semánticas codificadas en WordNet y además incluyen un formalismo común a los idiomas europeos que permite traducciones directas de ontologías.
- **SUO: Standard Upper Ontology** (SUO)**:** Ontología formal para aplicaciones de intercambio y extracción de información. Soportada por el organismo IEEE, usa como formato KIF (Knowledge Interchange Format).
- **SUMO: Suggested Upper Merged Ontology** (SUMO): Subconjunto de la ontología definida en SUO.
- **Generalized Upper Model** (GUM): Ontología desarrollada dentro del proyecto KOMET del centro de investigaciones alemán en Darmstadt. Al igual que WordNet es una ontología motivada por y para el procesamiento lingüístico.
- CyC (Lenat 95): Ontología con aproximadamente 3000 términos publicados que cuenta con millones de axiomas lógicos, desarrollada durante más de doce años.

Modelos básicos: Tiempo, Espacio

- **Ontología del Tiempo**: TimeML (Pustejosky et al 03) Desarrollada en la universidad de Brandeis en los EEUU dentro del grupo de James Pustejosky propone una especificación para anotaciones de eventos y expresiones temporales.
- **Ontologías del espacio:** (COBRA-ONT Space) basada en un subconjunto de la ontología de CyC para el espacio contiene conceptos para representar.

Modelos de dominios similares

- **Ontología geopolítica:** (CIA FACT BOOK DAML) contiene una representación semántica de la situación geopolítica del mundo.

- **Ontología de gobiernos:** (Teknowledge) ontología similar a la anterior centrada en la composición de los gobiernos.

Algunos repositorios de ontologías

- **Ontolingua:** http://www.ksl.stanford.edu/software/ontolingua: Conjunto de herramientas y servicios que permiten un desarrollo, uso y modificación colaborativos de ontologías.
- **DAML Library**: http://www.daml.org/ontologies. Listado de ontología definidas en el lenguaje DAML, organizadas por diferentes criterios de búsqueda. Contiene unas 170 ontologías.
- **UNSPSC**: www.unspsc.org Modelos que permite clasificar e identificar artículos de venta.
- **RosettaNet**: www.rosettanet.org Organización no lucrativa, que promociona la utilización y creación de estándares de intercambio de información en el cambio de comercio electrónico.

4.2.1.1. Preguntas de aptitudes

Las preguntas de aptitudes (Competency Questions) permiten acotar el dominio estudiado y determinar su grado de detalle, haciendo que sea capaz de modelar tanto las preguntas como respuestas. A continuación se muestra una lista abreviada de posibles preguntas:

1. ¿Países en guerra con Estados Unidos?
2. ¿Presidente del gobierno en México?
3. ¿Qué tipo de gobierno tiene Estados Unidos?
4. ¿Cuántos habitantes tiene Inglaterra?
5. Todos los miembros del gobierno de países de Latinoamérica
6. Países del Tratado del libre comercio

4.2.1.2. Resultado: Ontología de Dominio

Una vez estudiado y acotado el dominio (por ejemplo en economía) que se desee, se puede proceder a la construcción de una jerarquía sobre la relación de especificación (o generalización) de los conceptos. Para lo cual, se puede utilizar como entrada las propuestas existentes en dominios similares y una lista de términos relevantes.

El árbol del modelo semántico visto a lo largo de la relación de herencia es la siguiente:

- **Economía**: Clase que incluye la información relacionada con la economía de un país como es la tasa de desempleo, moneda, deuda externa, etc.
- **Sociedad:** Clase que representa a la sociedad de un país, la pirámide de población, la edad media de la población de un país, etc.

- **Agentes:** Clase que incluye a las personas u organizaciones que tienen capacidad de producir cambios en el dominio. Esta clase, como se puede ver en la jerarquía de actividades, se especializa en dos clases, Persona y Organización.
    - **Organización:** Clase que representa el concepto de organización o institución corporativa o similar, teniendo sus integrantes un propósito o función común.
        - **Diplomática:** Hace referencia a todas las organizaciones de carácter diplomática ya sean embajadas, consulados, etc.
        - **Gobierno:** Representa el concepto de organización gubernamental.
        - **Organización Internacional:** Esta clase representa a todas las organizaciones de ámbito internacional formada por países y cuyo ámbito de actuación es internacional.
    - **Persona:** Modela a un agente humano. Todas las personas relevantes del dominio deben de tener una instancia de esta clase.
- **Lugar**: Clase que hace referencia a un lugar geográfico (país, ciudad, etc.), como a emplazamientos de eventos, por ejemplo Edificios. Esta clase se especializa en cinco clases: Ciudad, Continente, Edificio, País y Región.
- **Ejercito**: Clase que incluye la información relacionada con la estructura militar de un país, el gasto militar en cuanto a capital humano.
- **Acuerdos:** esta clase hace referencia a los acuerdos, tratados, etc. que se dan entre los distintos organismos.
- **Eventos:** Clase que hace referencia a todos los posibles eventos que se pueden dar en el ámbito del dominio económico. Estos eventos van desde ataques terroristas, conferencias, crisis hasta guerras.
- **Relaciones:** La clase Relaciones permite modelar relaciones entre clases del dominio. Por "relación" entendemos un vínculo establecido entre dos ó más clases de dominio (por ejemplo: "ser secretario general del Instituto Politécnico Nacional" es la relación entre la clase persona y el concepto organización). Este vínculo se puede entender como un atributo cualquiera de los conceptos que participan en la relación. Pero a diferencia de los atributos, las relaciones al mismo tiempo son un concepto del dominio, estas se pueden caracterizar por sus propios atributos o participar en jerarquías.
    - **Relación Agente Participa:** Relación que modela la participación de cualquier persona u organización (gubernamental, diplomática, etc.) en un determinado evento, ya sea este una conferencia, una crisis etc. Ejemplo México participa en la Conferencia Internacional de Sida.

- **Relación Autoría Evento:** Relación que modela la ejecución por parte de un agente de un evento (por ejemplo: grupo terrorista ejecuta un ataque terrorista).
- **Relación Evento con Evento:** Relación que modela las relaciones entre distintos eventos, por ejemplo, el terremoto en México de 1985 origino la inflación en el mismo.
- **Relación Pertenencia:** Relación general de pertenencia.
- **Relación Suscribir Acuerdos:** esta relación representa qué acuerdos han sido suscritos por determinados agentes ya sean: personas, organizaciones o países. Un ejemplo de este tipo de relación es México suscribió el acuerdo del Tratado de Libre Comercio con la Unión Europea.
- **Relación Tiene Gobierno:** Relación que representa los distintos gobiernos que ha tenido un país, por ejemplo, las legislaturas.

Como herramienta de construcción se ha usado el editor Protege exportando en formato RDF(S).

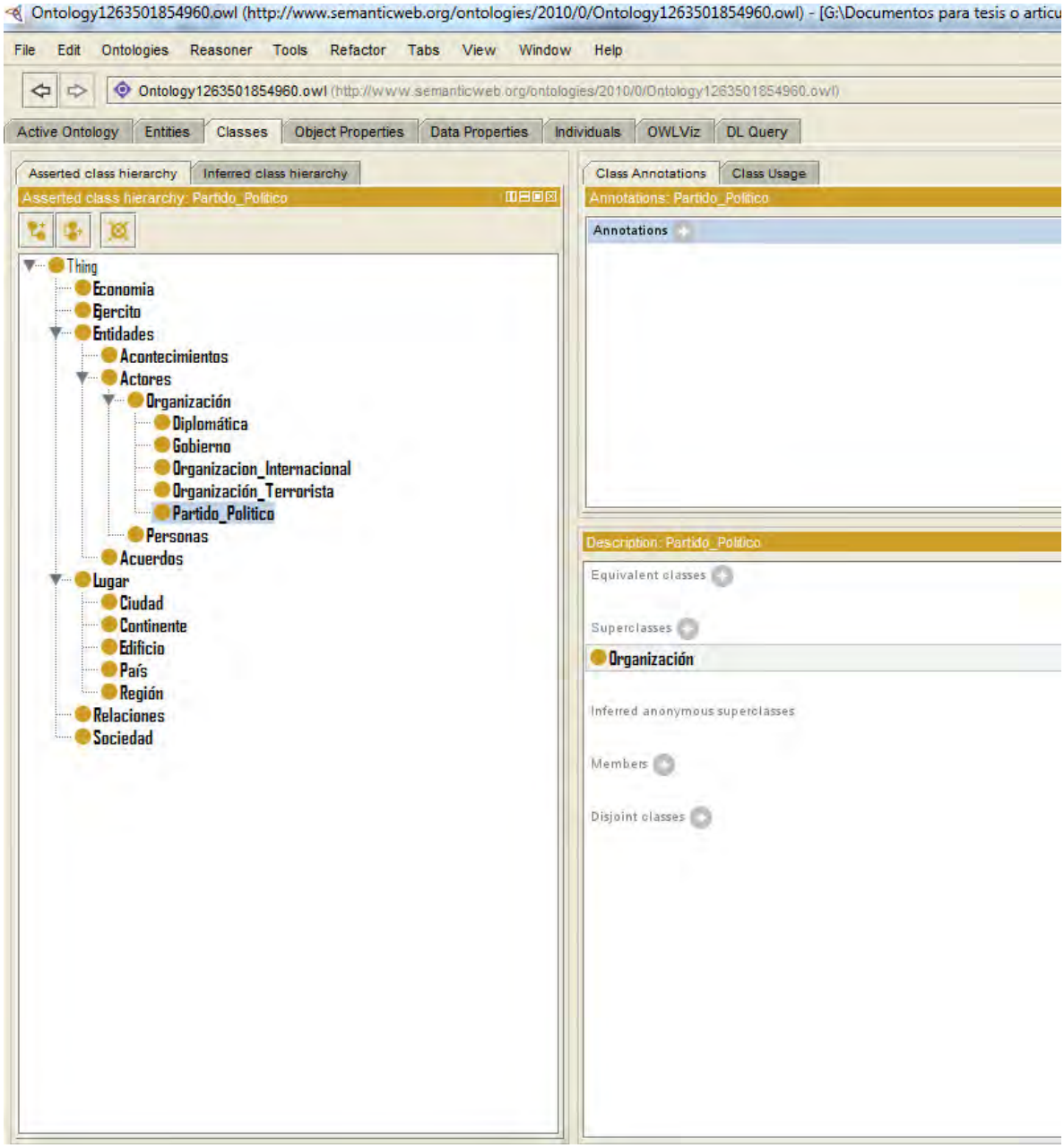


Figura 6 Ejemplo de la ontología de dominio en el editor Protégé

4.2.2. Fuentes Online Disponibles

Se han tomado en cuenta varias clases de fuentes para la agregación e integración de información en torno a un portal de relaciones internacionales.

CIA World Factbook. Web de la Agencia Central de Inteligencia de los EEUU que agrega información sobre la situación geográfica en el mundo. Su primera publicación fue en el año 1962 (como material clasificado, se hizo público en el año 1971), y desde entonces se

actualiza de manera periódica. Fue en el año 1997 cuando se hizo una versión online por Internet de esta información.

A continuación se muestran algunos ejemplos:

Figura 6 Página del portal de la CIA: World Fact Book

Figura 7 Documento sobre México en la CIA

Páginas de **NationMaster**, una Web con propósito educativo con datos geográficos del mundo que a su vez agrega información de varios sitios, generando gráficos comparativos, estadísticas, etc. En vez de descripciones absolutas, tal y como lo hace la Web de la CIA como se muestra en la figura 8 y 9.

Figura 8 Documento principal del portal NationMaster

Páginas del **Real Instituto Elcano:** organismo español que agrupa a expertos en el dominio de relaciones políticas, pública, análisis sobre eventos y situaciones políticas internacionales. La misión del Real Instituto Elcano supone un punto de partida para desarrollar los siguientes objetivos:

- Analizar el escenario internacional, con el fin de elaborar y producir análisis, estudios e informes con los que contribuir a la toma de decisiones.
- Difundir esos estudios y análisis, con la meta de conformar y participar en el debate público y social tanto nacional como global.
- Servir de foro de encuentro y debate, garantizando así una mayor y mejor comunicación entre agentes públicos y privados en el ámbito de las relaciones internacionales de seguridad.
- Aglutinar a su alrededor los programas, proyectos e ideas de la comunidad estratégica española, y en la medida posible, de la internacional.

North America > Mexico

View full size

FACTS AND FIGURES

- Age distribution
- Agriculture (274)
- Background (4)
- Crime (55)
- Currency (10)
- Democracy (32)
- Disasters (4)
- Economy (1361)
- Education (220)
- Energy (656)
- Environment (85)
- Food (13)
- Geography (48)
- Government (96)
- Health (165)
- Identification (20)
- Immigration (27)
- Industry (48)
- Internet (33)
- Labor (132)
- Language (10)
- Lifestyle (11)
- Media (185)
- Military (71)
- Mortality (2631)
- People (280)
- Religion (22)
- Sports (415)
- Taxation (36)
- Terrorism (26)
- Transportation (133)
- Top Rankings
- Bottom Rankings

View full size
(54 more maps)

BACKGROUND:
The site of advanced Amerindian civilizations, Mexico came under Spanish rule for three centuries before achieving independence early in the 19th century. A devaluation of the peso in late 1994 threw Mexico into economic turmoil, triggering the worst recession in over half a century. The nation continues to make an impressive recovery. Ongoing economic and social concerns include low real wages, underemployment for a large segment of the population, inequitable income distribution, and few advancement opportunities for the largely Amerindian population in the impoverished southern states. The elections held in 2000 marked the first time since the 1910 Mexican Revolution that an opposition candidate - Vicente FOX of the National Action Party (PAN) - defeated the party in government, the Institutional Revolutionary Party (PRI). He was succeeded in 2006 by another PAN candidate Felipe CALDERON.

BORDERS:
Belize 250 km, Guatemala 962 km, US 3,141 km

POPULATION:
109,955,400

GDP PER CAPITA:
$8,051.92 per capita

Figura 9 Documento sobre México en NationMaster

Figura 10 Página principal del Real Instituto Elcano

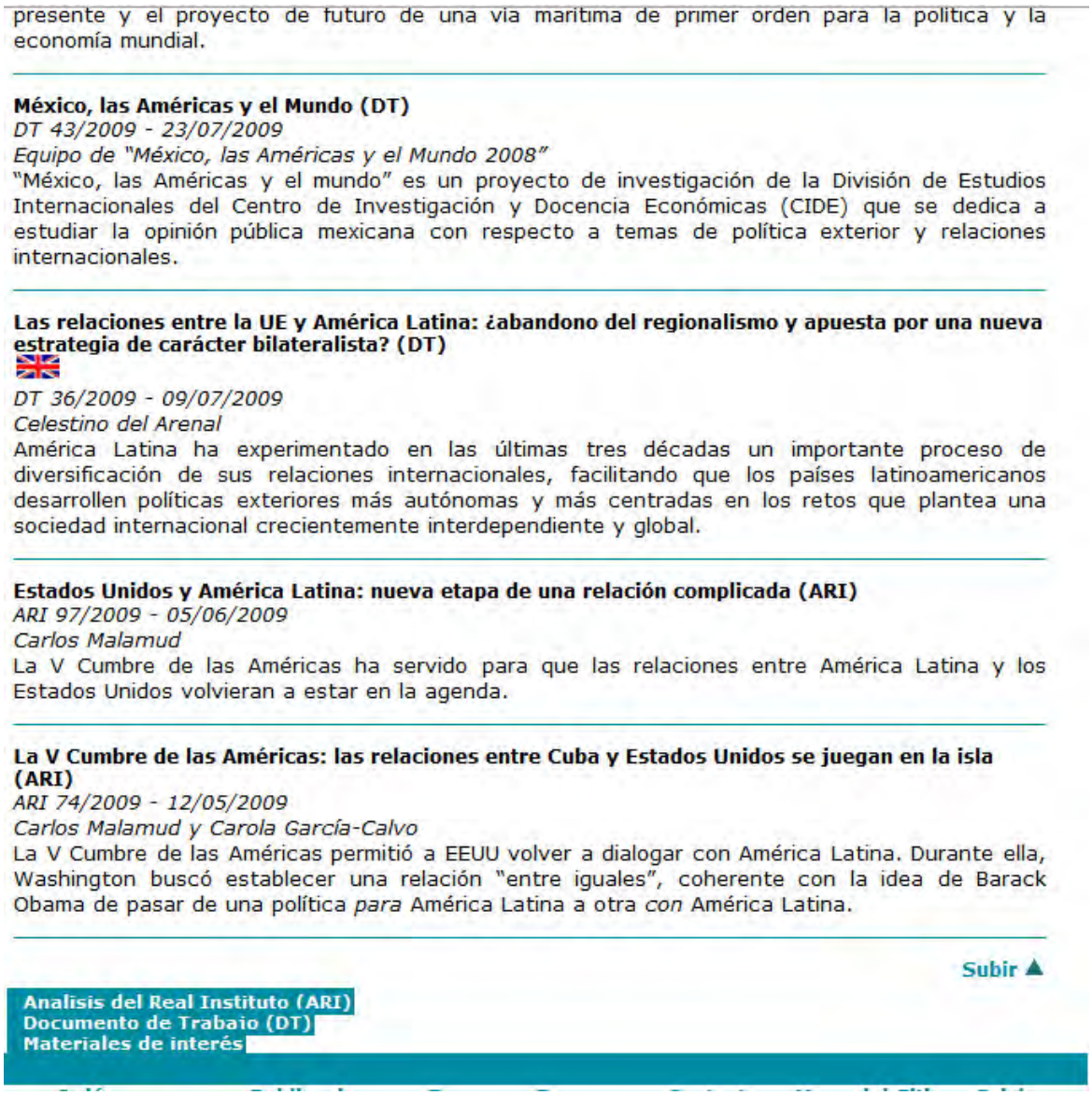


Figura 11 Detalle de un análisis ubicado en el Real Instituto Elcano

4.2.3. Ontología de Adquisición

En esta sección se mostrarán algunos ejemplos de especificación de la información necesaria para la identificación y relleno de datos a partir de las fuentes seleccionadas. Dadas las tres fuentes identificadas el objetivo del sistema es rellenar la ontología de dominio con datos a partir de las fuentes más fuertemente estructuradas (CIA World Factbook y NationMaster) para luego, aprovechando la información adquirida procesar documentos de análisis político, económico, cultural etc., del Real Instituto Elcano.

La ontología de adquisición contiene la información sobre los tipos de documentos y piezas de información que se pueden encontrar en las fuentes identificadas. Existen 7 clases de documentos:

- **Documento principal del CIA World Factbook** (figura 6): Representa el documento ubicado en la URL principal que sirve de punto de entrada al portal de la CIA. Desde allí se podrá seleccionar en una lista desplegable cualquier país del mundo para obtener información.
- **Documento principal del portal NationMaster** (figura 8): Documento de entrada al portal de NationMaster, con URL fija. Existe mucha información que se visualiza en esta página. Solamente se tendrán en cuenta las distintas regiones América del Sur, América del Norte, América central, Europa, Asia, etc., y a través de ellas llegar a los documentos descriptivos de los países.
- **Documento principal del Real Instituto Elcano** (figura 10): Punto de entrada al Real Instituto Elcano, a través del cual se podrán acceder los distintos análisis políticos. La URL de este documento es conocida.
- **Documento descriptivo de un país en el portal de la CIA** (figura 6): Las descripciones de los países en el portal de la CIA son documentos organizados por secciones donde cada dato viene precedido por una etiqueta, conocida de antemano. La URL de cada país no se conoce, pero el sistema podrá identificar cada uno de ellos gracias a los enlaces (definidos como piezas más adelante) ubicados en la página principal.
- **Documento descriptivo de un país en el portal NationMaster** (figura 8): descripción de un país con algunos datos interesantes. También está organizado en forma de tabla visual, es decir, cada dato viene precedido por una etiqueta explicativa. La URL se averigua durante la ejecución de la extracción.
- **Documento descriptivo de una región:** La organización de la información en el portal NationMaster es a través de regiones. Un navegante puede seleccionar primero una región, y dentro de la región puede enlazar con un país perteneciente. Como existen varias instancias de documentos de regiones, la URL se averigua en tiempo de extracción.
- **Documento descriptivo de un análisis** (figura 11): El Real Instituto Elcano publica análisis que son accesibles desde la página principal por medio de menús.

Es importante que en cada portal o conjunto de documentos online que se agregue como posible fuente exista al menos un documento con su URL conocida. De esta manera, el sistema podrá comenzar la exploración desde allí e identificará a otros documentos mediante relaciones definitivas entre ellos. En la aplicación propuesta, existirán tres portales distintos que agrupen los documentos fuente, y es por ello que se modelaran en

la propuesta de construcción, tres documentos con instancia única y URL conocida. Gracias a la arquitectura que se propone posible añadir operadores que podrán buscar documentos de alguna clase ya definida mediante diversas técnicas existentes.

El dominio que puede ser de cualquier índole, en nuestro caso por las fuentes seleccionadas puede ser los dominios económico, político y cultural, por lo tanto de las fuentes online tomadas en consideración para una propuesta de aplicación, dan lugar a las siguientes piezas de información extraíbles (solo se muestran algunas de ellas como ejemplo ilustrativo, pero pueden ser más):

- **Nombre de País**: esta pieza suele estar contenida en las páginas principales que sirven de menú hacia los distintos países en detalle.
    - Es nombre propio (desde el punto de vista de la interpretación de lenguaje natural.
    - Es un elemento HTML ejecutable (es decir: un enlace, un formulario, etc.).

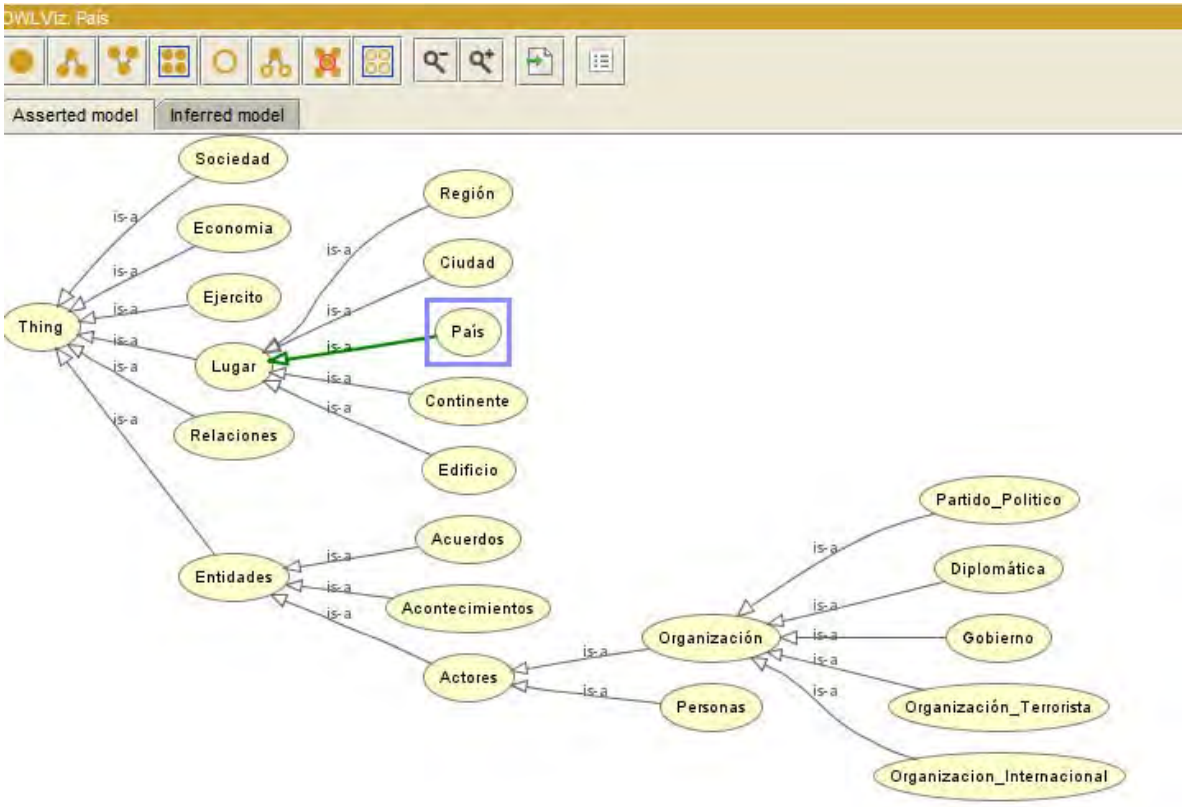


Figura 12 Detalle de la descripción de la ontología

- **Etiqueta de máximo representante de un país:** En las páginas de la CIA existe información sobre la composición del gobierno de cada país. Este dato es muy interesante para ser usado en el procesamiento de los análisis del Real Instituto Elcano.
- **Nombre del máximo representante de un país:** Pieza que describe el nombre de una persona (la interpretación de procesamiento de lenguaje natural deberá probar que es un nombre propio).

4.2.4. Proceso de Extracción

Las distintas estrategias tienen por objetivo secuenciar la ejecución de los operadores con el objetivo de generar una hipótesis sobre posibles rellenos en la ontología de dominio. En la propuesta de implementación del portal semántico para cualquier dominio que se desee en particular, se ha optado por la utilización de una estrategia de búsqueda con retroceso aumentada con alguna heurística para la optimización del proceso de relleno, disminuyendo el número de hipótesis creadas, y con ello perdiendo algunas soluciones posibles. En esta sección se presenta parte de una ejecución de la extracción sobre un portal Web a modo de ejemplo:

El proceso comienza con el procesamiento del contenido de la Web de Prueba, donde localiza la página inicial (home page). En esta página se localizan el nombre del objeto buscado, descritos como una pieza contenida en una lista despegable (combo box) que sirven de enlace, a través de una relación pieza-documento a las páginas descriptivas de la palabra computational. La ontología de adquisición especifica que la pieza de nombre del objeto en la página principal tiene una cardinalidad de (0: 300). El sistema encuentra 3 candidatos para la pieza de nombre computational, todas ellas incluidas en la lista despegable. Al ser el número de candidatos menos que la cardinalidad máxima permitida, el sistema tiene una única hipótesis sobre el documento donde incluyen todos los candidatos hallados como posible asignación a la pieza de nombre computational.

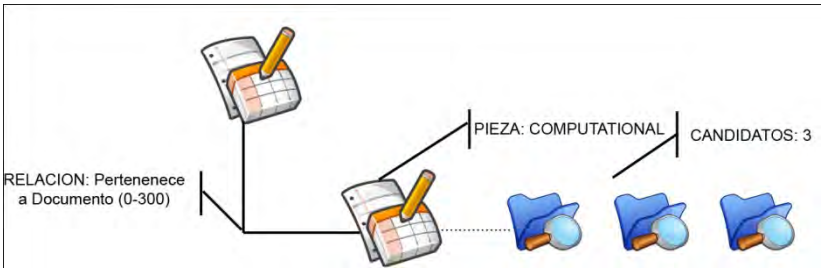


Figura 13 Especificación de la pieza nombre computational

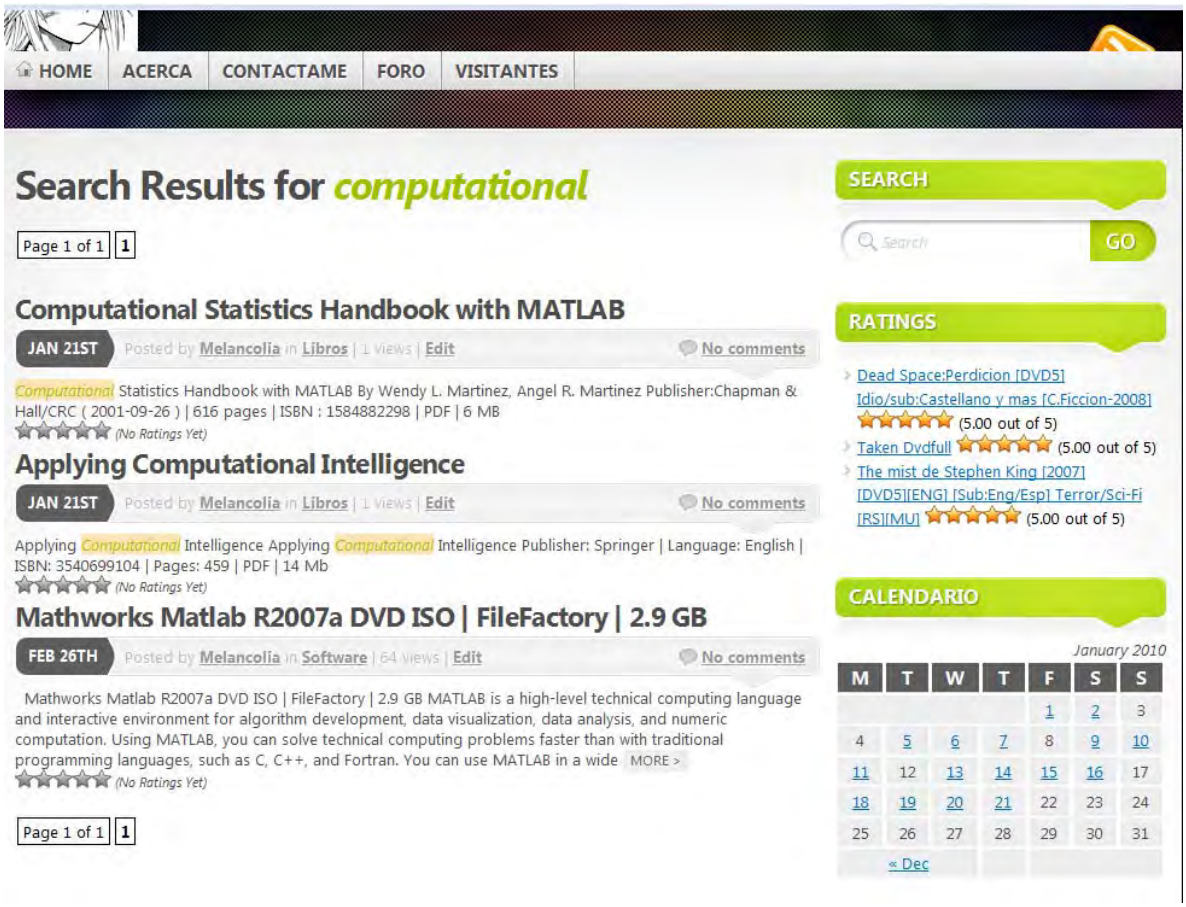


Figura 14 Búsqueda dentro del portal de prueba la palabra computational

En este paso existe una posible pérdida de información en el proceso de construcción de la hipótesis. En el caso de encontrar un número de candidatos menor que la cardinalidad máxima permitida de acuerdo a la estrategia de búsqueda con retroceso pura, deberían generarse tantas hipótesis como posibles combinaciones de asignación de los candidatos a la pieza, es decir N!(N factorial, siendo N el número de candidatos encontrado).

La estrategia propuesta para este dominio asume la posible pérdida de información, mejorando así la eficiencia de la recuperación tanto en tiempo como en recursos consumidos. La estrategia pura de búsqueda en anchura sería inviable en este caso, con 286! hipótesis creadas. Esta modificación en el algoritmo original es posible al alto grado de estructura presente en las fuentes.

Para la palabra computational localizada en el portal se procede a navegar hacia la página de descripción detallada. En esta página se localizan las piezas descritas en la ontología de adquisición. La descripción consta de un conjunto de piezas formadas por una etiqueta que está en línea visual con un dato buscado. Esta descripción permite una eficiente extracción a partir del documento gracias a su estructura bien definida.

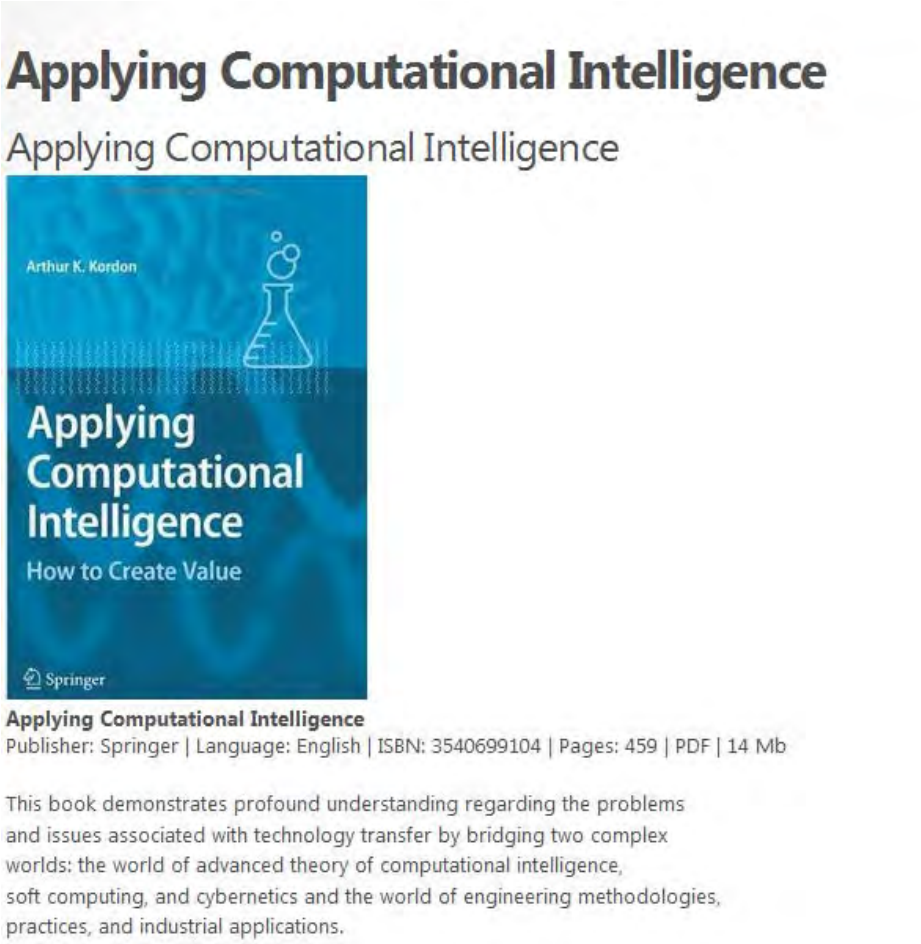


Figura 15 Parte del documento detallado de la palabra computational

Finalmente, tras procesar la Web de prueba el sistema obtiene 3 resultados posibles de los cuales se elije el que se desea con todos sus atributos.

CONCLUSIONES

La intención de este trabajo es contribuir a la superación de uno de los retos más importantes identificados en la consecución del éxito de la Web Semántica. Este trabajo comienza con una definición y descripción sobre cuál es la visión de la Web semántica. Cabe destacar que es una iniciativa sin precedentes sobre el propósito de la creación de una base de conocimientos global, formal y distribuida, equivalente a la WWW, pero a diferencia de está, la Web Semántica está definida para el proceso y consumo por parte de aplicaciones de software.

Desde los comienzos de la ciencia de la Inteligencia Artificial ha sido un problema la disponibilidad de conocimiento en forma estructurada para que las aplicaciones puedan procesarlo y realizar tareas cada vez más sofisticadas. Algunos autores ven en la falta de procedimientos automáticos de provisión de este contenido formal la causa de no proliferación a una amplia escala, de sistemas basados en el conocimiento y de agentes inteligentes. Con la visión de la Web Semántica se pretende crear un repositorio de contenido procesable automático que permita la aparición de este tipo de aplicaciones avanzadas y con esto emular el éxito de hace algunos años de la WWW actual cuando se convirtió en un artículo de uso común.

Uno de los focos más prometedores en la consecución de esta tarea en la conversión de documentos online, que actualmente constituyen la Web, en contenido semántico estructurado apto para procesamiento automático. El reto de la provisión automática de contenido a partir de documentos electrónicos se enmarca, desde un punto de vista tecnológico, en el área de extracción de información. La larga experiencia en este campo, tanto de investigaciones, como en aplicaciones existentes demuestra que el éxito y la eficiencia están estrechamente ligados al dominio sobre el cual se trabaja, el tipo de contenido que se trata y las tecnologías aplicadas para la extracción. Se observa que para documentos con alto grado de presencia de estructuras y dominios acotados, las técnicas de tratamiento de cadenas textuales tiene un éxito razonable alcanzando cotas de eficiencia excelentes. Así mismo para documentos con poca estructura, pero sí con presencia de formas lingüísticas completas (frases, párrafos, etc.) se hace necesario el uso de técnicas de procesamiento de lenguaje natural.

Este trabajo propone una arquitectura que permite implementar sistemas en los cuales las distintas tecnologías se unen en la consecución de la tarea de obtención de contenido semántico. El proceso de cooperación entre ellas va dirigido por una estrategia, adecuando el proceso al tipo de fuente y su dominio. Se trata de una arquitectura extensible que modela el proceso de extracción en tres fases: pre-proceso de las fuentes según su interpretación, extracción de la información y formación de hipótesis sobre su

semántica y finalmente la fase de inserción. El diseño de la arquitectura de manera abierta permite incorporar nuevas interpretaciones de las fuentes así como nuevas estrategias de control en el módulo de extracción, adecuando el sistema completo a las necesidades de cada dominio o cada aplicación. La flexibilidad y apertura de la arquitectura ha sido un requisito esencial en su concepción para servir de plataforma de desarrollo y extensión del alcance de los sistemas finales en nuevas fuentes y nuevas estrategias de procesamiento y extracción de información.

Los sistemas de extracción de información necesitan de una aplicación concreta que explote los datos estructurados para demostrar su valor añadido. En la visión de la Web Semántica, donde los datos se estructuran de acuerdo a modelos semánticos, se esboza una evolución de las posibles aplicaciones. En la escala más baja se encuentran aplicaciones que permiten un acceso avanzado de información aprovechándose del modelo subyacente para la mejora de funcionalidades de búsqueda y presentación. Este trabajo propone la utilización de un portal semántico sobre el dominio de cualquier tema. Dentro del sistema del portal semántico se incorpora una propuesta de publicación de modelos semánticos que permite que usuarios humanos tengan acceso a contenido diseñado para proceso automático.

La propuesta de implementación de la aplicación que se pretende construir de acuerdo a la arquitectura propuesta, es la de implementar un software necesario para las interpretaciones de lenguaje, aspecto, estructura HTML y texto plano, todas ellas al servicio de una estrategia de búsqueda con retroceso en distintos dominios.

TRABAJOS FUTUROS

Se presentan algunas posibles líneas de continuación sobre el trabajo presentado. La propuesta de arquitectura hecha, ha tenido desde sus orígenes en cuenta el requisito de apertura hacia su expansión, inclusión de nuevas aproximaciones tecnológicas y aplicación en distintos dominios. Es allí donde se centran las posibles futuras condiciones y ampliaciones.

En una evolución natural de los sistemas planteados sobre la arquitectura propuesta se engloban las ampliaciones de las posibles interpretaciones de fuentes digitales:

- ***Interpretación de fuentes PDF:*** Permitirá procesar documentos en formatos PDF para obtener una interpretación de aspecto de lenguaje natural, DOM y texto plano. Hoy en día existe mucha información online en este formato, sobre todo en el campo de publicaciones científicas y administración pública.
- ***Interpretación de Bases de Datos:*** Ya se ha comentado que la parte de Internet almacenada en bases de datos es hasta de 500 veces más grande que su parte textual. La arquitectura aquí presente permitirá procesar y extraer información de las publicaciones obtenidas a partir de datos almacenados. Otra forma es acceder a los gestores de bases de datos (con lenguajes como SQL, XPath, etc.) para extraer información útil.
- ***Interpretación de Sistemas de gestión documental***: Son sistemas documentales de grandes organismos e instituciones, normalmente protegidos por una intranet, constituyen valiosas fuentes de información para ser estructurada de acuerdo a los estándares de la Web Semántica.
- ***Multimedia***: Cada vez más contenido de la actual WWW se crea en formatos multimedia. Técnicas de procesamiento de imágenes, documentos audio y video así como la aparición de estándares que permitan añadir meta-datos a las fuentes (por ejemplo MPEG-7[1]) podrán ayudar a esta tarea.

Así mismo la arquitectura soporta la inclusión de nuevas estrategias de extracción. En la presente memoria se pretende presentar la estrategia de fuerza bruta y dos variantes de

[1] Consiste en una representación estándar de la información audiovisual que permite la descripción de contenidos (metadatos) para: Palabras clave, Significado semántico (quién, qué, cuándo y dónde), Significado estructural (formas, colores, texturas, movimientos y sonidos). Es un estándar de la Organización Internacional para la Estandarización ISO/IEC y desarrollado por el grupo MPEG. El nombre formal para este estándar es Interfaz de Descripción del Contenido Multimedia (Multimedia Content Description Interface). La primera versión se aprobó en julio del 2001 (ISO/IEC 15938) y actualmente la última versión publicada y aprobada por la ISO data de octubre del 2004.

la estrategia de búsqueda con retroceso. Son ejemplos de los extremos de la relación entre calidad y eficiencia. Mientras que la primera, prima la eficiencia sobre la calidad de los resultados, la segunda prima la calidad del resultado en detrimento de la eficiencia. La variante de la estrategia de búsqueda con retroceso presentada, introducirá alguna heurística que permitirá alcanzar soluciones de calidad aceptable mejorando la eficiencia del proceso de extracción. Entre los extremos enunciados existe todo un abanico de posibles estrategias que mediante uso de heurísticas e información adicional pueden construir secuencias de extracción adecuadas para cada aplicación o dominio. Podemos incluir estrategias basadas en:

- ***Heurísticas empíricas o estadísticas:*** Basándose en algunos estudios previos sobre las estructuras frecuentes presentes en las fuentes, es posible diseñar estrategias que optimicen el proceso de extracción.
- ***Información adicional externa al sistema:*** En una aproximación promiscua a la tarea de extracción, el sistema podrá consultar fuentes externas, tales como buscadores de propósito general, portales especializados en el dominio tratado, para resolver posibles ambigüedades en la construcción y resolución de hipótesis.
- ***Interacción con el usuario:*** Una de las posibles fuentes de información que podrán permitir la desambigüación y optimización del proceso de extracción es la interacción con el usuario.

También se ha estudiado la posibilidad de construcción de sistemas de extracción sobre nuevos dominios. Entre las propuestas de continuación se proponen los siguientes:

- ***Dominio financiero:*** Actualmente existen varias propuestas comerciales de aplicaciones que permiten agregar información financiera y de consumo a partir de fuentes WWW. Su alto grado de presencia de estructuras de aspecto permite alcanzar altas cotas de eficiencia en las tareas de extracción [GETSee]. Su mayor problema es el costo de mantenimiento de los *wrappers* encargados en localizar y extraer los datos útiles. Son sistemas basados en gramáticas y expresiones regulares con poco grado de flexibilidad y muy sensibles a cambios en las fuentes. Con la arquitectura presentada en esta memoria aumentaría el grado de adaptación de estos sistemas gracias a la combinación de tecnologías y aun modulo de control dotado de estrategias.
- ***Cultural:*** El dominio cultural permitirá probar sistemas con esta arquitectura en fuentes con menos estructura de aspecto y con más posibilidades para las tecnologías de procesamiento de lenguaje natural. Existen ontologías de dominio de humanidades centradas en las relaciones de autores [Benjamins et al 04],

movimientos y obras que han realizado que constituye una base para la identificación de fuentes posibles para su relleno.

REFERENCIAS

[ATT] http://www.getty.edu/research/conducting_research/vocabularies/aat/

[AltaVista] www.altavista.com

[Ambite et al 98] Ambite, J-L.; Ashish N, Barish G.; Knoblock, C. A.; Minton, S.; Modi, P. J.; Muslea, I.; Philpot, A.; and Tejada, S. 1998. Ariadne: A system for constructing mediators for internets sources. In Proceedings of the ACM SIGMOD International Conference en Management Data

[Ankolekar et al, 02] A. Ankolelar et al., "DAML-S: Web Service Description for the Semantic Web," Proc. 1st Int'l Semantic Web Conf. (ISWC 02), 2002.

[Benjamins et al 04] Richard V. Benjamins, Jesús Contreras, Mercedez Blázquez, Juan Manuel Dodero, "Cultural Heritage and the Semantic Web", 1st European Semantic Web Sympsium (ESWS), Heraklion, Grecia, May 2004.

[Benjamins et al 99] Benjamins, V. R., Fensei, D., Decker, S. and Gomez, A. Perez: (KA)2: Building Ontologies for the Internet: a Mind Term Report. In the International Journal of Human- Computer Studies, 51:687-712, 1999

[Bergman 01] Michael K. Bergman "The Deep Web: Surfacing Hidden Value" White Paper http://www.brightplanet.com/deepcontent/tutorials/DeepWeb/index.asp

[Bikel 97] Bikel, D. M.; Miller, S.; Schwarts, R.; and Weischedel, R. 1997. Nymble: a high-performance learning name-finder. In Procedings of ANLP-97, 194-201.

[Bizer 03] Bizer, Christian: D2R MAP – A Database to RDF Mapping Language, WWW2003, The Twelfth International Worl Wide Web Conference, Budapest, HUNGARY, 2003.

[Bozak et al 02] E.Bozsak, Marc Ehrig, Siegfried Handschuh, Andreas Hotho, Alexander Maedche, Boris Motik, Daniel Oberle, Christoph Schmitz, Steffen Staab, Ljiljana Stojanovic, Nenad Stojanovic, Rudi Studer, Gerd Stumme, York Sure, Julien Tane, Raphael Votz, Valentin Zacharias "KAON-Towards a large scale Semantic Web " In Kurt Bauknecht and A. Min Tjoa and Gerald Quirchmayr. E-Commerce and Web Technologies, Third International Conference, EC-Web 2002, Aix-en-Provence, France, September 2-6, 2002, Proceedings, volume 2455 of Lecture Notes in Computer Science, pp. 304-313. Springer, 2002.

[Brickley et al 99] http://www.w3.org/TR/1999/PR-rdf-schema-19990303/

[Broekstra et al 01] J. Broekstra, A. Kampman, and F. van Harmelen. Sesame: An Architecture for Storing and Queryng RDF Data and Schema Information. In D. Fensel, J. Hendler, H. Lieberman, and W. Wahister, editor, Semantics for the WWW. Mit Press, 2001.

[Budanitsky et al 2001] Budanitsky, Alexander and Hirts, Graeme. "Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures". Workshop on WordNet and

Other Lexical resources, Second meeting of the North American Chapter of the Association for Computational Linguistics, Pittsburgh, June 2001.

[Bussier et al, 02] Christoph Bussler, Alexander Maedche, Dieter Fensel A Conceptual Architecture for Semantic Web Enabled Web Services ACM Special Interest Group on Management of Data: Volume 31, Number, 4 Dec 2002.

[Castells 04] Aplicaciones de Técnicas de la Web Semántica, Electronic Commerce: Research and Applications, Volume 40, June 2002

[Chomsky 55] Noam Chomsky. The logical structure of linguistic theory. Plenum, New York, 1955

[CIA FACT BOOK DAML] http://www.cia.gov/cia/publications/factbook/

[Ciravegna 01] Fabio Ciravegna: "(LP) 2, an Adaptative Algoritthm for Information Extraction from Web- related Texts " In proceedings of the IJCAI-2001 Workshop on Adaptive Text Extraction and Mining to be heid in conjunction with the 17th International Conference on Artificial Intelligence (IJCAI-01), Seattle, August, 2001.

[Ciravegena 01b] F, Ciravegna. Adaptative Information extraction from text by rule induction and generalization. In 17th International Joint Conference on Artificial Intelligence, 2001.

[Citeseer] www.citeseer.com

[COBRA-ONT Space] Http://cobra.umbc.edu/ontologies.html

[Cohen 99] William W. Cohen "Recognizing Structure in Web pages using Similarity Queries" Proceedings of AAAI 1999

[Cunningham 02] H. Cunningham. GATE, a General Architecture for Text Engineering.

[Cunningham et al 02b] H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan. Gate: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL '02). Philadelphia, July 2002 PDF

[D63 Esperonto] www.esperonto.net Deliverable: D6.3 Semantic Web content visualization services.

[DAML+OIL] http://www.daml.org/2001/03/daml+oil-index.html

[Dean et al 02] M. Dean, D. Connolly, F van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P.F. Patel-Schneider, and L. A. Stein. OWL web ontology language 1.0 reference, July 2002. http://www.w3.org/TR/owl-ref/.

[Declerck 02] Declerck Th. A set of tools for integrating linguistic and non-linguistic information. Proceedings of the SAAKM workshop at ECAI, Lyon, 2002.

[DOM] http://www.w3.org/DOM/

[Dublin Core] http://dublincore.org/

[eMarketer 02] www.emarketer.com

[Esperonto] http://www.esperonto.net

[EWN] http://www.illc.uva.nl/EuroWordNet/

[Fensel et al 98] Fensel, D., Benjamins. V.R., Decker, S., Gaspari, M., Groenboom, R., Motta, E., Plaza, E., Schereiber, G., Studer, R., and Wielinga, B. (1998): The Unified Problem-solving Method description Language UPML, Deliverable Esprit Project 27169, IBROW3, 1998.

[Fensel, Bussier 02] D. Fensel and C. Bussier: The Web Service Modeling Framework WSMF, Electronic Commerce: Research and Applications, 1, 113-137, 2002.

[Forrester 01] How The X Internet Will Communicate (Forester Research, December 2001, http://www.forrester.com/ER/Report/Summary/0,1338,13387,00.html

[Fredge 23] Fredge, G. (1923), Logishe untersuchungen. Dritter teil: Gedankenguge, in "Beitrage zur Philosophie des Deutschen idealismus", Vol. III, pp.36-51.

[Garcia-Serrano et al 00] A. Garcia-Serrano, J. Hernandez, Knowledge Modeling Techniques to Support Intelligent Assistance to e-commerce IFIP Working Group 7.6: Workshop on Virtual Environments for Advanced Modeling 2000.

[García-Serrano et al 98] García-Serrano A.,k Contreras J., A Computational Platform for Ugaritc Morphological Analysis, PROC. LREC, pp: 879-884, 1998, ELRA eds.

[GETSee] www.getsee.com

[Gilbert et al 95] Gilbert D.; Aparicio, M.; Atkinson, B.; Brady, S.; Ciccarino, J.; Grosof, B.; O'Connor, P.; Osisek, D.; Pritko, S.; Spagna R. and Wilson, L., IBM Intelligent Agent Strategy, IBM Corporation, (1995).

[Gomez-Perez 99 et all] Gomez-Perez, A., & Benjamins, R. (1999). Overview of Knowledge Sharing and Reuse Components: Ontologies and Problem-Solving Methods. In Proceedings of the IJCAI-99 Workshop on Ontologies and Problem-Solving Methods (KRR5), Stockholm, Sweden

[Gomez-Perez et al 96] Gomez-Perez, A.; Fernandez, M.; De Vicente, A., Towards a Method to Conceptualize Domain Ontologies, Workshop on Ontological Engineering, ECAI'96, Pages 41-51, 1996

[Goñi 98] José Miguel Goñi Menoyo: "Arquitectura para representación del conocimiento léxico en sistemas de procesamiento de lenguaje natural", Tesis Doctoral, Universidad Politécnica de Madrid, 1998.

[Google] www.google.com

[GoogleNews] news.google.com

[Gruninger et al 95] Gruninger, M. and Fox, M.S. (1995) Methodology for the Design and Evaluation of Ontologies. In: Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI-95, Montreal

[Guarino et al 99] Guarino, N., Masolo, C. and Vetere, G. "OntoSeek: Content-Based Access to the Web. "IEEE Intelligent Systems 14(3), 1999.

[GUM] John A. Bateman, Renate Henshel, Fabio Rinaldi "The Generalized Upper Model 2.0": 1995, IPSI, Darmstadt.

[HALO final report] http://www.projecthalo.com7content/docs/halopilot-vulcan_finalreport.pdf

[HALO] http://www.projecthalo.com

[Handschuh et al 01] S. Handschuh, S. Staab, and A. Maedche. CREAM- Creating relational metadata with a component-based, ontology driven framework. In Proceedings of K-Cap 2001, Victoria, BC, Canada, October, 2001

[Handschuh et al. 02] S. Handschuh, S. Staab, and F. Ciravegna. S-cream- semi-automatic creation of metadata. In 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW02), October 2002.

[Hoog et al 02] Robert de Hoog, Bob Wielinga, Suzanne Jabel, Anjo Anjewwierden, Frans Verster, Yvonne barnard, Paola DeLuca,, Cyrille Desmoulins and Johan Riemersma. "Re-using technical manuals for instruction: document analysis in the IMAT project". In Workshop on Integrating Technical and training documentation, ITS 2002, San Sebastian, June 2002.

[Horrocks et al 00] Ian Horrocks, A denotational semantics for Standard OIL and Instance OIL, 2000, http://www.ontoknowledge.org/oil/downl/semantics.pdf

[Hotbot] http://serarchenginewatch.com/sereport/00/07-hotbot.html

[IceSoft, WebRenderer] www.icesoft.com; www.webrenderer.com

[IMAT] http://imat.swi.psy.uva.nl/

[JENA] http://www.hpl.hp.com/semweb/jena2.htm

[Karvounarakis et al 02] G. Karvounarakis, S. Alexaki, V. Christophides, D. Plexousakis, and M. Scholl. RQL: A Declarative Query Language for RDF. In The 11th Intl. World Wide Web Conference (WWW2002)

[Kiraz 95] George Anton Kiraz: Multi-Tape Two-Level Morphology: A Case Study in Semitic Non-linear Morphology 1995

[Knoblock et al 01] Knoblock, CA., Minton, S., Ambite, J.L. Muslea, M., Oh, J., Frank, M.Mixed. Iniative, Multi-source information Assistants. The Tenth International Worl Wide Web Conference (WWW10). Honk Kong 2001.

[Kobayashi 00] M. Kobayashi and K. Takeda, Information retrieval on the Web, IBM Research Report, RT0347, April 2000.

[Koster 94] Koster M. "A Standard for Robots Exclusion" http://www.robotsxt.org/wc/norobosts.html

[Kupiec 92] Kupiec, L. 1992. Robust part-of-speech tagging using a hidden Markov Model. Computer Speech and Language 6:225-242.

[Kushmerick 97] N.Kushmerick "Wrapper induction for information extraction" PHD dissertation Department of Computer Science and Engineering, University of Washington, Seatle.

[Lasie] http://www.dcs.shef.ac.uk/npl/funded/lasie.htm

[Lassila et al 97] http://www.w3.org/TR/WD-rdf-syntax-971002/

[Lawrence, Giles 99] S. Lawrence and C.L. Giles, "Accessibility of Information on the Web", Nature 400:107-109, July 8, 1999

[Lenat 95] Lenat, D.B. (1995). CYC: A large-scale investment in knowledge infrastructure. Communications of the ACM, 38(11).

[Looksmart] http://search.looksmart.com/

[Lucene] http://jakarta.pache.org/lucene/docs/index.html

[Lycos] www.lycos.com

[Miller 95] George A. Miller. Wordnet: a lexical database for English. Communications of the ACM, 38(11):39-41, November 1995. 129

[Motta 93] John Domingue, Enrico Motta, and Stuart Watt. The emerging VITAL workbench. In Knowledge Acquisition for Knowledge-Based Systems, 7th European Knowledge Acquisition Workshop, EKAW' 93, pp320, September 1993.

[Niles et al 01] Niles, I., and Pease, A. 2001. Towards a Standard Upper Ontology. In Proceedings of the 2nd International Conference on formal Ontology in Information Systems (FOIS-2001), Chris Welty and Barry Smith, eds, Ogunquit, Maine, October 17-19, 2001.

[OntoMat] OntoMat (2002) http://annotation.semanticweb.org/ontomat.html

[OWL] http://www.w3.org/TR/2002/WD-owl-guide-20021104/

[OWL-S] http://www.daml.org/services/owl-s/1.0/

[PCKIMMO] http://www.sil.org/pckimmo/pc-kimmo.html

[Popov et al 03] Borislav Popov, Atanas Kiryakov, Angel Kirilov, Dimitar Manov, Damyan Ognyanoff, Miroslav Goranov KIM- Semantic Anotation Platform 2nd international Semantic Web Conference (ISW2003), 20-23 October 2003, Florida, USA. LNAI Vol.2870, pp.834-849, Springer-Verlang Berlin Heidelberg 2003.

[Protégé 2009] http://protege.stanford.edu/

[Pustejosky et al 03] James Pustejovsky, José M. Castaño, Robert Ingria, Roser Sauri, Robert J. Gaizauskas, Andrea Setser, Graham Katz, Dragomir R. Radev: TimeMl: Robust Specification of Event and Temporal Expressions in Text. New Directions in Question Answering 2003: 23-34.

[Pustejovsky 91] The generative Lexicon, Computational Linguistics 17(4), 409-441

[RDF(S)] http://www.w3.org/RDF/

[Sauguet 00] A. Sahuguet and F. Azavant. Building Intelligent Web Applications Using Lightweight Wrappers to appear in: Data and Knowledge Engineering, 2000.

[Saint-Dizier, Viegas 95] Saint-Dizier, P. e E. Viegas (1995), "An introduction to lexical semantics from a linguistic and a psycholinguistic perspective", in Saint-Dizier e E. Viegas (eds.), Computational Lexical Semantics, Cambridge: Cambridge University Press, pp.1-29.

[Schak 75] Schank, R. C., Conceptual Information Processing, North-Holland, Amsterdam, 1975.

[Schereiber et al, 01] A. Th. Schreiber, B. Dubbeldam, J. Wielemaker,and B.J. Wielinga. Ontology-based photo annotation. IEEE Intelligent Systems, 2001.

[Seaborne 01] A. Seaborne. RDQL: A data oriented query language for RDF models. http://www.hpl.hp.com7semweb/rdql.html, 2001.

[SESAME] http://sourceforge.net/projects/sesame/

[SOW 84] Sowa. J.F., Conceptual Structures, Information processing in mind and machine, Addison.

[Sowa 93] Sowa J.F. 1993, Representing attributes roles and nondetachable pazrts. International Workshop on Formal Ontology in Conceptual Analysis and Knowledge Representation, Padua, Italia, pags 69-73, Institute for Systems Theory Biomedical Engineering of the Italian National Research Council.

[Staab et al 01] Staab, S, Schnurr, H-P., Studer, R., Sure, Y.: Knowledge processes and ontologies, IEEE Intelligent Systems 16, 1 (2001).

[Studer et al 98] Studer, R., Benjamins, VR, Fensel, D., "Knowledge Engineering, Principles and Methods. In Data & Knowledge Engineering" 25(1-2):161-197, March, 1998.

[SUO] http://suo.ieee.org/

[Tecknowledge] http://www.daml.org/ontologies/321

[Unicode] www.unicode.org

[Uschold y Gruninger 96] Uschold, M. and Gruninger, M., Ontologies: Principles, Methods, and Applications Knowledge Engineering Review, 11(2): 93-155, 1996

[Van Harmelen 02] P. Patel-Schneider and I. Horrocks and F. van Harmelen: Reviewing the Design of DAML+OIL: An Ontology Language for the Semantic Web", Proceedings of the Eighteenth National Conference on Artificial Intelligence", R. Dechter and M. Kearns and R. Sutton Eds. 2002

[Vargas-Vera et al 02] Maria Vargas-Vera, Enrico Motta, John Domingue, Mattia Lanzoni, Arthur Stutt and Fabio Ciravegna "MnM: Ontology Driven Semi-Automatic and Automatic Support for Semantic Markup", The 13th International Conference on Knowledge Engineering and Management (EKAW 2002), ed Gomez-Perez, A., Springer Verlag, 2002.

[W3C] www.w3c.org

[WebODE] http:77delicias.dia.fi.upm.es/webODE/

[Weingand 97] Weigand, H. (1997). "Multilingual Ontology-Based Lexicon for News Filtering- The TREVI Project", en K. Mahesh (1997): 138-159

[Wielinga et al, 01] Bob Wielinga, Guus Screiber, J Wielemaker, and J. A. C. Sandberg. From thesaurus to ontology. Internation Conference on Knowledge Capture, Victoria, Canada, October 2001.

[Wittgestein 53] Wittgenstein, L. "Philosophical Investigations", Blackwell Publishing

[Woods 95] Woods B: "What' I in a link: Foundations for semantic networks". En D.G. Bobrow y A. M. Collins (Eds), Representation and Understanding: studies in Cognitive Science, pags: 38-85, Academic Press

[Wooldridge, Jennings 95] Wooldridge, Michael and Nicholas R. Jennings (1995). "Agent Theories, Architectures, and Languages: a Survey, "in Wooldridge and Jennings Eds., Intelligent Agents, Berlin: Springer Verlang, 1-22

[WSDL] http://www.w3.org/TR/wsdl

[XPDF] D. Noonburg. Xpdf: A CC++ library for accesing PDF. www.foolabs.com/xpdf.

[XLS] http://www.w3.org/Style/XSL/

[Yahoo!] www.yahoo.com

[Yamron 98] Yamron, J; Carp, I.; Gillick, L; Lowe, S; and van Mul-bregt, P. 1998. A hidden Mrkov model approach to text segmentation and event tracking. In Proceedings of the IEEE ICASSP.

Anexo I: Construcción de una Ontología de dominio

Este Anexo esboza las metodologías existentes y los posibles pasos a tomar en la construcción de una ontología.

Con el avance de las tecnologías de la Web Semántica existen varias iniciativas académicas proponiendo metodologías para la construcción de ontologías de dominio:

- ***Metodología*** propuesta por Uschold y Gruninger [Uschold y Gruninger 96] para la construcción de ontologías. Se considera seis etapas en el diseño de una ontología: Escenarios, preguntas de aptitud (Competency Questions) informales, lista de términos, preguntas de aptitudes formales, axiomas y teoremas de completitud.
- ***On-To-Knowledge:*** (Staab et al 01) Metodología basada de procesos de gestión de conocimiento que cubre tanto la parte de creación de una ontología como parte de su mantenimiento. Junto con la metodología están disponibles herramientas que ayudan algunos procesos.
- ***Methondology*** (Gomez-Perez et al 96): Eleva la construcción de ontologías a un proceso de ingeniería.

Las operaciones o tareas que se deben llevar a cabo para definir una ontología junto con una colección de instancias (a este conjunto se le puede llamar base de conocimiento) se pueden resumir en las siguientes:

1. Definir clases en la ontología
2. Definir una jerarquía de éstas (relación de generalización)
3. Definir las propiedades y restricciones sobre sus valores.
4. Creación de instancias
5. Asignar valores a las propiedades de las instancias

No existe una única ontología de un dominio determinado. Es decir, un mismo dominio puede ser modelado de diversas maneras, según lo que se considere relevante para la aplicación final. La calidad de una ontología se mide en función del grado de cumplimiento de los requisitos marcados por los usuarios.

Los mejores resultados los ofrecen los métodos iterativos. En estos métodos participan expertos en el dominio de ingenieros y diseñadores de la ontología. Estos métodos consisten en el establecimiento de una versión validada y revisada, en la que se han ido añadiendo detalles y consolidando decisiones tomadas, estudiando y comparando posibles alternativas y viendo el alcance de cada propuesta.

La ontología es un modelo de la realidad y por eso debe reflejarla. Además, se verá sujeta a cambios durante todo su ciclo de vida. Cada vez que se realicen modificaciones y/o actualizaciones será necesario realizar comprobaciones. Esto se suele hacer simulado los procesos de la aplicación final (ya sean búsquedas, inferencias, etc.).

En este paso se establecen las fronteras del modelo, es decir, hasta qué nivel de detalle es necesario modelar; que granularidad deben representar los conceptos; cómo de general deben ser los niveles altos; que conceptos del dominio son interesantes para la aplicación final, etc. Todas estas cuestiones deben de quedar resueltas antes de proseguir.

Las cuestiones principales con respecto al alcance de la ontología se enumeran a continuación:

- ¿Cuál es el dominio que cubrirá la ontología? Se debe acotar bien el dominio, para no modelar cosas poco relevantes en perjuicio de las más importantes. En este paso se acotará de manera preliminar el detalle de la ontología.
- ¿Para qué se utilizara la ontología? De las funcionalidades de la aplicación final dependerá el punto de vista bajo el cual el diseñador deberá modelar los conceptos de la realidad. Un mismo dominio se puede modelar con clases o atributos distintos según sea el objetivo final de la ontología.
- ¿Qué tipo de preguntas deberá satisfacer? Es una ayuda a las dos anteriores. Es una manera fácil para el usuario final de delimitar el dominio y establecer un punto de vista. Se le suele denominar "Preguntas de aptitudes (Competency Questions). Las respuestas a estas preguntas sugieren lo que podrían ser las instancias de la ontología, y de allí se pueden deducir (generalizando) las clases de la misma.
- ¿Quién mantendrá la ontología? Se debe saber de antemano si la persona encargada de mantener la ontología. Tiene nociones del dominio (si debe de poder crear nuevas clases, relaciones o modificar la jerarquía) o solamente se limitará a introducir instancias.
    - Mantenimiento de la ontología: Labor de gran alcance ya que introduce cambios en el modelado del dominio, redefiniendo conceptos o atributos. Un cambio en la definición del modelado puede acarrear la pérdida de instancias introducidas previamente.
    - Mantenimiento de las instancias: Cambios en el contenido de información.

Incorporación de estándares disponibles

Unos de los pasos más importantes es la documentación sobre estándares existentes. La reutilización de estándares permite asegurar el carácter consensuado de una ontología. No es muy frecuente construir una ontología desde cero, y se suelen reutilizar algunas partes de la ontología construida. La reutilización puede ser a varios niveles:

- ***Reutilización de conceptos generales*** (de alto nivel): Existen varias propuestas de modelos generales. Son algunas iniciativas muy detalladas y estudiadas de modelos generales para aplicaciones de procesamiento de lenguaje natural, clasificadores, etc., sin centrarse en ningún dominio en particular. Conceptos de niveles altos como agente, persona, evento, relación suelen utilizarse en ontologías de dominio particulares.
- ***Reutilización de modelos de dominio similares:*** En algunos dominios ya existen ontologías o esquemas construidos. Es una buena estrategia apoyarse en ellos.

- ***Reutilización de partes generales a niveles medios y bajos:*** Algunos conceptos de uso común de niveles bajos suelen estar modelados. Aquí se incluyen el tiempo (eventos, duración, orden temporal, etc.), espacio (lugares geográficos, sistemas de coordenadas, etc.). Existen repositorios de ontologías donde es conveniente consultar antes de emprender todo el modelo.

Enumerar los términos importantes de la ontología

Es una buena práctica enumerar junto con los expertos del dominio todos los términos que se consideran importantes en la aplicación. Algunos de estos términos se transformaran en clases de las ontologías, otros en atributos o instancias.

Definir las clases y la jerarquía

Una vez estudiado y acotado el dominio, se debe de proceder a la construcción de una jerarquía sobre la relación de especificación (o generalización) de los conceptos. Para ellos se usará como entrada las propuestas existentes en dominios similares y la lista de términos relevantes.

En ocasiones se planteará la disyunción entre crear una clase nueva en la jerarquía para modelar una especialización de un concepto o diferenciar estas especializaciones por el valor de un atributo. La decisión se basa en si en la jerarquía de especialización los diferentes conceptos tienen un conjunto diferente de atributos. Si es así, se justifica el uso de la jerarquía. En caso contrario, es suficiente con la diferenciación por el valor de un atributo.

Existen varios maneras de abordar este paso.

- ***Top-down***: Se comienza por conceptos generales que se van especificando hasta llegar a conceptos básicos del dominio.
- ***Bottom-up:*** Se enumeran los conceptos de más bajo nivel, y se intentan generalizar.
- ***Combinación:*** estrategia compuesta por las dos anteriores.

Algunas reglas en la construcción de la jerarquía:

- Sinónimos no son conceptos diferentes. Si dos palabras identifican el mismo concepto del domino, la ontología debería reflejar este hecho como una sola clase, que a lo sumo puede tener distintos nombres.
- Evitar ciclos. No deben de aparecer ciclos en las jerarquías de la relación de especificación.

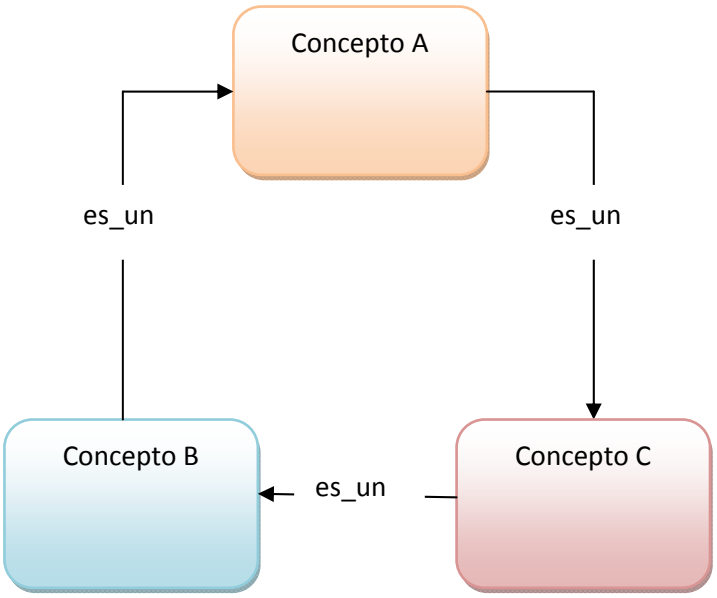


Figura 1 Ciclo propuesto en la relación de especialización

- Los hermanos deben ser equiparables. Los conceptos de un mismo nivel deben ser equiparables.

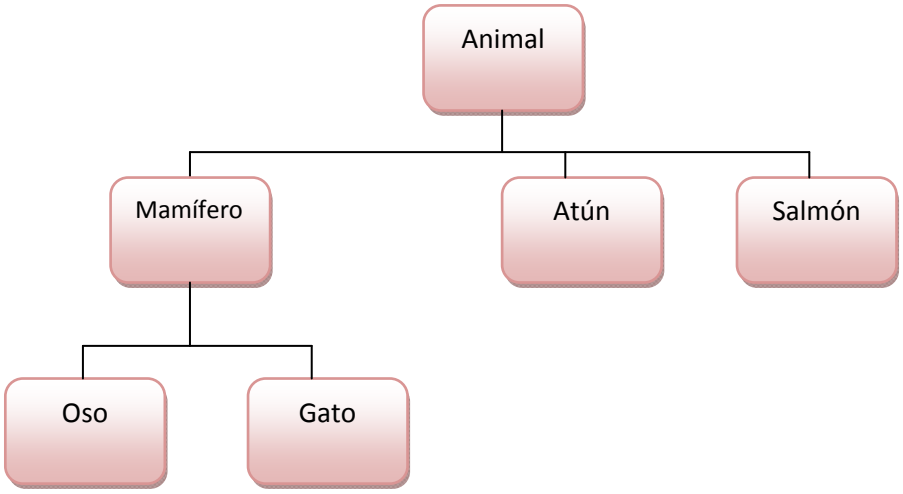


Figura 2 Ontología de animales con conceptos no equiparables

En la ontología debería haber un concepto PEZ entre el concepto ANIMAL y sus hijos ATUN y SALMON, ya que no es equiparable un MAMIFERO a un ATUN.

- Número de hijos: 2-12: El número de hijos debería ser mayor que uno, ya que un solo hijo cuestiona la existencia de su clase padre, y no debería sobrepasar la docena, ya que podría

haber implícita una clase intermedia. Es una regla orientativa para ontologías típicas. En algunos casos (construcción de diccionarios, listas de términos, etc., el número de hijos puede aumentar y quedar así una estructura plana de árbol).

Las reglas enumeradas anteriormente son una guía y no deben ser aplicadas de manera estricta, sino en función del dominio y del objetivo de la aplicación.

Definir propiedades de clases

Una vez definida la primera versión de los conceptos y de la jerarquía que forman se procede a definir las propiedades de cada uno de estos conceptos. Las propiedades cuyo valor son instancias de otros conceptos constituirán relaciones dentro del modelo. Si no fuera suficiente con habilitar una propiedad, existe la posibilidad de construir un concepto que modela dicha relación, pudiendo establecer las propiedades de la misma.

Propiedades:

- Intrínsecas: propias del concepto, que lo caracterizan por su naturaleza.
- Extrínsecas: propiedades añadidas al concepto, como el nombre, etc.
- Partes del concepto: Propiedades estándar en algunos formalismos Define las partes del concepto.
- Relaciones con otros conceptos: propiedades cuyos valores son instancias.

Se debe tener cuidado con la herencia, ya que los atributos del concepto padre pasaran automáticamente al concepto subordinado. Normalmente las herramientas de construcción de ontologías realizan esta tarea de manera automática.

Definir las restricciones de las propiedades

También llamadas facetas. Son propiedades de las propiedades y las más usadas son:

- Cardinalidad: número de valores que puede tomar la propiedad.
- Tipo de valor: Cadena, Número, Lógico Enumerado o Instancia de otro concepto.
- Dominio y Rango: Se restringe el valor a unos pocos.

Creación de instancias

De las fuentes de dominio, los expertos y fuentes externas se identifican instancias de la ontología y se introducen rellenando los valores de los atributos de los conceptos.

Es una decisión sobre el grado de granularidad de la representación, determinada por la aplicación. Las respuestas a las preguntas de aptitudes (Competency Questions) son individuos (instancias).

En la documentación de la ontología debe de quedar claro cuando se adopta cada una de las alternativas, y debe estar consensuado cualquier cambio en la metodología.

Pregunta de aptitudes

Es un método que permite que el experto en el dominio identifique los conceptos más importantes (sobre todo aquellos de bajo nivel, cercanos al dominio). Las preguntas de aptitudes son preguntas que el experto desea que la ontología responda. Para ello debe de contener información para elaborar la pregunta en forma de ***query*** (pregunta estructurada) así como disponer de conceptos que constituyan una respuesta a estas.

Una de las reglas que permite ayudar en la decisión de cuando añadir instancias o conceptos en la jerarquía en que las respuestas a las preguntas deben de ser instancias.

Anexo II: Diseño Software

Uno de los requisitos más importantes en la fase de análisis fue la extensibilidad de la arquitectura para ser aplicada en distintas tecnologías y distintos dominios. Desde el punto de vista de análisis para arquitectura conceptual se hecho un esfuerzo por modelar las distintas partes de los sistemas de extracción de información y relleno de ontologías. El proceso total se ha separado en tres fases diferenciadas y se han establecido mecanismos de incorporación de nuevas tecnologías y así como mecanismos de descripción y extensión de estrategias de control.

En este capítulo se presenta una propuesta de diseño de software que apoya el cumplimiento de los requisitos identificados en fases previas. El diseño presentado se ha utilizado con éxito en la implementación de un sistema de extracción y relleno de ontologías según el dominio que se maneje. En el diseño se ha utilizado el paradigma de la programación orientada a objetos y las decisiones tomadas en el modelado han obedecido los siguientes criterios:

- ***Posible reutilización de librerías ya existentes de terceros:*** la existencia de algunas librerías de procesamiento de documentos online y navegación por fuentes hipertextuales ha determinado el diseño final de la aplicación.
- ***Posible reutilización de librerías construidas como parte del sistema:*** Partes de las funcionalidades se han diseñado para poder reutilizarlas con facilidad en aplicaciones distintas a la implementada aquí. Se trata de los módulos de procesamiento de fuentes HTML (paquete parser), módulos de procesamiento de lenguaje natural paquete NLP) y módulos de gestión de ontologías que permiten el manejo de distintos estándares y sistemas de almacenamiento de ontologías (paquete Ontology).
- ***Extensibilidad del sistema*** *para la incorporación de nuevas posibles interpretaciones de fuentes (de acuerdo a diferentes tecnologías existentes) y de nuevas estrategias de control de la extracción e inserción.* Para ello se ha optado por el uso de patrones de diseño en la programación orientada a objetos, llamados patrones de factoría que permiten la incorporación de nuevas funcionalidades sin la necesidad de recompilación del programa fuente.

Proposición de patrón de diseño: Factoría

El patrón de diseño de la factoría proporciona una interfaz para crear familias de objetos sin la especificación de ninguna clase de forma concreta. De esta manera la comprobación de la existencia de las clases para la creación de instancias se traslada al tiempo ejecución permitiendo así la incorporación de nuevas librerías sin necesidad de reconstrucción del código objeto de la aplicación. Cada parte del diseño donde sea deseable permitir una fácil extensión de funcionalidades cuenta con una clase de factoría que crea instancias necesarias para el procesamiento. El tipo de instancia que se ha de crear se especifica con ficheros de configuración y la existencia del código necesario para su creación se hace en tiempo y ejecución en base a la ontología de documentos técnicos que se eligió.

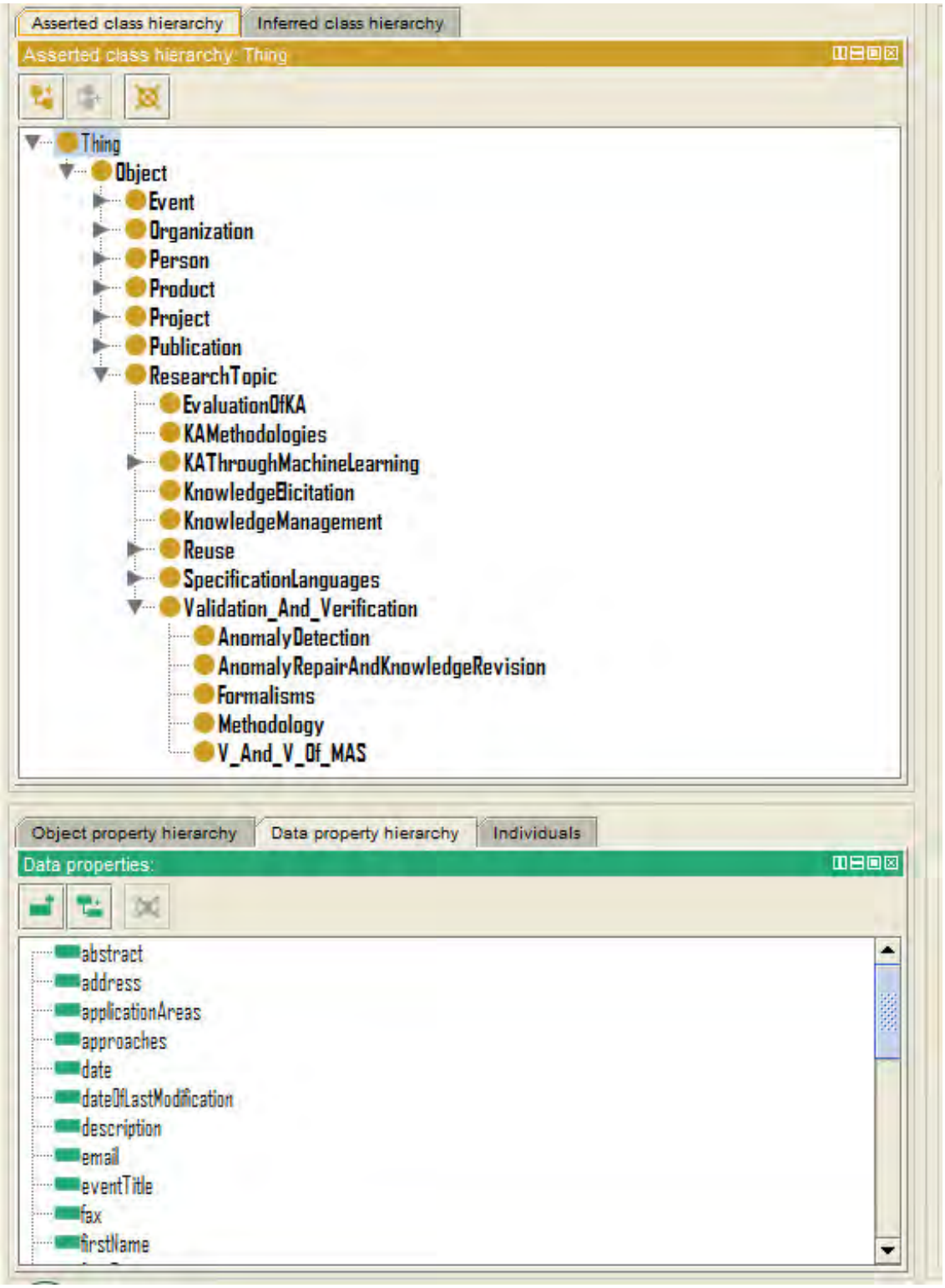


Figura 1 Ontología de documentos técnicos

Por ejemplo, para la creación de posibles interpretaciones de un documento fuente existe una factoría de documento. En el sistema propuesta se permite la creación de cuatro posibles

interpretaciones (Aspecto, Lenguaje, DOM o Texto Plano) de un documento, según el operador que se decida aplicar desde el modulo de estrategia. Si se añadiese un nuevo operador cuyo requisito fuese la disponibilidad de una nueva interpretación de un documento (por ejemplo: información estadística o interpretación de contenido multimedia), sería suficiente con incluir su presencia en el fichero de configuración.

Diagramas UML

A continuación se presenta un esbozo del diseño UML de algunas partes del sistema implementando a nivel paquetes. A un nivel más alto el diseño divide la aplicación en tres paquetes principales:

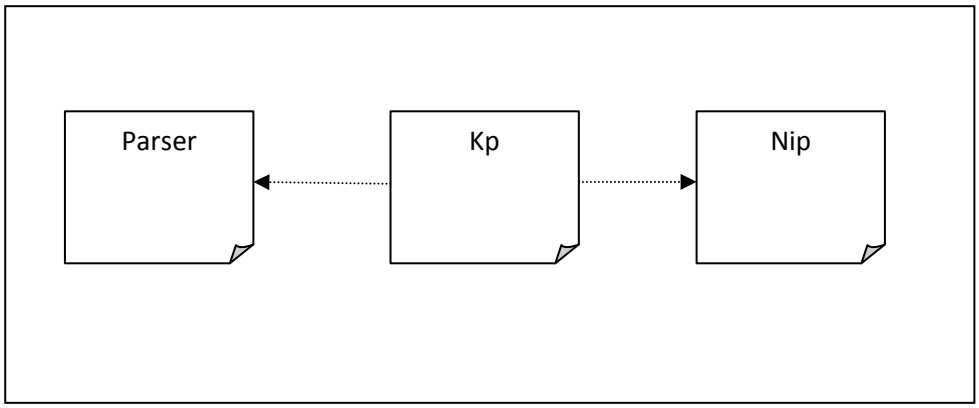


Figura 2 Paquetes UML de alto nivel

- ***Kp*** Paquete central que implementa el sistema (KP es abreviatura del inglés: Knowledge Parser)
- ***Nip*** Paquete que agrupa las clases con funcionalidades de procesamiento de lenguaje natural.
- ***Parser*** Paquete que agrupa las clases con funcionalidades de procesamiento de fuentes HTML.

Esta separación de los paquetes de procesamiento de lenguaje natural y procesamiento de fuentes HTML se ha hecho por razones de reutilización del software. A un nivel más bajo, ya dentro del paquete ***Kp*** se dividen las clases en tres categorías:

- Funcionalidad principal: Paquetes correspondientes a las partes funcionales de la arquitectura:
    - *Document:* Paquete (incluye un patrón de factoría) para la creación de distintas interpretaciones de los documentos fuente. Se corresponde con la fase de preproceso de los documentos.

- *Decision:* Paquete (incluye un patrón de factoría) para la toma de decisiones en el proceso de ejecución de estrategias que tiene por objetivo la construcción de hipótesis. Forma parte de la fase segunda, de extracción de información.
- *Operator:* Paquete (incluye un patrón de factoría) que engloba los distintos operadores que se pueden ejecutar desde las estrategias. También forma parte de la segunda fase, de extracción.
- *Ipo:* Abreviatura del inglés de intelligent Population of Ontology, es el paquete encargado de la inserción de la información en la ontología de dominio a partir de las hipótesis creadas por la segunda fase. Pertenece a la tercera y última fase del proceso.

- *Funcionalidades de apoyo:* Funcionalidades de nivel horizontal respecto al diagrama de las distintas fases del proceso, que dan apoyo a tanto preproceso, como extracción o relleno de información. Se trata de estos dos paquetes:
  - *Ontology:* Paquete de manejo y gestión de ontologías, tanto de dominio como de adquisición.
  - *Data:* Paquete que engloba en intercambio de datos entre las distintas fases del proceso.
- *Utilidades:* Funcionalidades de utilidad básicas.
  - *Util:* Manejo de cadenas de caracteres, estructuras de datos, etc.
  - *Log:* sistema de traza.
  - *Config*: Sistema de gestión de parámetros de configuración.

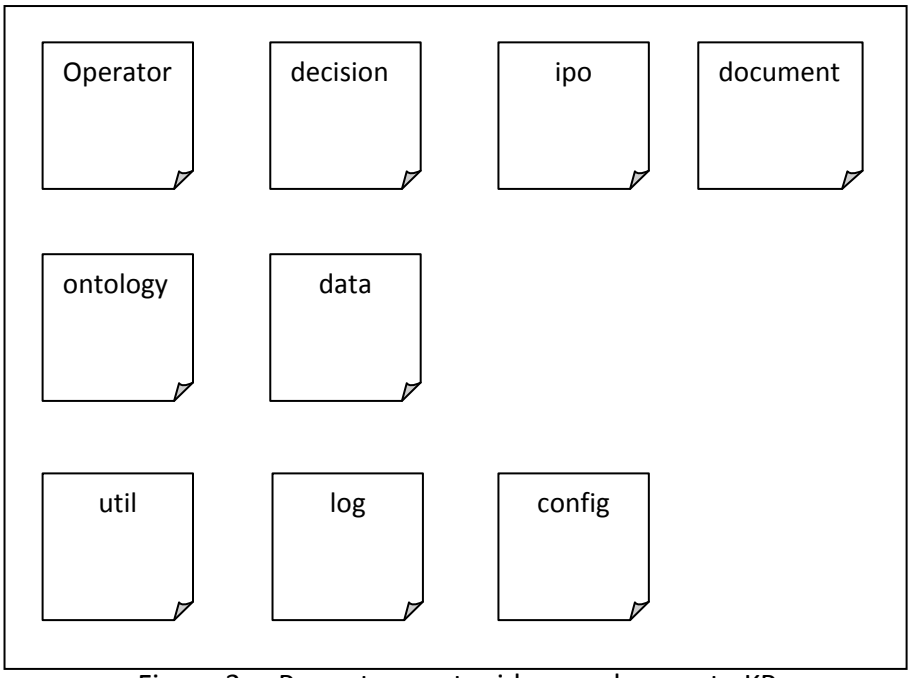


Figura 3 Paquetes contenidos en el paquete KP

Ontology

Paquete de gestión de ontologías implementa una interfaz de acceso a modelos semánticos usados tres posibles librerías: JENA (JENA) en su versión 2.6.2 y WebODE 2.0.9 (WebODE) y SESAME 2.3.1 (SESAME).

Las ontologías se dividen en dos partes:

- Esquema que indica las clases, los atributos y las relaciones existentes.
- Datos que contiene las instancias de las clases, relaciones y los valores de los atributos.

La librería implementada ofrece un interfaz común al tratamiento de las ontologías representadas en RDF. Para ello se apoya de tres librerías externas que permiten el manejo de las ontologías a bajo nivel así como almacenarlas en fichero o base de datos, además de apoyarse del siguiente código para la interacción entre ontologías.

```
/*************************************************************************
*****
 * Source code information

*************************************************************************
****/

// Package
///////////////
package jena.ontology.persistentOntology;


// Imports
///////////////
import java.util.*;

import com.hp.hpl.jena.db.*;
import com.hp.hpl.jena.ontology.*;
import com.hp.hpl.jena.rdf.model.*;


/**
 * <p>
 * Application of using the persistent db layer with ontology models.
Assumes
 * that a MySQL database called 'jenatest' has been set up, for a user
named ijd.
 * </p>
 *
 * @author Gerson Villa, IPN
 *         (<a  href="mailto:gvilla@ipn.mx" >email</a>)
 * @version CVS $Id: PersistentOntology.java.html,v 1.4 2010/01/17
10:44:23 gerson_villa Exp $
```

```
 */
public class PersistentOntology {
    // Constants
    //////////////////////////////////


    // Static variables
    //////////////////////////////////

    // Instance variables
    //////////////////////////////////

    // Constructors
    //////////////////////////////////

    // External signature methods
    //////////////////////////////////

    public void loadDB( ModelMaker maker, String source ) {
        // use the model maker to get the base model as a persistent
model
        // strict=false, so we get an existing model by that name if it
exists
        // or create a new one
        Model base = maker.createModel( source, false );

        // now we plug that base model into an ontology model that also
uses
        // the given model maker to create storage for imported models
        OntModel m = ModelFactory.createOntologyModel( getModelSpec(
maker ), base );

        // now load the source document, which will also load any imports
        m.read( source );
    }

    public void listClasses( ModelMaker maker, String modelID ) {
        // use the model maker to get the base model as a persistent
model
        // strict=false, so we get an existing model by that name if it
exists
        // or create a new one
        Model base = maker.createModel( modelID, false );

        // create an ontology model using the persistent model as base
        OntModel m = ModelFactory.createOntologyModel( getModelSpec(
maker ), base );

        for (Iterator i = m.listClasses(); i.hasNext(); ) {
            OntClass c = (OntClass) i.next();
            System.out.println( "Class " + c.getURI() );
        }
    }


    public ModelMaker getRDBMaker( String dbURL, String dbUser, String
dbPw, String dbType, boolean cleanDB ) {
```

```
        try {
            // Create database connection
            IDBConnection conn  = new DBConnection( dbURL, dbUser, dbPw,
dbType );

            // do we need to clean the database?
            if (cleanDB) {
                conn.cleanDB();
            }

            // Create a model maker object
            return ModelFactory.createModelRDBMaker( conn );
        }
        catch (Exception e) {
            e.printStackTrace();
            System.exit( 1 );
        }

        return null;
    }

    public OntModelSpec getModelSpec( ModelMaker maker ) {
        // create a spec for the new ont model that will use no inference
over models
        // made by the given maker (which is where we get the persistent
models from)
        OntModelSpec spec = new OntModelSpec( OntModelSpec.OWL_MEM );
        spec.setImportModelMaker( maker );

        return spec;
    }
```