**UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA**

**DEPARTAMENTO DE INFORMÁTICA Y SISTEMAS**

**TESIS DOCTORAL**

# APROXIMACIÓN A UNA ESTACIÓN LEXICOLÓGICA ORIENTADA A INTERNET

**ZENÓN J. HERNÁNDEZ FIGUEROA**

Las Palmas de Gran Canaria, Abril del 2002

64/2001-02

**UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA**
**UNIDAD DE TERCER CICLO Y POSTGRADO**

**Reunido el día de la fecha, el Tribunal nombrado por el Excmo. Sr. Rector Magfco. de esta Universidad, el/a aspirante expuso esta TESIS DOCTORAL.**

**Terminada la lectura y contestadas por el/a Doctorando/a las objeciones formuladas por los señores miembros del Tribunal, éste calificó dicho trabajo con la nota de** SOBRESALIENTE CUM LAUDE (POR UNANIMIDAD)

**Las Palmas de Gran Canaria, a 27 de mayo de 2002.**

**El/a Presidente/a: Dr.D. Manuel Alvar Ezquerra,**

**El/a Secretario/a: Dr.D. José Rafael Pérez Aguiar,**

**El/a Vocal: Dr.D. Antonio Núñez Ordóñez,**

**El/a Vocal: Dr.D. Francisco Sanchís Marco,**

**El/a Vocal: Dra.Dña. Margarita Díaz Roca,**

**El Doctorando: D. Zenon José Hernández Figueroa,**

# UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA

DEPARTAMENTO DE INFORMÁTICA Y SISTEMAS





# TESIS DOCTORAL

# Aproximación a una estación lexicológica orientada a Internet


**Autor: D. Zenón J. Hernández Figueroa**
**Director: Dr. D. Octavio Santana Suárez**

**Abril 2002**

**UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA**

DOCTORADO EN INFORMÁTICA

DEPARTAMENTO DE INFORMÁTICA Y SISTEMAS

Programa Informática Documental y Lingüística Computacional

# Aproximación a una estación lexicológica orientada a Internet

*Tesis doctoral presentada por:D. Zenón J. Hernández Figueroa*
*Dirigida por el Dr. D. Octavio Santana Suárez*

*El director* *El doctorando*

*Las Palmas de Gran Canaria . Abril 2002*

# RESUMEN

Esta tesis es una proyección natural de los trabajos realizados por el Grupo de Estructuras de Datos y Lingüística Computacional de la ULPGC en los últimos años. Estos trabajos se han desarrollado en el ámbito de la Lingüística Computacional y han dado lugar, entre otros resultados, al desarrollo de herramientas de reconocimiento y generación morfológica. En esta tesis se propone la utilización de dichas herramientas como parte de nuevas aplicaciones cuyo objetivo es obtener provecho del enorme caudal de información lingüística que supone Internet.

Se caracterizan dos clases de aplicaciones —en función del grado de interactividad de los estudios lingüísticos que se pretenda realizar— y se desarrollan sendos prototipos —denominados DAWeb y NAWeb— con una arquitectura estudiada para obtener los rendimientos más adecuados a cada caso. NAWeb está diseñado para la exploración en detalle de documentos individuales bajo supervisión directa del usuario y reúne las características típicas de un navegador web pero además incorpora numerosas opciones para el análisis de las páginas accedidas. DAWeb se orienta al estudio conjunto de grandes volúmenes de documentos de forma desasistida y adopta el formato de un descargador de páginas con la diferencia de que en vez de bajar las páginas que accede, las analiza y almacena sólo los resultados.

Las modalidades de análisis abarcan: la detección de neologismos, estudio del uso de las palabras con diversas medidas cuantitativas y cualitativas, y aspectos cercanos a la sintaxis tales como colocaciones léxicas o regímenes preposicionales.

Las dos aplicaciones comparten los módulos relacionados con la obtención de los textos y su posterior análisis —esto incluye un módulo optimizador de búsqueda morfológica que aumenta sustancialmente la velocidad de reconocimiento. DAWeb se caracteriza por tener una arquitectura orientada al procesamiento en paralelo para minimizar los costos de acceso a Internet, mientras que el aspecto más destacado de NAWeb es su marcada interactividad.

Ambas aplicaciones aportan un novedoso complemento al concepto de Estación Lexicológica que algunos autores —especialmente en el campo de la lexicografía— han postulado con anterioridad, centrándose sobre todo en la gestión de la información disponible —mediante sistemas de bases de datos— y la generación de productos a partir de la misma —diccionarios. En esta tesis el foco se pone en la fase de obtención de información lingüística a partir de una fuente como la metarred, no disponible en el pasado, pero con una fuerte proyección de futuro. Además, pueden —especialmente NAWeb— tener otras utilidades como el estudio de estilos o la enseñanza del español a extranjeros; con este fin se ha dotado a NAWeb de la capacidad de analizar otros formatos —texto plano, documento de Microsoft Word— distintos del típico HTML de las páginas web.

# AGRADECIMIENTOS

Quiero expresar mi agradecimiento en primer lugar al Dr. D. Octavio Santana Suárez que como tutor y director de esta tesis no ha dejado de aplicar el empuje adecuado para que la misma llegase a ser una realidad y cuya motivación ha resultado saludablemente contagiosa.

También quiero mencionar especialmente a D. Gustavo Rodríguez Rodríguez, mi compañero de despacho, cuya ayuda resultó esencial para allanar las dificultades que se fueron presentando en el camino.

El resto de los miembros del Grupo de Estructuras de Datos siempre han estado disponibles cuando me ha hecho falta su apoyo, bien en temas relacionados con la tesis, bien en temas no relacionados que se han podido resolver gracias a ellos sin que fuese necesario restarle tiempo al objetivo principal. Por ello merecen también mi gratitud.

# Índice.

# 1.- Objetivos y antecedentes.

El objetivo general de la presente tesis consiste en obtener una estación lexicológica orientada a Internet que integre un conjunto de aplicaciones informáticas especializadas en tareas de análisis de textos en documentos electrónicos disponibles en la metarred. En este propósito confluyen, de una parte, Internet como ente generador y suministrador de recursos lingüísticos, de otra, la investigación lingüística como cliente y beneficiaria del nuevo nicho de oportunidades de estudio abierto por la red y, de una tercera, las técnicas informáticas de gestión y presentación de información compleja como vehículo de intermediación y catalización entre las dos anteriores.



## 1.1.- Lenguaje e informática.

La relación de la informática con el lenguaje no es nueva. El que se considera el primer ordenador electrónico de propósito general —ENIAC— data de 1 946 y la que parece ser la aplicación más antigua reconocible en el campo del procesamiento del lenguaje natural —un sistema de búsqueda en diccionario desarrollado en el Birkbeck College de Londres— data de 1 948. Con la traducción automática arranca el interés por el tema en EEUU en 1 949; en esa época se creía

posible resolver la traducción del lenguaje natural por extensión de los problemas de criptografía —bastante desarrollados a raíz de la segunda guerra mundial. El interés americano se extendió rápidamente a Francia, Inglaterra y la URSS; estuvo primero centrado en la traducción del alemán —por la ingente cantidad de documentos científicos capturados al terminar la guerra— y posteriormente en el ruso —consecuencia de la guerra fría. Sin embargo, los enormes esfuerzos en esta época resultaron improductivos por mor de la ingenuidad del planteamiento inicial, ya que los lenguajes naturales son extraordinariamente más complejos que cualquier código criptográfico. Tal reconocimiento desplaza el foco de atención hacia la investigación lingüística como disciplina capaz de desbrozar esa complejidad y proporcionar el conocimiento adecuado para conseguir las metas propuestas.

La interacción entre la investigación lingüística y la informática puede considerarse mutua. De un lado, el trabajo del lingüista es indispensable para acercar los grandes objetivos del procesamiento del lenguaje natural y que el ordenador "entienda" y se exprese al menos en un subconjunto amplio de dicho lenguaje; por otro, las herramientas desarrolladas por la informática pueden ayudar de forma importante al lingüista en la realización de su labor —la colaboración entre ambas disciplinas redunda en que los útiles mejorarán y se adaptarán más a las necesidades reales del investigador. El abanico de herramientas que la informática ofrece abarca

diversos grados de complejidad y especificidad: desde el procesador de textos hasta las estaciones de trabajo lexicológicas.

Aunque no da mucho juego como herramienta de investigación lingüística, un simple procesador de textos ya es una herramienta útil, al menos en la elaboración de documentos y resultados. Los sistemas de gestión de bases de datos (SGBD) constituyen una herramienta informática consolidada y pueden jugar un papel bastante relevante en la organización de información lingüística. En [MALD98 ] se lee: "...yo quería trabajar con fichas informáticas equivalentes a las fichas de toda la vida, ordenadas alfabéticamente en aquellas cajas verdes de siempre... Es decir, yo estaba demandando una base de datos" , y en [ALVA98] se afirma: "El paso más adelantado en la redacción de diccionarios asistida por ordenador lo constituyen las bases de datos". De especial relevancia en muchos aspectos de la investigación lingüística son los sistemas de hipertexto; se hace referencia a ellos en [MORR99], donde se analizan las dos formas en que la informática se incorpora al campo de la crítica textual: aparece como consecuencia natural de las corrientes que, en busca de la máxima objetividad y rigor en el proceso de depuración de la obra, preconizaban el acercamiento de la literatura al campo de las ciencias exactas —en tal concepción de la crítica textual, existe un importante aspecto mecánico en el que la informática puede jugar un papel primordial como herramienta auxiliar en la fijación y elaboración de ediciones.

En consecuencia, se dispone de programas útiles para la colación y filiación de textos, tanto desarrollados específicamente como tomados de otras áreas de aplicación, así como de otros que ayudan a preparar el texto para la imprenta; no obstante, ninguna de las metodologías desarrolladas consigue resultados congruentes de forma automática sin intervención final del especialista. No parece la única —quizás ni siquiera la principal— manera de aplicar la informática al campo de la crítica textual ni ahora ni en un futuro. Otras corrientes descartan por inviable la búsqueda de la versión "ideal" y se decantan por la publicación de cada uno de los documentos que forman la historia de una obra literaria —lo que resulta complicado en formato impreso tradicional—; el hipertexto parece la herramienta informática adecuada para proporcionar una visión integradora de la obra, aunque puede resultar inútil si se limita a un simple "amontonamiento" de versiones. La aportación de la informática debe conducir a la superación del enfrentamiento entre ambas posturas: reunir en formato electrónico el texto ideal y el proceso que permite llegar hasta él, de modo que el lector pueda elaborar su propio juicio.

Cuando el planteamiento evoluciona hasta integrar un conjunto de herramientas orientadas a la investigación lingüística y agrupadas en un entorno de trabajo específico, aparecen las llamadas "estaciones de trabajo" —término tomado del inglés 'workstation' que se emplea para designar un sistema de 'hardware' y 'software' integrados, relativamente potente y que funciona como herramienta

centralizadora del trabajo personal de un investigador o desarrollador. En [MILL99] se describe una estación de trabajo filológica como "un entorno informático diseñado para manejar textos aislados o en conjunto y que contiene: 1) los datos, 2) las herramientas para su utilización y 3) la plataforma de desarrollo de la obra resultante —todo ello de un modo integrado y con una interfaz adecuada". En esta definición encaja la estación de trabajo lexicográfica propuesta en [ZAMP91], con la que el lexicógrafo interactúa con los córpora para buscar formas específicas de palabras, formas de palabras que casen con una cadena específica, coocurrencias de formas de palabras y/o cadenas en un espacio determinado del texto, etc. En general, el lingüista hará muchas otras cosas: localizar concordancias, realizar estudios estadísticos sobre el uso de las palabras, de sus componentes, de segmentos de texto formados por varias palabras —casuales, frases hechas, perífrasis, ...

Desde el punto de vista de la correlación informática-lingüística, los aspectos referidos a la plataforma de desarrollo y la interfaz integrada se consideran eminentemente tecnológicos; la gestión de los datos y las herramientas encaminadas a su estudio y proceso resultan más relevantes: deben permitir al lingüista realizar toda clase de estudios, búsquedas, análisis y clasificaciones con los datos, según sus necesidades. Debido al carácter mayoritariamente textual y múltiple de los datos conviene usar técnicas de hipertexto para su organización.

Uno de los problemas con los que solía encontrarse la investigación lingüística cuando pretendía usar la informática como instrumento era la escasez de textos disponibles en formato electrónico de forma apropiada. Ante ello cabía: 1) "apañarse" con lo que hubiera, con lo que se acababa estudiando una colección de documentos reunidos al vuelo, de dudosa proyección sobre la realidad del lenguaje y prácticamente nula representatividad de lo estudiado; 2) que cada cual intentara pasar a formato electrónico los textos que no lo estuvieran, bien escaneándolos —proceso tedioso que generalmente requiere una cuidadosa revisión posterior— o bien tecleándolos directamente —lo que no es menos trabajoso. Afortunadamente, hoy día parece que se evoluciona a una situación en la que casi todo acabará estando disponible de una u otra forma en formato electrónico —el fenómeno conocido como Internet juega un papel nada despreciable.



## 1.2.- Internet.

Internet es el resultado de un programa de investigación puesto en marcha en 1 973 por la Defense Advanced Research Projects Agency (DARPA) de Estados Unidos con el objetivo de desarrollar protocolos de comunicaciones que permitiesen a los ordenadores comunicarse de forma transparente a través de múltiples redes.

Internet puede definirse como una metarred o "red de redes" —técnicamente

es una definición correcta—; se fundó sobre la idea básica de aglutinar múltiples redes independientes. Una resolución del "Federal Networking Council" de 24 de octubre de 1995 establece : "El término 'internet' designa el sistema de información global que: i) está lógicamente interconectado por un espacio de direccionamiento único basado en el Internet Protocol (IP) o sus extensiones; ii) es capaz de proveer comunicaciones usando el Transmission Control Protocol/Internet Protocol (TCP/IP) o sus extensiones y otros protocolos compatibles con IP; y iii) suministra, usa o hace accesible, de forma pública o privada, servicios de alto nivel basados en las infraestructuras de comunicaciones y relacionadas aquí descritas".

Sin embargo, la cuestión es que Internet va más allá de la dimensión tecnológica, no en vano a lo largo de la última década ha acabado convirtiéndose en un fenómeno social; ha revolucionado la informática y las comunicaciones a nivel mundial de una forma que no tiene precedentes, a lo que han contribuido de manera fundamental algunos "servicios de alto nivel" capaces de facilitar la interacción de los usuarios.

A finales de la década de los 80 Internet ya servía a miles de usuarios; sin embargo, los modos de interacción eran muy poco amigables —complejas interfaces de línea de comando—, lo que dificultaba su aproximación al gran público no especializado que empezaba a estar familiarizado con las interfaces gráficas de los

ordenadores personales —mucho más intuitivas. El salto cualitativo que puso en marcha la acelerada expansión del uso de Internet fue el World-Wide Web (WWW).

El Worl-Wide Web nació en el CERN (Centro Europeo de Investigación Nuclear) como proyecto para crear un sistema hipermedia distribuido. La necesidad que tenía el CERN de un sistema de este tipo residía en su dependencia de unos equipos de investigación muy caros que obligaba necesariamente a la colaboración e intercambio de datos entre los investigadores. Aunque el WWW no fue el primer sistema hipermedia, contó con características específicas que facilitaron su enorme éxito: destaca que su plataforma de implantación es toda Internet y no un sistema particular. En 1 993 se desarrolló en el NCSA (National Center for Supercomputing Applications) un nuevo navegador con interfaz gráfica —Mosaic— que, aparte de otras innovaciones, apostó por la difusión al distribuirse gratis por la propia red; Mosaic inauguró así una costumbre que luego se ha mantenido con otros navegadores —sólo en el primer mes se distribuyeron 40 000 copias. En cualquier caso, de nada serviría disponer de un navegador gratuito si no hay contenidos en los que navegar, y ahí se encuentra otro de los puntos fuertes del WWW: la facilidad para convertirse en servidor de contenidos por parte de cualquiera que tenga interés en hacerlo.

La proliferación de sitios web constituye una fuente extraordinariamente rica para el estudio documental y lingüístico, no sólo por lo que se refiere a nuevos usos inducidos por la red, sino también por la mayor facilidad de acceso a medios y documentos tradicionales que dan lugar a verdaderas bibliotecas virtuales —en constante expansión— en las que se pueden encontrar versiones cuidadosamente editadas —y otras, no tanto— de obras clásicas y contemporáneas.

Hoy en día todos los diarios y publicaciones periódicas relevantes ofrecen ediciones electrónicas, lo que supone una disponibilidad de documentos superior a cualquier otra época histórica. El investigador contemporáneo dispone, con mayor facilidad que si se encontraran en el quiosco de la esquina, de acceso casi inmediato a cualquier publicación de cualquier parte del mundo; basta una sencilla búsqueda para encontrar cientos de diarios en español, en un ámbito geográfico que va de España a Australia y de Texas a Patagonia.

No sólo las correspondientes ediciones electrónicas de publicaciones tradicionales, sino cualquier documento de Internet puede ser objeto de interés para la investigación lingüística y documental —sin mayor dificultad, por su mejor factura. En la medida en que las publicaciones tradicionales que se suman a la red lo hacen según los criterios y metodologías más comunes y según las tecnologías mejor consolidadas en Internet se facilita la construcción de herramientas de ayuda

al análisis —los problemas surgen cuando desde el medio tradicional se insiste en trasladar a la red formatos y métodos que no le son propios.

Es previsible y deseable que el volumen de documentos disponibles continúe en ascenso; en el Anuario 2000 de la serie "El español en el mundo" [WEB04] que elabora el Instituto Cervantes se lee: "La presencia firme del español en internet exige asegurar para el español una banda de uso en la red de redes de entre un 15 y un 25 por ciento en los próximos cuatro años, referida a los accesos que son en inglés. (La situación, según datos de julio de 1 999, es de un 10,1 por ciento —el chino, el alemán y el japonés tienen magnitudes superiores. Si se incluye el inglés, el español se usa en un 4,3 por ciento de las conexiones, el inglés en un 57,4 por ciento, y ninguna otra lengua alcanza un 10 por ciento del uso total). El aumento de la presencia sólo puede lograrse mediante un incremento de los contenidos en español y un desarrollo comparable de los sistemas de recuperación inteligente de la información y su análisis.". Un estudio más reciente presentado en [DAIL01] cifra en un 5,4% el porcentaje de hispanohablantes sobre un total estimado de 476 millones de internautas y predice que se modificará rápidamente a medida que Sudamérica y Asia completen su infraestructura de Internet. A pesar de que una presencia de entre el 4,3 y el 5,4 por ciento puede ser menos de lo que corresponde a una lengua como el español, en valores absolutos representa una cantidad enorme de información; el objetivo propuesto para los próximos cuatro años es muchísimo

mayor, sobre todo teniendo en cuenta que, dado que internet crece globalmente a buen ritmo, un 15 por ciento dentro de cuatro años representará en valores absolutos mucho más que el mismo porcentaje hoy día.

Paradójicamente, el extraordinario volumen de información disponible constituye el principal problema del nuevo medio y también uno de sus principales puntos fuertes. Antes, el investigador tenía acceso a unos pocos documentos, ahora puede obtener en unos minutos muchos más de los que daría cuenta en el pasado durante toda una vida de trabajo —la dificultad está en cómo conseguir extraer de ese inmenso volumen de información un producto útil. Los documentos se generan a un ritmo muy superior al que cualquier ser humano puede emplear en estudiarlos. En consecuencia, por poco que se pierdan las referencias es posible encontrarse extraviado —mucha más información, con menos conocimiento. Esta situación justifica plenamente cualquier esfuerzo que se haga en la dirección de desarrollar todo tipo de herramientas destinadas a desbrozar, cual machete de explorador, la jungla de información que es Internet; este empeño facilitará que el investigador pueda, con un esfuerzo no necesariamente superior al que debía emplear en el pasado, alcanzar unos objetivos bastante más ambiciosos gracias a los recursos disponibles —la riqueza de información asequible frente a los medios tradicionales es comparable a la exuberancia de la selva tropical frente a la de la estepa castellana, sin desmerecer la belleza de esta última. Los objetivos propuestos en esta tesis son

un esfuerzo más en la línea marcada por la frase final del párrafo del citado Anuario 2 000 del "Español en el mundo": "...y un desarrollo comparable de los sistemas de recuperación inteligente de la información y su análisis.".

## 1.3.- Trabajos previos del Grupo de Estructura de Datos.

El Grupo de Estructuras de Datos y Lingüística Computacional del Departamento de Informática y Sistemas de la Universidad de Las Palmas de Gran Canaria empezó a desarrollar herramientas de Lingüística Computacional a partir de 1 990 como proyección natural y aplicación práctica de sus desarrollos previos en estructuras de datos e informática documental. Los trabajos se han centrado fundamentalmente en el campo de la morfología computacional aplicada y en determinadas aplicaciones en el área de la lexicografía computacional —principalmente en relación con diccionarios de sinonimia e ideológicos—; también mantiene líneas de desarrollo en el campo de la sintaxis y la semántica, iniciadas con posterioridad. Como resultado de sus investigaciones, el grupo ha generado diversas aplicaciones y publicaciones, de las que, desde el punto de vista de lo que resulta relevante para el presente trabajo, cabe destacar: FrecText [SANT93a], FLAVER [SANT97c], FLANOM [SANT99b] y GEISA [SANT95b, SANT97b].

En [SANT93a] se trató una aplicación de ayuda a la elaboración de documentos —FrecText— que entre sus características contaba con la realización de algunas estadísticas básicas —cálculo de frecuencia de aparición— y permitía realizar búsquedas según diversos criterios de similitud entre cadenas de caracteres, así como localizar las formas de un verbo que aparecen en un texto.

En [SANT93b] se presentó un programa de conjugación verbal básico que contemplaba algo más de 11000 verbos para los que se habían sistematizado sus irregularidades. El artículo señalaba algunas carencias: no se contemplaban los verbos defectivos ni los participios irregulares que tienen algunos verbos.

En [SANT97c] se diseñó una aplicación informática —FLAVER— útil para lematizar formas verbales —identificaba su infinitivo, categoría gramatical y flexión— y generar una forma verbal a partir de su infinitivo y flexión; en ambos procesos se consideraban las modificaciones debidas a la presencia de pronombres enclíticos y prefijos. También se consideraba la flexión del participio como adjetivo verbal y el diminutivo del gerundio.

En [SANT98] y [SANT99b] se complementaba la anterior con un lematizador y generador de formas nominales —FLANOM— capaz de identificar: forma canónica, categoría gramatical y flexión o derivación. Esta herramienta consideraba: género y número en los sustantivos, adjetivos pronombres y artículos;

heteronimia por cambio de sexo en los sustantivos; grado superlativo en los adjetivos y adverbios; adverbialización y adverbialización del superlativo en los adjetivos; derivación apreciativa en los sustantivos, adjetivos y adverbios; formas canónicas múltiples en todas las categorías gramaticales y formas invariantes. El universo de referencia ascendía a 109 194 formas canónicas.

En [SANT95b] y [ SANT97b] se expuso el desarrollo de una aplicación de gestión de sinónimos y antónimos en español —GEISA— que tenía en cuenta los accidentes gramaticales con los siguientes objetivos: 1) almacenamiento estructurado —minimiza la ocupación y el tiempo de respuesta— de un diccionario de sinónimos y antónimos, 2) posibilidad de consultas sobre el diccionario en un entorno amigable —ventanas y menús desplegables—, 3) devolución del sinónimo y/o antónimo afectado de los mismos accidentes gramaticales que la palabra original, y 4) desarrollo modular que permitía su incorporación ulterior a sistemas de manipulación de textos más complejos.

En resumen, las principales líneas de trabajo del Grupo de Estructuras de Datos y Lingüística Computacional han estado centradas desde 1 990 en torno: 1) al desarrollo de herramientas de reconocimiento y generación de entidades morfológicas, 2) a la búsqueda de sus aplicaciones en el análisis de textos y 3) a la ayuda en la producción textual. Lo que aquí se pretende es poner en marcha nuevos

desarrollos y expandir el ámbito de aplicación a nuevos territorios a partir de los logros ya consolidados.

## 2.- Planteamiento y desarrollo.

El objetivo establecido en la presente tesis es la realización de un conjunto de herramientas especializadas en el análisis de textos orientado prioritariamente al aprovechamiento de las posibilidades de Internet como fuente de obtención de documentos —en su faceta de fenómeno sociológico, se constituye también como una entidad generadora de lenguaje con valor propio desde el punto de vista filológico. Las herramientas desarrolladas deben ser susceptibles de formar parte de lo que podría llamarse una "Estación lingüística orientada a Internet".

El tipo de aplicaciones que se pretende englobar como herramientas de análisis léxico orientadas a Internet comprende dos aspectos de desarrollo: 1) el análisis lingüístico propiamente dicho y 2) el relacionado con la navegación y acceso a los documentos en la red. Como punto de partida, se cuenta con herramientas morfológicas de lematización —las desarrolladas en los trabajos del Grupo de Estructuras de Datos y Lingüística Computacional— y se pretende probar sus posibilidades de integración en aplicaciones de análisis lingüístico. También se cuenta con la experiencia previa en la realización de herramientas de ayuda a la generación de documentos, estudio de estructuras de datos y sus aplicaciones, integración de herramientas informáticas y diseño de interfaces.

Para conseguir el objetivo propuesto, se procede en primer lugar a transformar el motor morfológico de las aplicaciones descritas en [SANT97c], [SANT98] y [SANT99b] en una librería de enlace dinámico (DLL) que proporcione servicios de lematización y generación utilizables desde diferentes aplicaciones.

El siguiente paso consiste en la caracterización general de la tipología de las aplicaciones que será adecuado desarrollar, atendiendo tanto a las necesidades derivadas del análisis de textos como a los imperativos generados por su ubicación en la red. Desde el punto de vista del estudio documental pueden distinguirse dos modalidades de actuación: 1) intensiva y 2) extensiva. La modalidad intensiva se centra en el estudio de un documento desde el mayor número de ángulos posible mediante su disección cuidadosa y paciente hasta obtener el nivel de detalle deseado. Por el contrario, la modalidad extensiva está dirigida al análisis de grandes conjuntos de documentos —probablemente con menor detalle— con unos objetivos centrados en la localización de pautas extendidas de usos lingüísticos que permitan obtener conclusiones más generalizables —existe una correspondencia entre estas dos modalidades de análisis y dos formas corrientes de acceso a los recursos de Internet.

La forma "natural" de usar el WWW es mediante un navegador, que toma una dirección URL —Universal Resource Location—, contacta a través de la red con el servidor al que corresponde y solicita la página asociada; una vez que el navegador

captura el documento actúa como "mostrador" y permite al usuario la visualización de la página, en la que suelen aparecer resaltados los hiperenlaces que contiene —el usuario que pica en uno cualquiera con el ratón envía el navegador a la dirección asociada. El problema radica en que los documentos WWW suelen tener una estructura dispersa debido a su propia naturaleza conceptual; la velocidad de acceso —a veces a un gran número de páginas— impone un coste que dificulta su estudio remoto —la solución la proporcionan los descargadores. A un descargador se le proporciona un conjunto de direcciones y parámetros y obtiene de forma desasistida un conjunto de páginas que almacena localmente para permitir un estudio posterior más reposado —sin la premura ni la demora que impone la conexión a la red.

Al correlacionar los posibles modos de actuación en la red con los posibles modos de aproximación al análisis lingüístico y documental, se decidió desarrollar dos herramientas diferenciadas y complementarias: NAWeb (Navegador y Analizador de Webs) y DAWeb (Descargador y Analizador de Webs). NAWeb es un navegador con las funcionalidades típicas de este tipo de herramientas que aparecen en cualquiera de los navegadores más populares (como Microsoft Internet Explorer o NetScape Navigator) e integra un conjunto de opciones de análisis aplicables discrecionalmente a los documentos accedidos. Aunque DAWeb adopta el aspecto y funcionamiento de una herramienta de descarga de sitios web, lleva a

cabo muchas de las opciones de análisis en línea, por lo que en realidad no precisa descargar y almacenar localmente los documentos accedidos.

Los diferentes factores que se estudian en un texto pueden tener generalmente una contrapartida intertextual —especialmente apropiada para la herramienta que se ha llamado DAWeb— aplicada al estudio comparativo o comprensivo de un conjunto de textos relacionados o relacionables.

Debe preverse que los estudios que se desee realizar no se inscriban exclusivamente en el conocimiento sincrónico, sino más bien en el diacrónico: analizaría la evolución de determinados fenómenos lingüísticos a lo largo de un periodo de tiempo más o menos amplio —por ejemplo, se puede someter a la prensa diaria a un estudio sistemático que permita seguir la aparición, evolución y consolidación o desaparición de neologismos.

Con el planteamiento establecido, las aplicaciones a desarrollar pueden contemplar varias utilidades.

## 2.1.- Detección de neologismos.

La idea es sencilla: si se asume que el conjunto de palabras reconocidas por el lematizador pretende ser completo, cualquier palabra que no reconozca es en principio novedosa y seguramente valdría la pena prestar atención a la misma y al contexto en que aparece. Si resulta un neologismo podría intentarse encontrar otras apariciones en nuevos documentos —a fin de estudiar la extensión de su uso— o seguir su evolución temporal —para ver si se consolida o desaparece. Podría ocurrir que no se tratase de un neologismo, podría ser un nombre propio, o una secuencia especial —en una página web se suele encontrar secuencias alfanuméricas específicas de uso muy particular que no cabría considerar palabras del lenguaje—; también podría ocurrir que una palabra no fuera reconocida porque no constar en la base del lematizador, en cuyo caso habría que incorporarla.



## 2.2.- Relacionadas con la palabra.

Se puede estudiar también el uso de las palabras, sean o no nuevas. Aquí tienen cabida medidas cualitativas basadas en el estudio de una palabra en su contexto; en las aplicaciones desarrolladas deberá facilitarse mediante herramientas

de búsquedas flexibles, así como intervenciones en la interfaz encaminadas a resaltar las situaciones identificadas como resultado de dichas búsquedas.

## 2.3.- Medidas cuantitativas.

También se pueden buscar medidas cuantitativas de uso —frecuencia de aparición, etc. Habría que tener en cuenta que, si bien las frecuencias más bajas corresponden a palabras poco usadas, las frecuencias más altas corresponden a aquellas que algunos autores denominan "gramaticales", y que, en el campo de la recuperación de información se conocen simplemente como "vacías": artículos, preposiciones, conjunciones, etc. —recogidas como otras categorías gramaticales— que actúan a modo de argamasa en las construcciones sintácticas, según las reglas del lenguaje, pero que no son depositarias de información relevante en sí mismas, aunque pueden modular el sentido de la información contenida en el conjunto.

Otro dato de interés es la distribución de las palabras. La idea básica es que la frecuencia por sí sola no proporciona una idea suficiente acerca del uso de una palabra en un texto. Evidentemente, una frecuencia alta indica que una palabra se utiliza bastante, pero no puede equiparase una palabra que concentre todas sus apariciones en una misma sección de un texto con una que se halle uniformemente distribuida a lo largo del mismo, aunque ambas tengan la misma frecuencia; se

trataría por tanto de medir la uniformidad —o falta de uniformidad— de la distribución. Considerando que es un concepto con matices difíciles de reflejar en una única medida numérica, se plantea el cálculo de hasta cuatro variables con las que se pretende proporcionar una visión más rica; no obstante, si lo que se quiere es un sólo número, podría elegirse una de ellas, o combinarse varias del modo que se considere oportuno.

El primer cálculo concierne a la probabilidad real de encontrar ocurrencias de la palabra a intervalos uniformes frente a la probabilidad teórica proporcionada por su frecuencia de aparición: si una palabra aparece **f** veces en un texto de longitud **l**, se obtendría una distribución totalmente uniforme si se hallase una ocurrencia de la palabra a intervalos de distancia **l/f**; en consecuencia, se puede obtener una idea de cuánto se aproxima la distribución real a la ideal dividiendo el texto en intervalos de tamaño **l/f** y contabilizando en qué porcentaje de los mismos se encuentra alguna ocurrencia de la palabra; el resultado es un número entre 0 y 100 que indicará una mejor distribución cuanto mayor sea; sin embargo, este número no lo dice todo; en la figura 1a, se observa cómo una palabra que apareciese cuatro veces en el texto podría alcanzar un valor de 75, distribuyéndose a lo largo de las tres cuartas partes del mismo, pero también podría hacerlo, figura 1b, concentrándose en poco más de una cuarta parte; asimismo, se observa cómo una palabra con frecuencia de aparición

mayor podría obtener un resultado relativamente malo —43,75 para la distribución de la figura— bien por estar concentrada en una parte del texto, figura 1c, o bien por formar pequeñas agrupaciones distribuidas de manera más o menos uniforme, figura 1d.

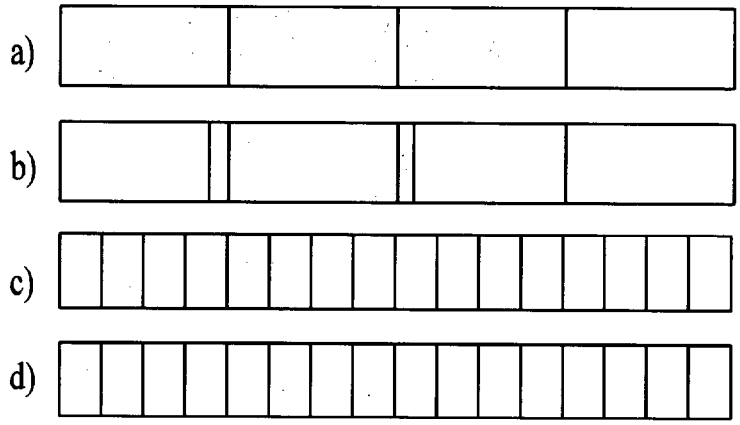



**Figura 1** Uniformidad de la distribución

Una medida complementaria que podría aportar información adicional sería la de calcular el porcentaje de texto comprendido entre la primera y la última aparición de la palabra —*recorrido* de las apariciones de la palabra. De esta manera, una medida de distribución alta con el recorrido pequeño —posiblemente para frecuencias bajas— indicaría una especial concentración en un corto número de intervalos, figura 1b, mientras que una baja distribución con recorrido amplio —especialmente para frecuencias altas— podría indicar una probable distribución en pequeñas agrupaciones, figura 1d.

Se plantean además, figura 2, dos medidas de *centralidad*: 1) interna, que trata de averiguar si se da algún desplazamiento apreciable del centro de gravedad del recorrido —indicaría una tendencia a romper la uniformidad de la distribución— y 2) externa, que sitúa el recorrido en relación al texto para detectar la zona en la que se usa la palabra cuando el recorrido es significativamente menor que el texto.

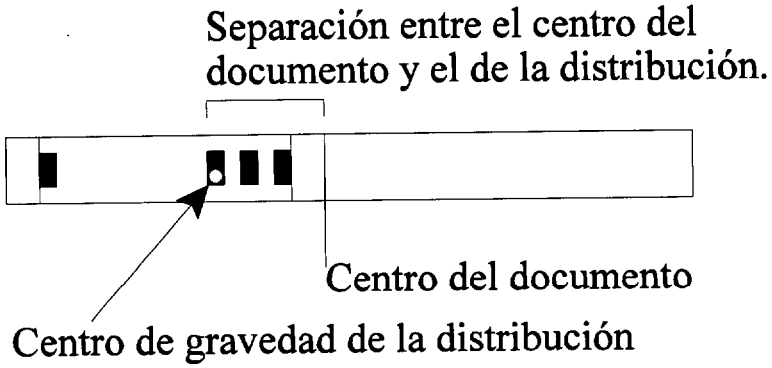



**Figura 2** Centralidades interna y externa

Una observación que quizás sea innecesario realizar es que todas estas medidas deben de estudiarse en relación con la frecuencia de aparición, a la que están profundamente ligadas.

Cabe pensar que puede tener interés calcular todas estas estadísticas —y otras que pudieran plantearse— no sólo en relación con las palabras que aparecen tal cual

en el texto, sino con las formas canónicas de las que derivan. Dado que se dispone de un potente lematizador, no resultaría a priori excesivamente complicado.

## 2.4.- En la proximidad de la sintaxis.

Por encima del nivel de las palabras aisladas, y sin entrar en el desarrollo de herramientas sintácticas complejas, es posible abordar determinados fenómenos relacionados con la coocurrencia de palabras —o formas canónicas. Además de la coocurrencia arbitraria o casual, se tiene conocimiento de construcciones tales como: 1) colocaciones léxicas, término que se emplea para designar combinaciones frecuentes de unidades léxicas que no estén fijadas —en caso de que lo estén, se habla de colocación gramatical—, 2) perífrasis verbales, composiciones formadas por un elemento verbal con morfema de persona seguido de otro en una forma no personal —infinitivo, gerundio o participio— que funcionan como un solo verbo y 3) regímenes preposicionales —abarcando las proposiciones que acompañan a verbos, sustantivos y adjetivos.

# 3.- Arquitectura de DAWeb.

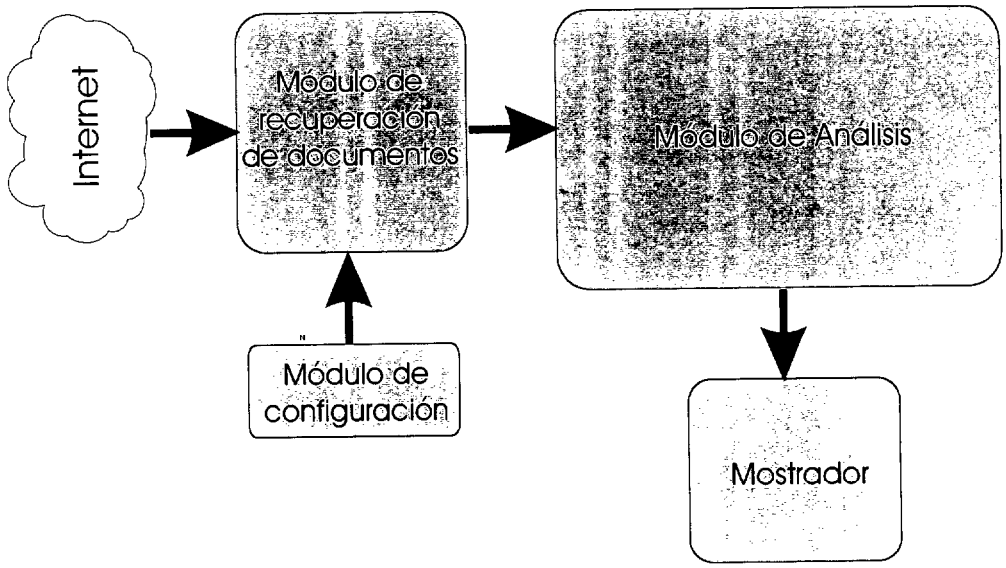



**Figura 3** Arquitectura de DAWeb

DAWeb se halla estructurado, figura 3, en tres módulos principales: 1) *módulo de configuración*, 2) *módulo de recuperación de documentos* y 3) *módulo de análisis* en línea; se complementa con una aplicación externa —programa mostrador— que se encarga de presentar los resultados —generalmente voluminosos— en formas adecuadas para su estudio eficiente.

En las siguientes secciones se habla pormenorizadamente de cada uno de estos módulos, se exponen sus funciones, se detallan las interrelaciones que se

establecen entre los mismos y se justifican las políticas de funcionamiento adoptadas.

## 3.1.- Módulo de recuperación de documentos.

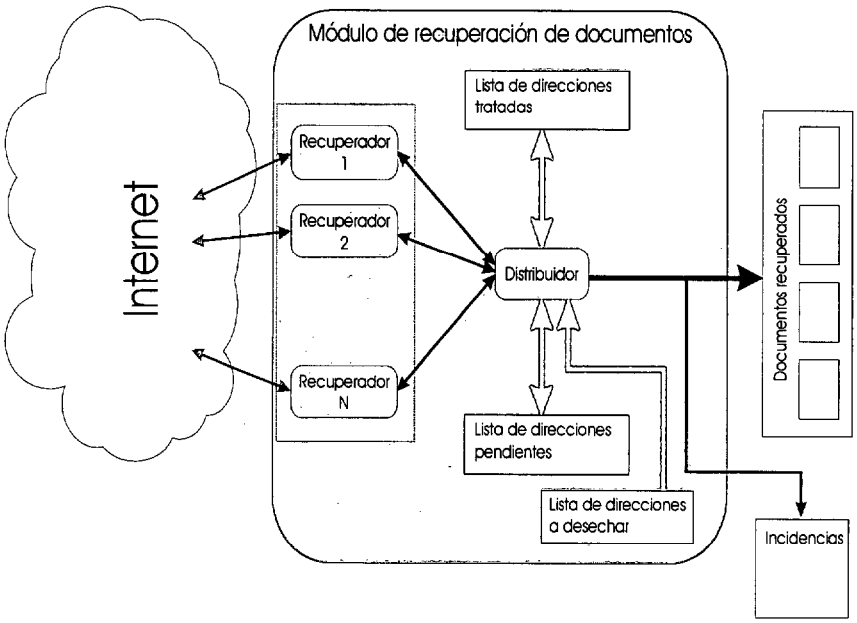



**Figura 4** Módulo de recuperación de documentos

El *módulo de recuperación de documentos* está compuesto, figura 4, por un *módulo distribuidor* y un número variable de *módulos recuperadores* que interactúan con Internet.

### 3.1.1.-El módulo distribuidor.

El módulo distribuidor se encarga de repartir y coordinar el trabajo entre los recuperadores y de recibir los resultados que obtengan; les da forma y los deja preparados para su entrega al módulo de análisis o a cualquier otro que pudiera realizar algún tipo de tarea con los mismos.

El módulo distribuidor toma direcciones de la lista de "direcciones pendientes" —creada inicialmente por el *módulo de configuración de la recuperación*, submódulo del *módulo de configuración*—, y entrega una a cada recuperador hasta que todos tengan la suya o hasta que la lista de direcciones pendientes esté vacía. A partir del momento en que todos los recuperadores tengan una dirección o todas las direcciones hayan sido asignadas, el quehacer del módulo distribuidor consiste en esperar hasta que alguno de los recuperadores obtenga resultados de su gestión; cuando ocurre, el módulo distribuidor requiere el documento que obtiene el recuperador en el acceso a la dirección encomendada y lo incluye en la lista de documentos recuperados —a efectos de esta tesis funciona como una cola de documentos pendientes de analizar—, también interpela al recuperador acerca de la lista de direcciones asociadas a los hiperenlaces del documento conseguido.

Las direcciones que el recuperador se ha encargado de extraer deben confrontarse por triplicado:

1. Con la lista de direcciones pendientes para no duplicar una dirección incluida en la petición inicial u obtenida como resultado de otros accesos.

2. Con la lista de direcciones ya recuperadas, para evitar redundancias en la recuperación.

3. Con los criterios de direcciones para desechar —los establece el *módulo de configuración*— para comprobar que constituyen candidatos aceptables para posteriores expansiones de la búsqueda de documentos en curso.

Las direcciones que salven este triple filtro se añadirán a la lista de direcciones pendientes y contribuirán a engrosar el conjunto de materiales que se obtengan por desarrollo de la petición inicial hasta el límite posible.

Una vez que se ha extraído toda la información que el recuperador es capaz de proporcionar, el distribuidor asigna al recuperador una nueva dirección a partir de la lista de direcciones pendientes y vuelve al estado de espera.

El trabajo del módulo de recuperación de documentos concluye cuando la lista de direcciones pendientes está vacía y ningún recuperador se halla navegando

o intentando navegar —circunstancias de las que se apercibe el módulo distribuidor. Si la lista de direcciones pendientes está vacía, pero algún recuperador está ocupado, podría ocurrir que obtuviese algún documento con hiperenlaces, y habría que esperar para saber si el proceso aún debe continuar.

De fracasar el encargo asignado a un recuperador, el *módulo distribuidor* evaluará las circunstancias que han provocado tal situación y optará por volver a intentar la misma dirección con posterioridad o la desestimará por considerar que no va a ser posible acceder a la página —dirección errónea o acceso fallido y se anota en *Incidencias*—; en cualquier caso, si hay direcciones pendientes le entregará una nueva al recuperador malogrado —en ningún caso reintentará la misma dirección inmediatamente, ya que el recuperador lo habrá probado hasta el límite establecido en la configuración antes de decidirse a comunicar su fallo.

El módulo distribuidor genera un informe de *Incidencias* en el que se relacionan: las páginas solicitadas, las páginas obtenidas y las páginas no recuperadas —indica la causa aparente del fallo.

### 3.1.2.-Los módulos recuperadores.

Cada módulo recuperador es un gestor de transacciones HTTP (HiperText Transfer Protocol) que se ejecuta en un hilo independiente. En paralelo pueden ocurrir: 1) otras recuperaciones, 2) el distribuidor obtiene datos de los recuperadores que hayan finalizado con éxito su gestión y 3) el analizador procesa documentos recuperados con anterioridad. Se pretende aprovechar al máximo el tiempo de uso del procesador, de forma que se minimicen los "tiempos muertos" de espera de unas tareas por la finalización de otras.

El trabajo del recuperador se inicia cuando recibe una dirección por parte del distribuidor; el gestor de transacciones HTTP iniciará una petición de conexión con la dirección indicada y pasará al estado de espera mientras llega una respuesta. Si la respuesta llega en forma de establecimiento de la conexión, comienza la transferencia de paquetes de información hasta completar la recepción del documento asociado a la dirección especificada. En caso de que ocurriera un fallo —por ejemplo que no se encuentre el servidor especificado, que se interrumpa la transferencia, etc.— o se agotara el tiempo de espera, el recuperador volverá a intentar la conexión un número limitado de veces antes de reportar un mensaje de error.

Si se logra completar la recepción de un documento, el recuperador emitirá un mensaje para informar de tal circunstancia al distribuidor y quedará a la espera de instrucciones que normalmente consistirán de dos requerimientos del distribuidor. El primero será para que le sean entregadas las direcciones referenciadas por los hiperenlaces del documento recuperado —extraídas por el recuperador—; se desestimarán todas aquellas que conduzcan a enlaces ajenos al dominio cuyo análisis se solicitó en un principio, a menos que se trate de un redireccionamiento —única medida que se estima válida para "podar" la navegación, ya que otras que limitan su profundidad no son suficientemente adaptables a la diversidad de los dominios web. Lo siguiente que solicitará el distribuidor al recuperador será el documento completo —incluye los códigos de formato HTML (HyperText Markup Language), ya que podrían necesitarse según el uso que se fuera a dar al documento y no es trabajo del recuperador eliminar dichos códigos.

Cuando el recuperador ya ha entregado toda la información de que dispone, lo único que queda es esperar una nueva dirección, o bien la señal de desactivación, si el proceso ha concluido.

El número de recuperadores es configurable entre uno y diez —al inicio de un nuevo proyecto se establece por defecto en cinco. El usuario puede cambiar su número en cualquier momento y cuantas veces estime oportuno mientras que el

proyecto no esté en ejecución. Con esta configurabilidad se consigue una alta flexibilidad para adaptarse a aspectos tales como las condiciones de la red en un momento determinado o a las necesidades concretas de un estudio: con un sólo recuperador se asegura un orden determinista en la llegada de los documentos que no variará entre ejecuciones distintas del proyecto a menos que los hiperenlaces sufran modificaciones.

La figura 5 muestra el efecto de la variación del número de recuperadores en el estudio de una web no demasiado voluminosa y de acceso relativamente cercano: la del Grupo de Estructuras de Datos y Lingüística Computacional del Departamento de Informática y Sistemas de la ULPGC. Están representados el tiempo de descarga y el tiempo total que dura la ejecución del proyecto —incluye el de descarga y el de análisis de documentos. Dado que el analizador funciona en un hilo de ejecución separado, los tiempos no son aditivos —se solapa la descarga de un documento con el análisis de otro.

Se observa que cuando el número de recuperadores es bajo —menor que

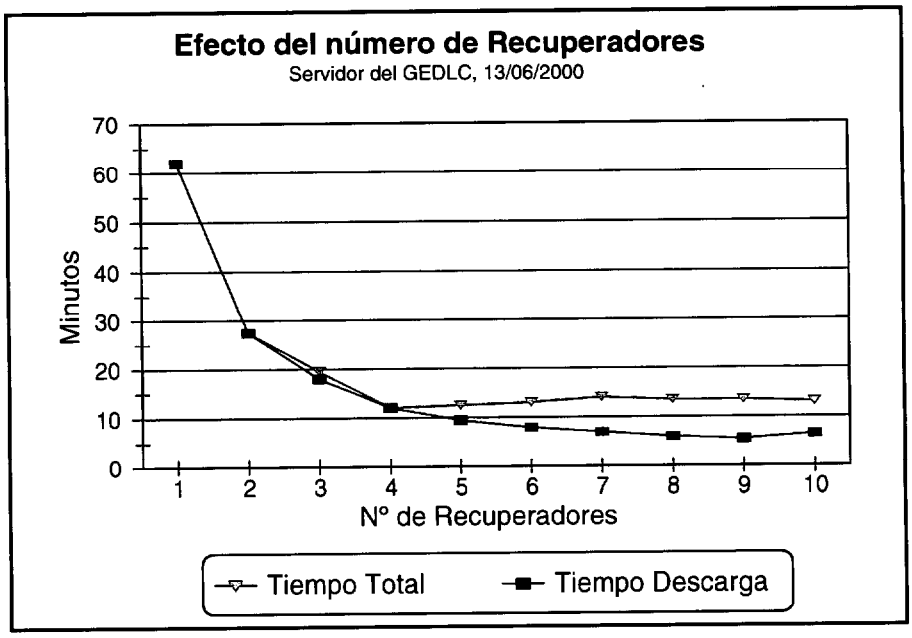



**Figura 5**

cinco—, el tiempo total coincide en la práctica con el de descarga; se debe a que la recuperación es altamente lineal: todo documento que llega se analiza inmediatamente, pero cuando este análisis termina y aún no han llegado otros documentos el analizador debe quedar en situación de espera —en la que pasa la mayor parte del tiempo. La adición de un nuevo recuperador tiene un efecto drástico en la reducción del tiempo de descarga, y en consecuencia, del tiempo total de ejecución.

Cuando el número de recuperadores es superior a cinco, un recuperador más no mejora sustancialmente el tiempo de descarga; la competición de un mayor número de hilos entraña un aumento del tiempo —los recursos empleados en la concurrencia empiezan a consumir los tiempos muertos que su gestión intentaba repartir entre los recuperadores. Dado que la velocidad de análisis es casi constante, un determinado volumen de documentos no podrá analizarse en un tiempo inferior al resultante de dividir la cantidad de palabras que contengan por el número de palabras que el analizador es capaz de resolver por unidad de tiempo; tal barrera no puede rebajarse por medio de los recuperadores que, al fin y al cabo se mueven en otra dimensión; en consecuencia, con suficiente número de recuperadores, los documentos deben esperar para ser tratados por un analizador que no está nunca ocioso y el tiempo total viene condicionado por el tiempo de análisis.

Ya que el freno parece ponerlo la velocidad de análisis, podría pensarse en disponer de más de un analizador en paralelo, pero tal disposición carece de sentido; por actuar en la máquina local y a su máxima velocidad, el analizador no deja tiempos muertos significativos que pudieran aprovecharse para mejorar el rendimiento como ocurre en el caso de la recuperación, que sí depende de factores externos no controlables localmente. Las mejoras que puedan obtenerse en cuanto a velocidad de análisis han de conseguirse por otras vías: principalmente, por la

optimización del uso del analizador en la dirección de aprovechar el trabajo ya procesado gracias a la frecuente repetición de palabras en cualquier texto.

### 3.2.- Módulo de análisis de documentos.

Como muestra la figura 6, el *módulo de análisis de documentos* está integrado por un total de ocho submódulos: 1) de *extracción de texto*, 2) *gestor de análisis*, 3) *selector de palabras*, 4) *gestor de segmentos*, 5) *optimizador de búsquedas morfológicas*, 6) *reconocedor morfológico*, 7) *recuentos* y 8) *entrega de resultados*.

Cuatro de estos módulos —el de *extracción de texto*, el *selector de palabras*, el *optimizador de búsquedas morfológicas* y el *reconocedor morfológico*— se tratarán separadamente en la sección 5 —Módulos comunes— por estar compartidos con NAWeb.

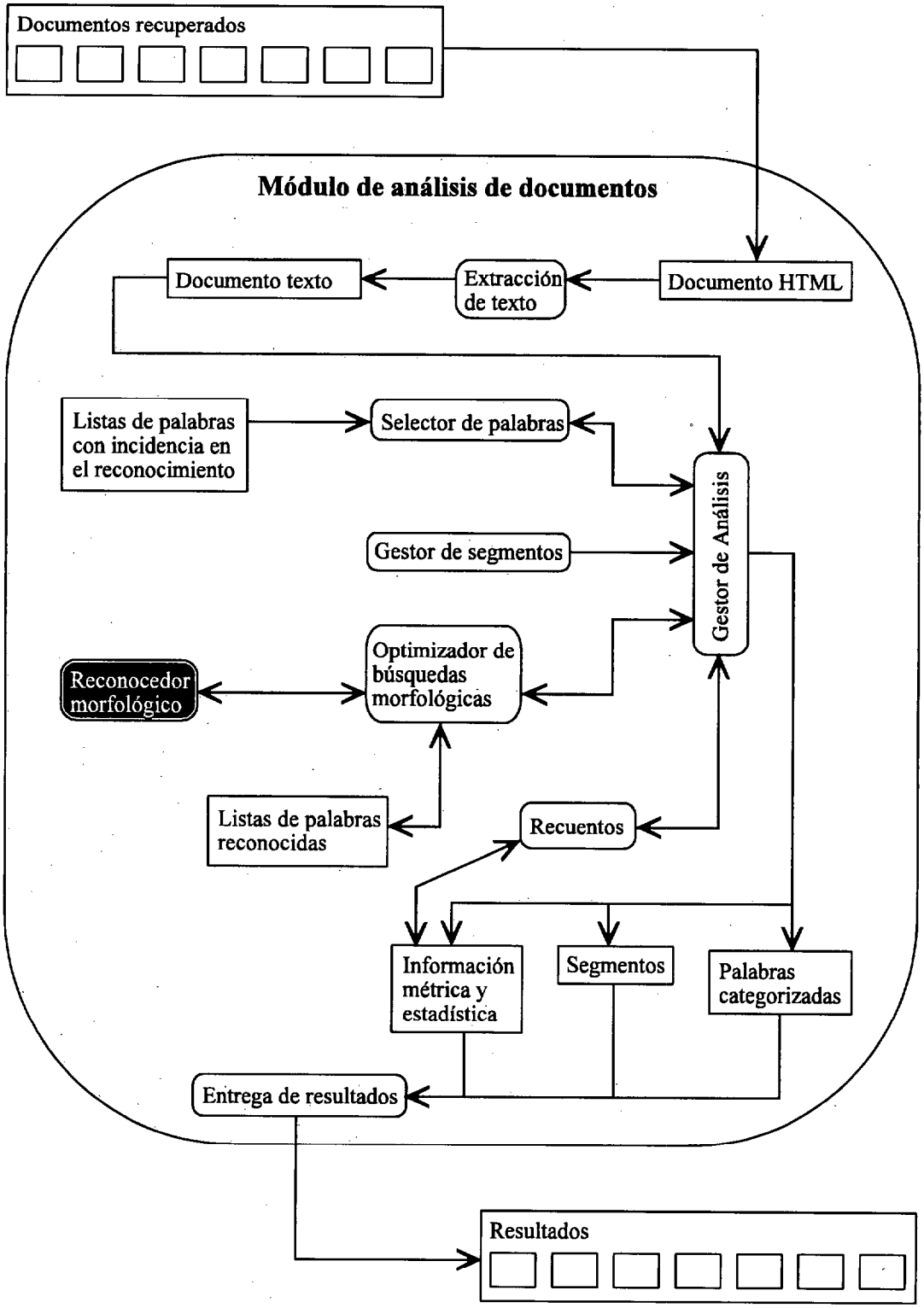



**Figura 6**: Módulo de análisis de documentos

El módulo de recuperación pone en cola los documentos recuperados a la espera de que les llegue el turno para ser analizados. El *módulo de análisis* toma los documentos de esta cola pero deberá eliminar las marcas de formato que confieren al documento su aspecto visual —tarea de la que se ocupa el *submódulo de extracción de texto*. En este momento entra en juego el *gestor de análisis* para conducir el proceso: mediante invocaciones al *selector de palabras* separa las palabras y echa mano del *reconocedor morfológico* —complementado con un módulo *optimizador de búsquedas morfológicas*— para lematizarlas y poder realizar los análisis configurados. La actividad del *gestor de análisis* está condicionada por la configuración establecida, mediante la cuál se determina qué se analiza —se pueden configurar listas de "palabras vacías", que no se tendrán en cuenta, o "listas de palabras significativas", cuyas apariciones se buscarán—, qué análisis se hacen —reconocimiento de palabras, estadísticas, segmentos, etc.— y qué resultados se proporcionan. Los submódulos de *gestión de segmentos* y *recuentos* intervienen en el análisis: el primero, localizando los segmentos cuya búsqueda se haya configurado y el segundo, realizando los cálculos estadísticos pertinentes.

El submódulo de *Entrega de resultados* se encarga de elaborar los informes sobre los resultados de los análisis —quedan almacenados para su estudio posterior fuera de línea. La generación de estos informes se lleva a cabo de manera

incremental —se añade la información que aporta el análisis de un documento en cuanto concluye—, de forma que se minimice el riesgo de pérdida de información ya disponible si ocurriera una interrupción abrupta del proceso que no supusiera la destrucción del soporte de la información.

Tal como se muestra en la figura 7, los resultados se organizan según una jerarquía dimanante del binomio dominio-página —por dominio se entiende la dirección de partida de un proceso—. El listado de *Información por página* está compuesto por un registro de información por página analizada, y consta de seis campos: 1) la dirección del dominio desde el que se ha accedido a la página —es necesario debido a que las páginas se recuperan y analizan en un orden desconocido, de manera que la información de dominios diferentes aparecería entremezclada—, 2) La dirección de la página a la que se refiere la información del registro, 3) una pareja de enlaces que identifica la región que corresponde a la página en el listado de *Palabras por página*, 4) otra pareja de enlaces que identifica la región que corresponde a la página en el listado de *Palabras no reconocidas*, 5) otra pareja de enlaces más que identifica la región que corresponde a la página en el listado de *Segmentos* y 6) un enlace que relaciona la página con el listado de *Perfiles*.

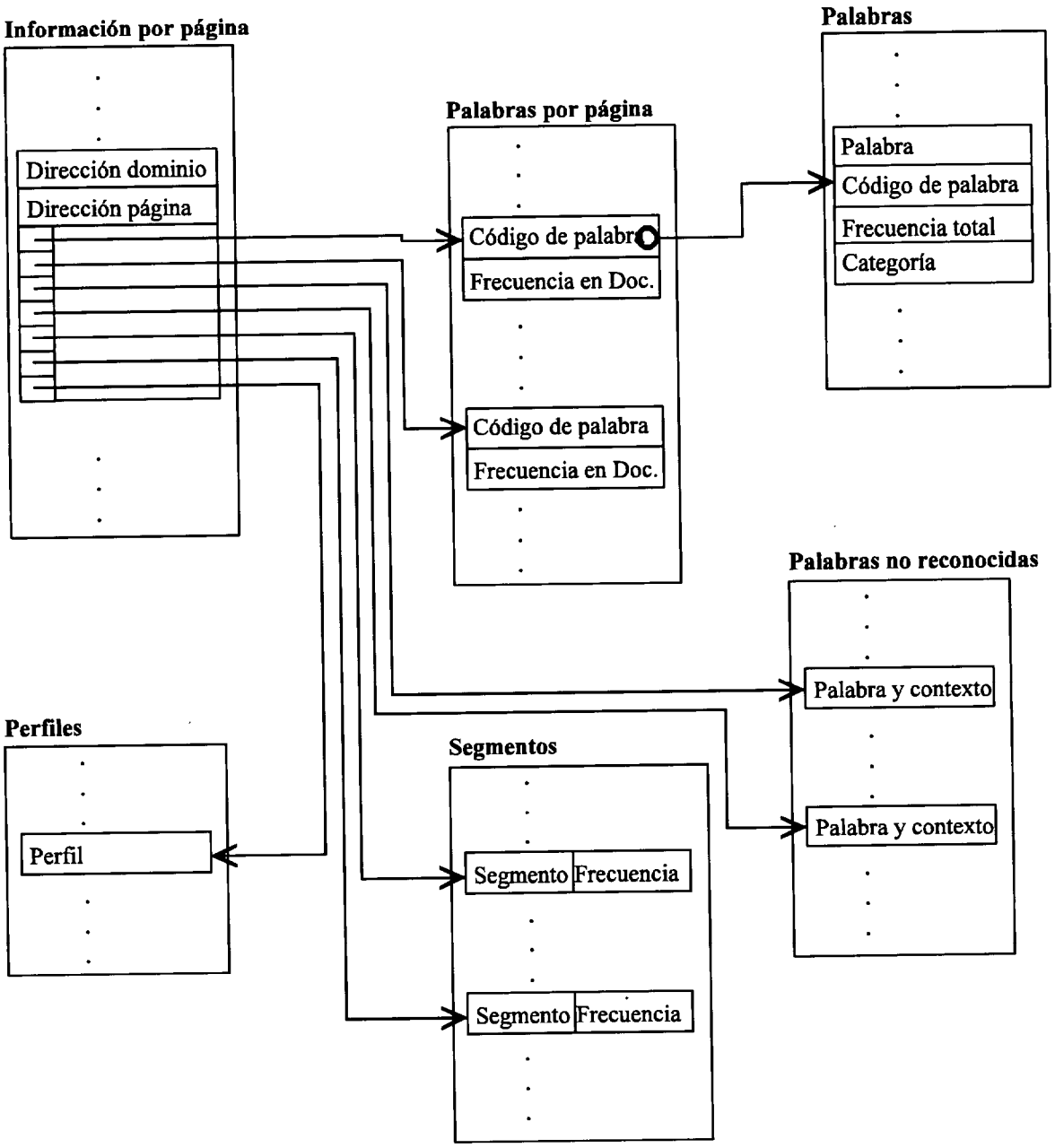



**Figura 7** Estructura de los resultados de DAWeb

El listado de *Palabras por página* está formado por un conjunto de pares de

valores numéricos —un *Código de palabra* y su frecuencia de aparición en la página. El *Código de palabra* sirve para localizar la información de la palabra que representa en el listado general de *Palabras*; tal listado contiene la siguiente información acerca de todas las palabras encontradas y reconocidas: la *Palabra*, su *Código* de identificación, su *Frecuencia de aparición total* y su *Categoría* gramatical. Las palabras no reconocidas se incluyen en el listado de *Palabras no reconocidas,* tantas veces como aparezcan y acompañadas en cada ocasión por un extracto del contexto en que aparecen —una región de unos 80 caracteres en torno a la palabra. El listado de *Segmentos* contiene los segmentos que cumplan con los requisitos de longitud y frecuencia mínima que se establecieron en la configuración junto con su frecuencia de aparición. El listado de *Perfiles* contiene para cada página un *Perfil* de la forma en que se incorporan las palabras al texto: se divide el texto en partes y se calcula para cada parte el número de palabras que aparecen por primera vez frente al número total de palabras que se usan —es un dato que permite hacer comparaciones, incluso gráficas, entre textos, particularmente si tienen una extensión parecida.

Toda la información generada queda almacenada en un conjunto de ficheros agrupados bajo una carpeta cuya denominación se forma mediante la unión del nombre dado por el usuario en el momento de configuración del proyecto con la

fecha y hora en la que éste fue ejecutado —los resultados obtenidos en instantes diferentes quedan así perfectamente identificados y separados.

### 3.3.- El Mostrador.

Como complemento a DAWeb se ha desarrollado un programa que permite la visualización general de los resultados obtenidos —sin que ello excluya la posibilidad de realizar programas específicos cuando se crea conveniente para un análisis preciso de los mismos en función de necesidades particulares. El programa está organizado en múltiples ventanas: la ventana inicial permite seleccionar el proyecto cuyos resultados se quieren ver —opción *Archivo/Abrir*— y mostrar información general del mismo, figura 8.

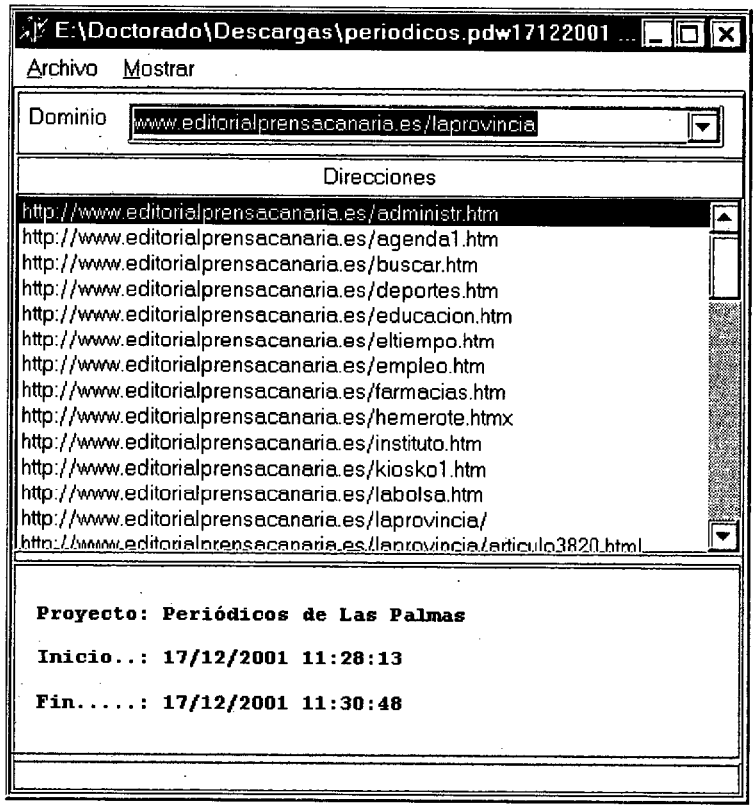



**Figura 8** Ventana principal del mostrador

La entrada *Dominio* permite seleccionar cualquiera de los dominios —direcciones de partida— incluidos en el proyecto; tal acción actualiza automáticamente la lista de direcciones con las propias del nuevo dominio. Debajo de la lista de direcciones aparece el título del proyecto y las fechas y horas de comienzo y finalización de la ejecución cuyos resultados se muestran. La opción del menú *Mostrar* permite elegir la visualización de otras cinco ventanas: palabras *reconocidas* y *no reconocidas*, secuencias *no alfabéticas*, segmentos y *perfiles* de los documentos.

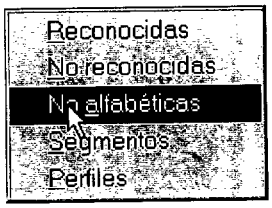



**Figura 9** Submenú de *Mostrar*

La información mostrada en cada una de estas cinco ventanas está dividida en tres secciones: información total del proyecto —*Totales*—, del conjunto del dominio seleccionado —*Por dominio*— y de la dirección individual seleccionada —*Por Dirección*, figura 10. Las secciones *Totales* y *Por Dominio* se hallan divididas a su vez en tres tablas que presentan la información ordenada según los criterios de orden alfabético, de frecuencia de aparición y de distribución —el número de documentos diferentes en que aparece, carece de sentido en la sección *Por Dirección*. La ventana de *palabras reconocidas* recoge información acerca de la categoría gramatical de las palabras —no disponible en el resto de las ventanas. Todas las ventanas tienen en la parte superior una hilera de botones que permite abrir cada una de las otras.

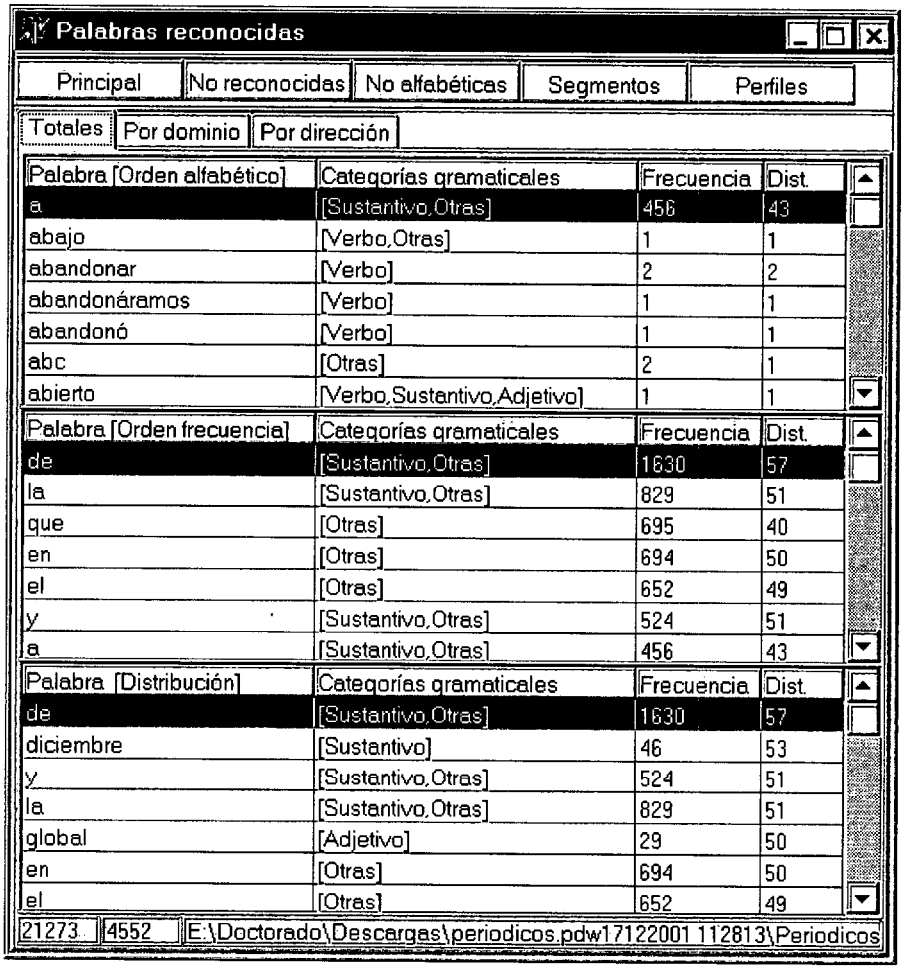



**Figura 10** Ventana de palabras reconocidas

La sección *Por dirección* de las ventanas de *No reconocidas* y *No alfabéticas* se diferencian del resto en que muestra la información de contexto de aquellas secuencias de caracteres que no han podido identificarse, figura 11.

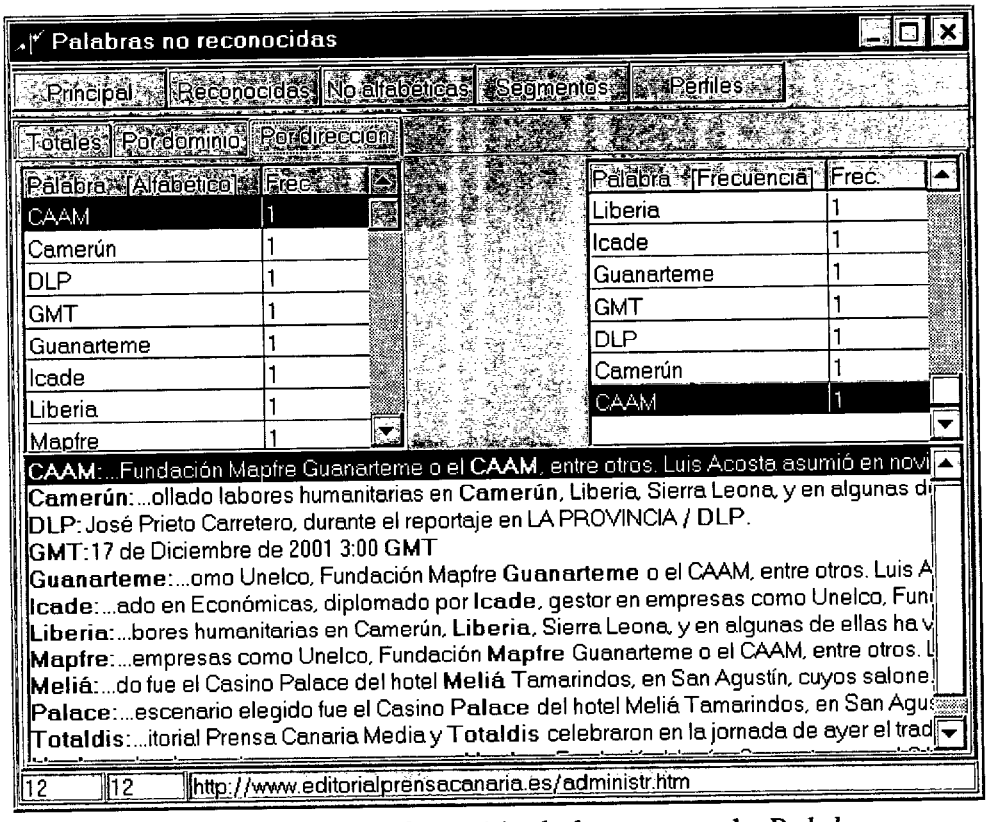



**Figura 11** Sección *Por dirección* de la ventana de *Palabras no reconocidas*.



## 3.4.- El módulo de configuración.

El módulo de configuración recoge los datos introducidos a través de la interfaz y gestiona su almacenamiento y acceso en un fichero de configuración. Cuenta con un submódulo de automatización que permite configurar ejecuciones programadas de los proyectos. La ejecución programada de un proyecto se configura estableciendo una fecha y hora de ejecución; a partir de ahí el submódulo de

automatización se encarga de examinar periódicamente los proyectos pendientes y si alguno hubiera alcanzado la hora programada lo ejecuta. Durante la ejecución del proyecto, el submódulo de automatización queda suspendido para evitar que varios proyectos entren en competencia.

# 4.- Arquitectura de NAWeb.

NAWeb se compone, figura 12, de 7 módulos principales: 1) el de *navegación*, 2) el de *extracción de texto*, 3) el de *lematización*, 4) el de *desambiguación* 5) el de *clasificación*, 6) el de *búsqueda* y 7) el de *exportación de resultados*.

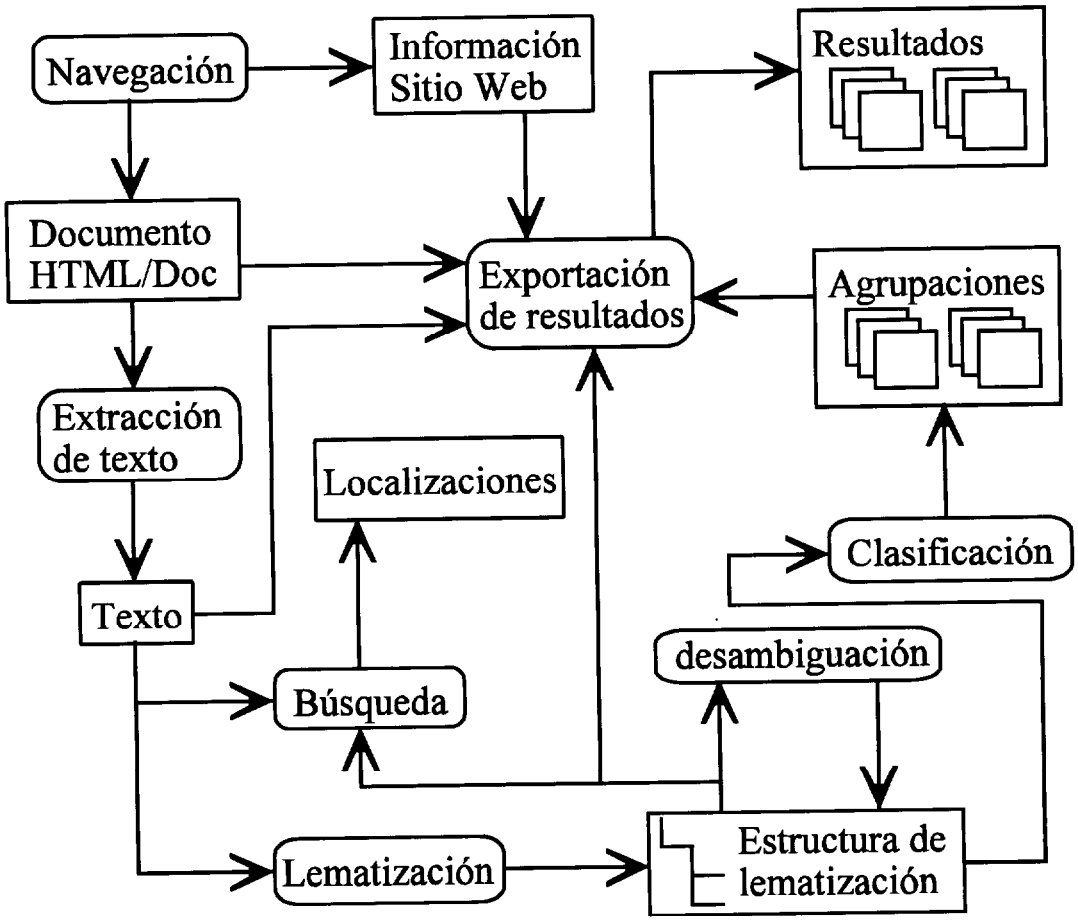



**Figura 12** Arquitectura de NAWeb

El módulo de *navegación* es un componente TWebBrowser estándar que aporta a la aplicación las funcionalidades básicas de Microsoft Internet Explorer —se ha elegido en vez de los componentes TNMHTTP (gestores de transacciones HTTP) usados en DAWeb dado el distinto carácter de la aplicación que requiere una relación más activa con el usuario. El módulo de *extracción de texto* es el mismo que se utiliza en DAWeb para los documentos en formato HTML. Se ha dotado a NAWeb de la capacidad adicional de acceder a documentos en formato MS-WORD, lo que abre un enorme campo de trabajo en modo local que resulta muy útil —aunque el objetivo principal de está tesis es el trabajo con Internet—; para los documentos en formato MS-WORD, el módulo de extracción de texto se comporta de modo transparente, ya que el módulo de navegación tiene la capacidad de extraer este tipo de textos. Del resto de los módulos se habla con detalle en los siguientes apartados.



## 4.1.- El módulo de lematización.

El módulo de lematización, figura 13, trabaja sobre el texto extraído del documento accedido una vez que el *modulo de extracción de texto* lo ha privado de las marcas de formato y genera la *estructura de lematización* del documento. Sus componentes básicos son: 1) un *Gestor de análisis*, 2) un *Selector de palabras*, 3) un

*Optimizador de búsquedas morfológicas* y 4) un *Reconocedor morfológico*. Los tres últimos componentes son los mismos que sus homónimos de DAWeb; el Gestor de análisis tiene la función de crear la *Estructura de lematización* con la información que proviene del selector de palabras y los módulos morfológicos.

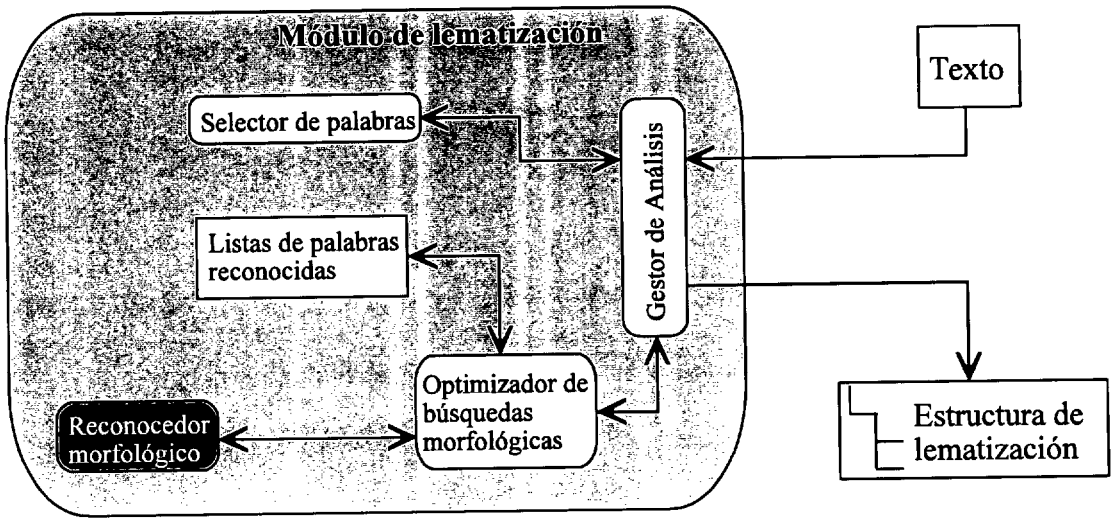



**Figura 13** Módulo de lematización

El módulo *optimizador de búsquedas morfológicas* alcanza su máxima eficiencia cuando el volumen de documentos analizados es alto; en el caso de la aplicación interactiva NAWeb, aunque la orientación a un usuario que se concentra en el análisis detallado de uno —o unos pocos— documentos cada vez impide que se llegue a alcanzar el máximo de prestaciones, se aprecia la mejora del rendimiento

a medida que se analizan nuevos documentos. La ventaja de tener precargadas un conjunto selecto de palabras de uso muy frecuente —en general o en entornos particulares— se aprecia mejor que cuando se hacen descargas masivas, como en DAWeb —el aumento que se produce en la velocidad de análisis, inmediato y nada despreciable, queda diluido frente a la velocidad promedio final que se alcanza con grandes volúmenes de información

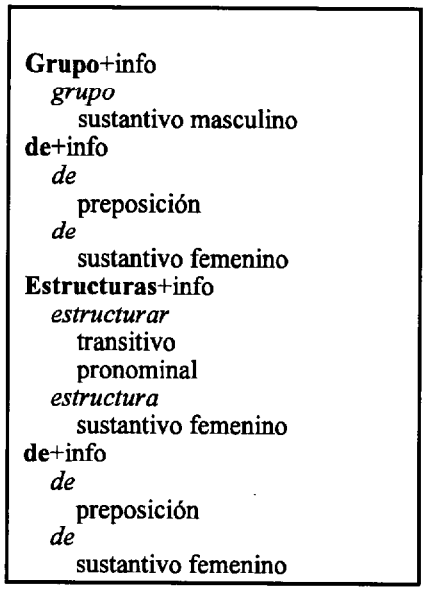



**Figura 14** Estructura de lematización

El producto del módulo de lematización —*Estructura de lematización*— es una estructura jerárquica en la que se almacenan: 1) todas las palabras encontradas en el texto, con información sobre su ubicación —orden secuencial en el texto y

posición medida en caracteres—, así como de los signos de puntuación que las anteceden y suceden —permite tener una información mínimamente elaborada acerca de la estructura del texto— 2) para cada palabra, las formas canónicas de las que podría provenir con su categoría gramatical, flexión y demás características del reconocimiento.

## 4.2.- El módulo de desambiguación.

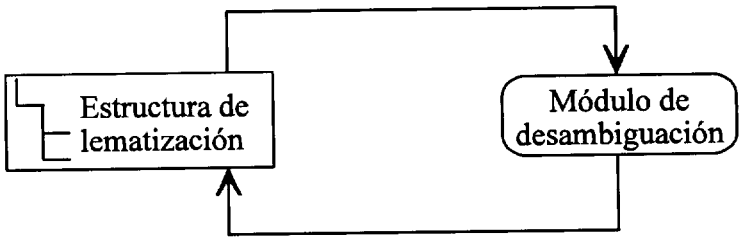



**Figura 15** El módulo de desambiguación

La mayoría de las palabras que se reconocen pueden cumplir más de una función gramatical. El reconocedor informa de todas las posibles categorías gramaticales y flexiones con las que concuerda una forma dada, pero no determina —no es su función— cuál es el papel que juega en un texto determinado. Cuando se

utiliza el reconocedor morfológico sobre un documento, lo que se obtiene es una lista ordenada de palabras, cada una con una lista de posibles formas canónicas de las que puede provenir, y, para cada forma canónica, una lista de categorías y flexiones con las que concuerda —algunas de las palabras estarán marcadas como "no reconocidas" por no figurar en la base de datos del reconocedor. En una primera aproximación, tal reconocimiento basta para un gran número de ocasiones, pero los estudios más afinados requieren que se pueda identificar con exactitud el papel de cada palabra.

Los dos problemas que impiden conocer exactamente la función de cada palabra son: 1) el de las palabras no reconocidas —por no figurar en la base— no tiene grandes posibilidades de automatización: la solución pasaría porque el usuario les asigne manualmente una categoría —habría que tenerlas en cuenta en el futuro para su adición a la base de datos en sucesivas revisiones—; para aliviar el trabajo del usuario, se podría evitar que examinara toda la lista de palabras y que fuera el programa el que buscara y mostrara —en su contexto— las palabras que previamente ha marcado como "no reconocidas" y diera la opción de elegir una categoría para las mismas —en función del contexto de la palabra, incluso podría recomendar alguna— y 2) el problema del reconocimiento múltiple también se puede atacar de forma manual, pero dado su mayor volumen —el número de palabras no reconocidas es

mucho menor— y la existencia de opciones donde elegir, parece factible la utilización de algún mecanismo de desambiguación automático —al menos parcial.

El problema de la desambiguación morfológica se ha tratado tradicionalmente mediante dos técnicas distintas: 1) los métodos probabilísticos basados en la estadística —predominantes desde principios de los 80— resuelven casi todas las ambigüedades, pero a costa de una alta tasa de error y 2) los modelos basados en reglas cometen pocos errores, pero dejan ambigüedades sin resolver. La mayoría de los sistemas estocásticos obtienen sus probabilidades a partir de corpus etiquetados manualmente; también se usan lematizadores basados en modelos de Markov y derivados sobre corpus no etiquetados que obtienen altas tasas de éxito; aunque algunos desarrollos basados en reglas no les van a la zaga.

La desambiguación ha sido abordada por el Grupo de Estructuras de Datos y Lingüística Computacional como línea de investigación en un periodo relativamente reciente y, aunque ya se han obtenido interesantes resultados en la identificación y clasificación de reglas de desambiguación, no cabe duda de que aún están por alcanzarse mayores logros.

En el ámbito de esta tesis, se ha seguido el mismo criterio de modularización aplicado al resto de los elementos, de manera que se ha incluido un módulo de

desambiguación basado en el estado actual de los trabajos desarrollados que puede ser sustituido sin problemas a medida que se obtengan resultados más afinados. Básicamente, lo que se ha hecho es aprovechar el cuerpo de reglas existente para aplicarlas al resultado de la lematización del texto de un documento. En principio, y dado que, aunque sea poco, la desambiguación requiere un consumo extra de recursos y tiempo, y quizá no se precise siempre, se ha optado por no aplicarla de modo automático —figura como una opción que el usuario debe activar cuando le convenga sobre un texto previamente lematizado.

El proceso de desambiguación automática se efectúa recorriendo la estructura resultante de la lematización de un documento. Se localizan las palabras que tengan un grado de reconocimiento superior a uno —con más de una posible lematización. Cuando se encuentra una palabra en estas circunstancias, se toma en cuenta la palabra anterior y la siguiente y se prueba qué combinaciones de categorías resultan válidas y cuáles no.

El programa también admite la posibilidad de que el usuario asigne categorías a las palabras procediendo de esta manera a una "desambiguación manual" que puede ser útil en pequeñas dosis, sobre todo para proporcionar puntos de apoyo a la desambiguación automática en textos especialmente complejos.

## 4.3.- Módulo de clasificación.


**Figura 16** Módulo de clasificación

El módulo de clasificación tiene por objeto generar listas con las palabras del texto agrupadas según diversos criterios; consta de dos partes: 1) el módulo de *clasificación métrica* y 2) el módulo de *clasificación morfológica.*

El módulo de *clasificación métrica* no depende realmente del resultado de la lematización, ya que lo que ofrece son ordenaciones de las palabras en función de criterios tales como su frecuencia de aparición, su relación alfabética directa o inversa y su longitud —características que pueden calcularse directamente a partir del texto sin lematizar. Excepto las que tienen que ver con el cálculo de distancias entre palabras, todas las clasificaciones métricas se hacen cuando se analiza el texto; la clasificación por distancias —se activa por separado— ordena por cercanía a la que el usuario selecciona del conjunto de las palabras del texto —tiene que realizarse cada vez que el usuario elija una palabra distinta—; se han considerado dos posibles clasificaciones —el usuario escoge la que desea— en función de la distancia de

Levenstein [WEB06,WEB07] o de la subsecuencia común más larga [COR90] —no necesariamente contigua— entre la palabra elegida y las demás.

El módulo de *clasificación morfológica* distribuye las palabras del texto agrupándolas por sus categorías gramaticales; obtiene listas separadas de los verbos, sustantivos, adjetivos y otras formas, así como de las palabras no reconocidas y secuencias alfanuméricas no clasificables como palabras. La clasificación se realiza inicialmente con el resultado de la lematización y es reconsiderada si se lleva a cabo algún proceso de desambiguación o de asignación de categorías a las palabras clasificadas como no reconocidas. Las palabras con ambigüedad se clasifican según todas sus posibilidades.

# 5.- Módulos comunes.

Las dos herramientas desarrolladas tienen una orientación divergente hacia formas de aplicación claramente diferenciadas; como consecuencia presentan un diseño muy distinto entre sí. Ambas herramientas intentan llevar a cabo el análisis de documentos obtenidos en Internet y hay determinadas tareas —relacionadas fundamentalmente con la manipulación de los textos obtenidos— que deben realizar de modos muy parecidos, por lo que se ha considerado lógico desarrollar para su ejecución módulos únicos que puedan ser compartidos —otras tareas que aparentemente estarían en las mismas circunstancias, como la navegación, requieren implementaciones específicas adaptadas a las respectivas filosofías. Los módulos comunes compartidos por ambas herramientas son: 1) el de extracción de texto, 2) el selector de palabras, 3) el de análisis morfológico y 4) el optimizador de búsquedas morfológicas.

## 5.1.- Módulo de extracción de texto.

Una página o documento web es básicamente un texto etiquetado utilizando lenguaje HTML. Las etiquetas determinan el aspecto visual que tomará la página al

ser mostrada en la ventana de un navegador —fija los colores, la disposición y la estructura del texto y otros muchos detalles. Para el tipo de análisis que se pretende realizar —morfológico y morfoestadístico—, las etiquetas HTML no suelen ser relevantes y cuando aportan información útil lo hacen desde un punto de vista sintáctico o de análisis de las partes del texto. En ningún caso son susceptibles de ser analizadas, dado que no forman realmente parte del texto.

El entorno de programación Delphi incorpora un componente llamado TWebBrowser que ofrece operaciones para extracción del texto de una página, pero dichas operaciones no resultan plenamente satisfactorias para este tipo de aplicación. La primera razón es que el componente TWebBrowser, diseñado con la perspectiva de un navegador interactivo, no es operativo en aplicaciones del tipo "descargador" que realizan gran cantidad de operaciones de conexión y recuperación, además de manera concurrente. Ha sido demostrado empíricamente que, en un régimen de trabajo como el descrito, el componente TWebBrowser no es capaz de reutilizar adecuadamente los recursos del sistema —deriva en una curva exponencial hasta el colapso. En consecuencia, DAWeb se implementa utilizando componentes TNMHTTP como recuperadores, ya que trabajan muy bien realizando transacciones masivas en régimen concurrente —su robustez ha quedado patente en las pruebas realizadas, donde ha sido capaz de descargar más de 20 000 páginas con hasta 10

hilos en paralelo; por contra, entregan como resultado un documento HTML, sin ninguna opción para la extracción automática del texto útil.

Independientemente de cómo se realicen las transacciones, podría pensarse en utilizar localmente un componente TWebBrowser, no con funciones de navegación, sino exclusivamente como extractor de texto. La forma de actuar, en este caso, sería alimentarlo con los documentos obtenidos por los recuperadores, utilizando sus operaciones de extracción para obtener el texto sin etiquetas HTML; dado que el funcionamiento del analizador es lineal, que obtiene los documentos de la cola donde los sitúa el módulo de recuperación y que analiza cada uno completamente antes de pasar al siguiente, no surgiría ningún problema en relación con la gestión de los recursos del sistema. Sin embargo, el texto que se obtiene de esta manera sigue sin ser del todo válido. La principal razón es que las operaciones de extracción de texto del componente TWebBrowser se limitan, casi, a quitar mecánicamente todas las marcas HTML, y ello puede producir que se junten elementos del texto que son distintos. Por ejemplo, el siguiente texto:

Esto es un ejemplo<br>esto es otro.

Se muestra en dos líneas separadas en un navegador:

Esto **es** un ejemplo
esto es otro.

Pero al ser sometido al proceso de extracción produce:

Esto es un ejemploesto es otro.

Este resultado se debe a que las etiquetas HTML han sido eliminadas sin ninguna consideración adicional. Esto es correcto en el caso de la etiqueta de estilo negrita (, ), pero no lo es en el caso de la etiqueta de ruptura de línea (<br>), que al desaparecer, provoca que se junte la última palabra de una línea con la primera palabra de la siguiente; es necesario, en consecuencia, desarrollar un módulo de extracción "inteligente" que trate las etiquetas HTML según su tipo.

Las etiquetas HTML se dividen en: 1) las que definen elementos embebidos en el texto sin romper el flujo del mismo —la etiqueta de estilo de fuente negrita pertenece a esta categoría— y 2) las que definen elementos de ruptura del texto —saltos de línea o cambio de párrafo, entre otras muchas. Las primeras deben ser eliminadas. Las que definen elementos de ruptura son sustituidas por una marca especial que el *gestor de análisis* emplea como separador — proporcionan

información sobre la estructura del texto que puede resultar útil más allá de los objetivos de esta tesis. Hay que tener en cuenta además, las marcas de caracteres especiales —permiten usar acentos o alfabetos nacionales— que al desaparecer deben sustituirse por el carácter correspondiente.

El módulo de *extracción de texto* es un analizador sintáctico de una pasada, constituido por un autómata de transición de estados gobernado por la secuencia finita de caracteres del documento. En el estado inicial, $E_0$, el autómata avanza por la secuencia de caracteres mientras no encuentre uno de los símbolos "&" o "<" —el primero señala un código de carácter especial incrustado en el texto y el segundo indica el inicio de una etiqueta HTML. Si se encuentra un "&", se pasa al estado $E_a$ y si el carácter siguiente es "#" —significa que el código incrustado está expresado como un valor numérico— se pasa al estado $E_{a1}$ y si no —secuencia alfabética— se pasa al $E_{a2}$; en ambos casos se acumula la secuencia de caracteres que sigue hasta encontrar ";" —el símbolo de finalización del código de carácter especial— y se accede con esa secuencia a una tabla que muestra el carácter que corresponde a la misma —a cada uno de estos dos estados le corresponde una tabla diferente ya que con secuencias distintas se debe obtener un mismo carácter—; la secuencia comprendida desde el "&" hasta el ";" —ambos inclusive— se sustituye por el

carácter indicado en la tabla correspondiente[1] y se regresa al estado $E_0$; si no se encuentra el ";" se entiende que el carácter "&" debe aparecer tal cual en el texto por no disponerse de una secuencia de sustitución correctamente construida.

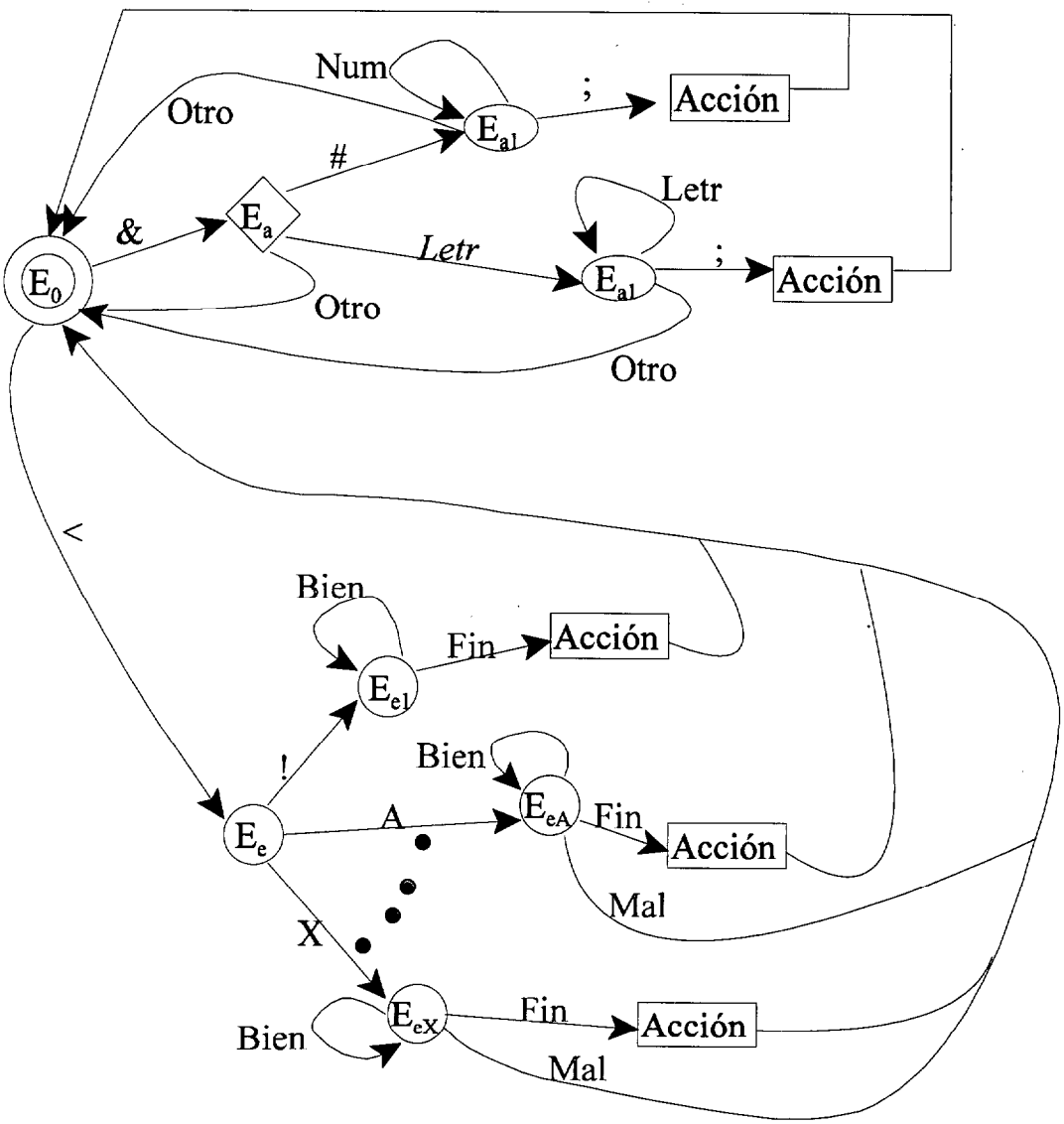



**Figura 17** Esquema idealizado del autómata

[1] El anexo I muestra las correspondencias entre secuencias y caracteres.

Cuando en el estado $E_0$ se encuentra el símbolo "<" se pasa al estado $E_e$ en el que se inicia el análisis del tipo de etiqueta para determinar la actuación que se va a realizar. Si el carácter que sigue al símbolo "<" es "!", en el estado $E_{e1}$ se discrimina entre un comentario —se pasa al estado $E_{e2}$ en el que se avanzan todos los caracteres hasta el final del comentario— o una definición de tipo de documento SGML —se elimina en el estado $E_{e3}$. Si el carácter que sigue al símbolo "<" es una de las letras con las que comienza una etiqueta[2] —"A", "B", "C", "D", "E", "F", "H", "I", "K", "L", "M", "N", "O", "P", "Q", "S", "T", "U", "V", "X"—, se pasa al estado correspondiente —$E_{eA}$, $E_{eB}$, $E_{eC}$, etc.— en el que se examinan los siguientes caracteres hasta que sea posible determinar de qué etiqueta se trata y qué acción hay que tomar respecto a ella: eliminar sólo la propia etiqueta o eliminar todo el texto comprendido entre ella y la etiqueta de cierre que la empareja —por ejemplo, si la etiqueta es <APPLET> todo lo que se encuentre hasta la etiqueta </APPLET> no es realmente texto, sino una llamada a una aplicación elemental en lenguaje Java y desde la óptica del análisis textual planteado carece de valor. Cuando en el estado $E_e$, el símbolo "/" sigue al "<" se trata de una etiqueta de cierre que empareja con otra de apertura ya eliminada, sin que se requiera la desaparición del texto encerrado entre el par de etiquetas; la solución radica en forzar una llamada reentrante en el estado

[2] En el anexo II se relacionan las etiquetas con las acciones que le corresponden.

$E_e$ con el siguiente carácter para eliminar sin esfuerzo adicional la etiqueta de cierre. Tras eliminar los códigos etiquetados se regresa nuevamente al estado $E_0$ para continuar la exploración del texto.

## 5.2.- Módulo selector de palabras.

El *selector de palabras* realiza un proceso de exploración progresiva del texto que proporciona el submódulo limpiador: en la primera invocación avanza desde el principio seleccionando los caracteres hasta formar la primera palabra; en las subsiguientes, retoma la exploración desde el carácter en que se detuvo la vez anterior y avanza hasta completar otra palabra. El proceso se repite mediante peticiones del gestor de análisis hasta que todo el texto ha sido recorrido. Para extraer las palabras se hace distinción entre cinco clases de caracteres: alfabéticos, numéricos, signos de puntuación, terminadores y otros. Algunos símbolos pueden cumplir papeles diferentes dependiendo del contexto en que se encuentren: así un punto puede servir de signo de puntuación —actúa al mismo tiempo de terminador de palabra— o como conector —clase otros— en una dirección URL, por ejemplo. Lo que el selector de palabras extrae pertenece a una de tres posibles categorías: secuencia alfabética —son las palabras propiamente dichas—, secuencia

alfanumérica —formada por letras y números, como los identificadores típicos en informática— y otras secuencias —incluyen caracteres especiales, como el punto en las direcciones URL. Además, a estas secuencias las acompañan informaciones concernientes a los signos de puntuación que hay en su entorno.

## 5.3.- Módulo de análisis morfológico.

Las secuencias de caracteres producidas por el selector de palabras se hallan etiquetadas según su categoría; de ellas sólo las secuencias alfabéticas se consideran palabras por la herramienta de reconocimiento morfológico y en consecuencia, sólo ellas son sometidas a su acción (a las otras las cataloga como no reconocidas).

La herramienta de reconocimiento morfológico es un módulo externo que trabaja tomando una palabra y dando en respuesta la lista de formas canónicas de las que podría provenir y las categorías gramaticales que le serían aplicables. Para obtener este resultado se empieza por descomponer la palabra en sus posibles pares raíz-terminación, prefijos y, en el caso de los verbos, pronombres enclíticos. La raíz pasa a un módulo de índices que determina su localización para que un módulo de accesos externos compruebe si la raíz admite la terminación, determine a qué flexión

o derivación corresponde, deduzca su forma canónica y proporcione su categoría gramatical.

## 5.4.- Módulo optimizador de búsqueda morfológica.

El análisis morfológico de los textos obtenidos constituye una parte fundamental de las aplicaciones desarrolladas; en consecuencia, la eficiente utilización del módulo que lleva a cabo dicho análisis tiene una gran influencia en el rendimiento global. El analizador morfológico consta en realidad de dos submódulos que realizan por separado la lematización de formas verbales y formas no verbales. En [SANT99b] se dice que el reconocedor de formas no verbales es capaz de lematizar un texto a un ritmo de 450 formas por segundo, según pruebas realizadas con un procesador Pentiun II a 300MHz con 128 Mb de memoria RAM; asimismo, en [SANT97c] se afirma que se reconocen alrededor de 370 formas verbales por segundo, en este caso con un Pentium a 100MHz con 16 Mb de RAM —se pronostican resultados similares en ambos lematizadores al homologar las plataformas. Como una palabra puede pertenecer a cualquiera de las dos categorías, es necesario efectuar siempre los dos procesos de reconocimiento, con lo que cabe

esperar que la velocidad media obtenida estuviese en torno a la mitad —entre 220 y 250 palabras por segundo en un Pentium II a 300 Mhz con 128 Mb RAM.

Con el objetivo de poder cuantificar las mejoras posibles en la utilización del módulo de reconocimiento morfológico, se realizó un experimento consistente en analizar 50 documentos seleccionados aleatoriamente de entre los 500 más visitados de la Biblioteca Virtual Miguel de Cervantes [WEB08] según figuraban el 16/12/2000. Estos 50 documentos se distribuyen en 605 páginas web y contienen un total de 1 815 208 palabras. Las pruebas se realizaron en un Pentium II a 300 Mhz con 128 Mb RAM, se registró el tiempo acumulado por cada 50 000 palabras analizadas más el último tramo de 15208. Los resultados se muestran en la figura 18, donde se observa que la velocidad promedio de reconocimiento alcanza las 247 palabras por segundo —cae dentro de los márgenes esperados. A esta velocidad, el análisis de los 50 documentos escogidos requiere un tiempo superior a las dos horas.

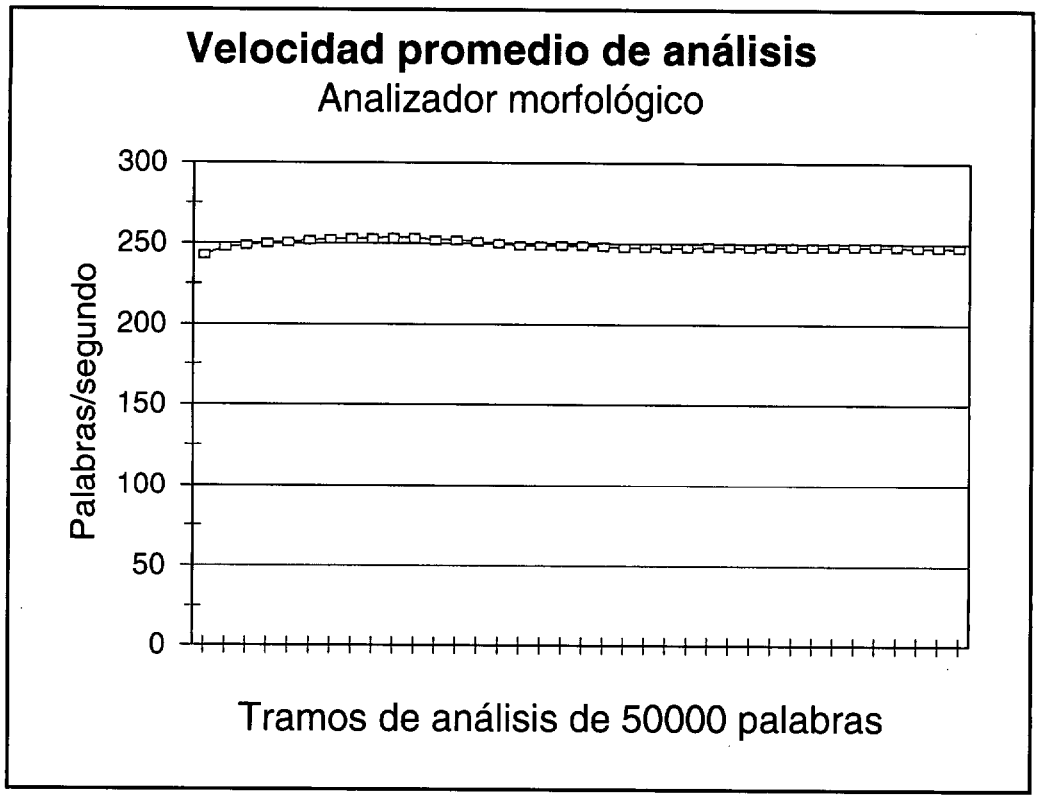



**Figura 18**

En un texto, las palabras no se distribuyen de manera uniforme, sino que habitualmente un número muy reducido de ellas se repiten mucho y un grupo no muy grande aparece una sola vez. Concretamente, en el ejemplo estudiado, de las 1 815 208 palabras sólo 94 104 son diferentes, lo que supone un promedio de repetición de 19 a 1; al estudiar la distribución de frecuencias de las palabras, se obtienen los resultados de la tabla mostrada a continuación, en la que se observa que con sólo 133 palabras (el 0.141% de las diferentes) se alcanza el 50% de las ocurrencias y que el 95% de éstas se deben al 31% de las palabras diferentes.

| Porcentaje del texto | Nº de ocurrencias de las palabras en el texto | Nº de palabras diferentes | Porcentaje sobre el total de diferentes |
|---|---|---|---|
| 5% | 101171 | 1 | 0.001% |
| 10% | 231441 | 3 | 0.003% |
| 15% | 291853 | 4 | 0.004% |
| 20% | 381169 | 6 | 0.006% |
| 25% | 470615 | 9 | 0.010% |
| 30% | 546420 | 13 | 0.014% |
| 35% | 640423 | 20 | 0.021% |
| 40% | 730037 | 35 | 0.037% |
| 45% | 818576 | 71 | 0.075% |
| 50% | 907647 | 133 | 0.141% |
| 55% | 998643 | 258 | 0.264% |
| 60% | 1089441 | 446 | 0.474% |
| 65% | 1180035 | 786 | 0.835% |
| 70% | 1270750 | 1357 | 1.442% |
| 75% | 1361414 | 2315 | 2.460% |
| 80% | 1452202 | 3996 | 4.246% |
| 85% | 1542940 | 7086 | 7.530% |
| 90% | 1633691 | 13389 | 14.228% |
| 95% | 1724450 | 29440 | 31.285% |

Aunque pudieran parecerlo, los datos de la tabla no son en absoluto espectaculares; en el estudio expuesto en [ALAM95] las 100 palabras más frecuentes ocupan el 53% del texto —bastan sólo 15 para obtener el 35%. La palabra más

frecuente "de" supone un 5,8% del total en [ALAM95] y un 5,57% en el caso presente.

Cuando una palabra aparece por segunda o sucesivas veces en el texto, se reincide en el esfuerzo de análisis que se realizó en su primera aparición. Dado que hay palabras que se repiten mucho, aparece como una alternativa interesante la posibilidad de evitar las sucesivas llamadas al reconocedor que volverían a obtener los mismos datos de la primera vez. La solución consiste en implantar algún tipo de estructura de acceso rápido en la que se almacenarían los datos resultantes del reconocimiento de cada palabra que aporte el analizador morfológico —tal estructura estaría tanto más justificada cuanto mayor sea el grado de repetición de las palabras. La arquitectura del reconocimiento se modificaría de forma que cuando se obtenga una palabra del texto, no se invoque directamente al analizador morfológico, sino que se consulte previamente la *lista de palabras reconocidas* para averiguar si se ha lematizado con anterioridad; la sobrecarga que representa la consulta de palabras que aparecen por primera vez y que de todas maneras hay que lematizar, quedará ampliamente compensada por la superior velocidad de acceso a la estructura frente al proceso de reconocimiento morfológico.

La estructura de la *lista de palabras reconocidas* es una tabla de dispersión de claves —las palabras— implementada en memoria principal y dimensionada a

199999 entradas en DAWed y 10000 en NAWeb —la diferencia se justifica por sus respectivas orientaciones. Se utiliza una función de dispersión de doble rotación binaria con listas encadenadas separadas para la resolución de colisiones —la longitud promedio inferior a1,3 nodos por lista es debida tanto a la bondad de la función de dispersión como a la holgura de la tabla.

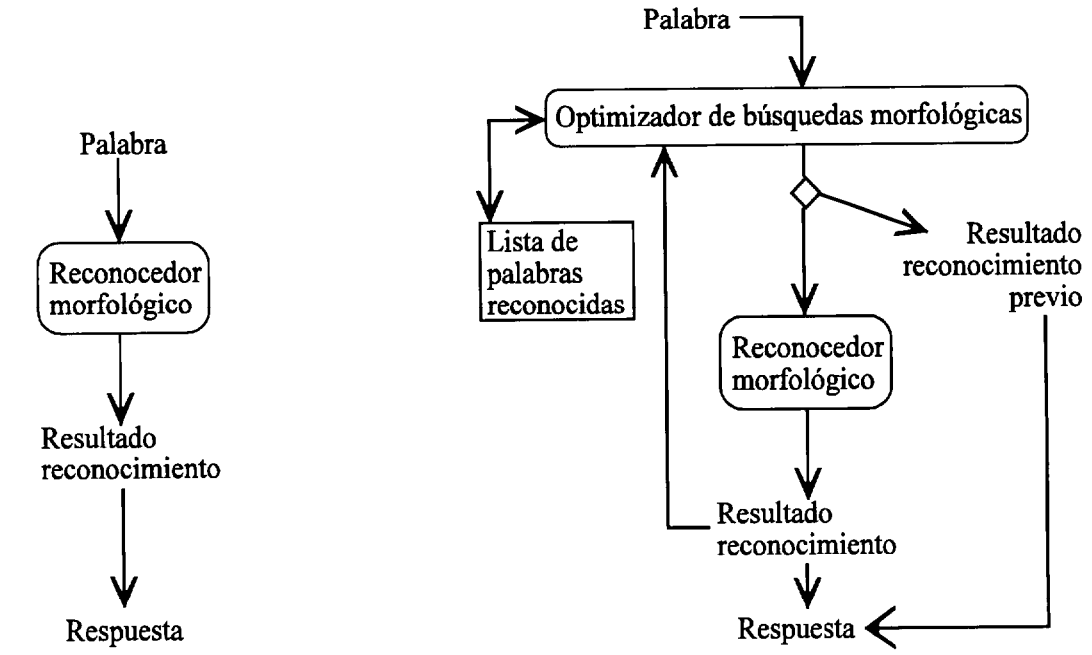



**Figura 19** Proceso de reconocimiento sin y con optimizador

La ganancia queda de manifiesto en la figura 20, donde se observa que la velocidad promedio de reconocimiento llega a experimentar una mejora superior al

1 800% con los datos del experimento. Al utilizar la *lista de palabras reconocidas*, el claro perfil creciente de la curva que representa la velocidad promedio acumulada se corresponde con la disminución de las llamadas al reconocedor morfológico a medida que se amplía la lista; la pendiente decreciente se explica porque también disminuye la probabilidad de que un tramo aporte palabras nuevas cuyo reconocimiento pueda ser aprovechado en los siguientes tramos.

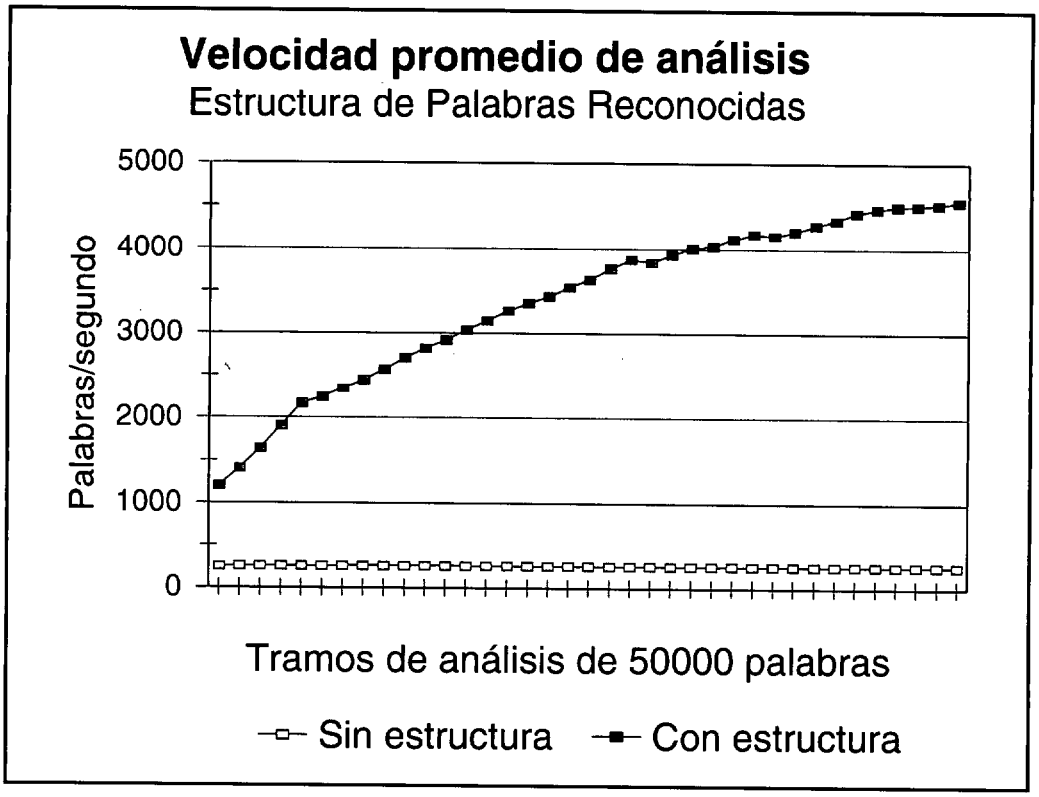



**Figura 20** Velocidad promedio

La estrecha relación que existe entre el número de llamadas al reconocedor —aparición de palabras no encontradas con anterioridad— que hay que realizar en cada tramo y las variaciones de velocidad queda claramente expresada en la figura 21, donde se observa que cada incremento de la velocidad viene provocado fundamentalmente por un descenso equivalente en el número de llamadas. Obsérvese que la línea que representa las sucesivas velocidades acumula una ganancia que da lugar al carácter creciente del promedio; por el contrario, la línea que representa la aparición de palabras nuevas tiende decididamente a bajar.

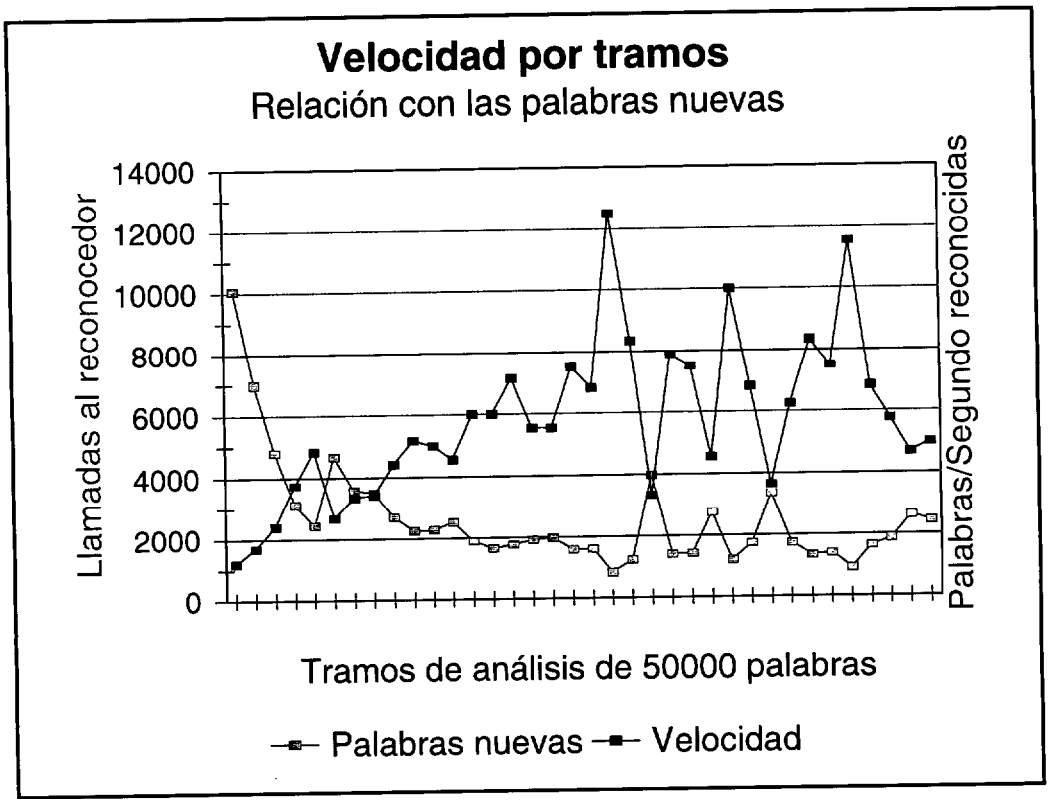



**Figura 21** Velocidad por tramos

La situación ideal sería no tener que hacer ninguna llamada al reconocedor y que todas las palabras estuviesen precargadas en la *lista de palabras reconocidas*. Esa situación es inviable, ya que el desconocimiento de qué palabras aparecen en un texto haría necesario tener todas las posibles y el gigantismo de la estructura de datos resultante no sólo supondría problemas en cuanto a espacio de almacenamiento, sino que, muy probablemente, produciría una ralentización de los accesos con peores resultados que el reconocedor a solas; y la lista siempre sería incompleta, pues una de las cosas que caracterizan al lenguaje natural es la creatividad. Dado que un porcentaje muy alto de las palabras de un texto cubre un número muy reducido de palabras diferentes y que un porcentaje significativo de éstas lo constituyen palabras con valor funcional de uso general en muchos contextos diferentes, la posibilidad de precargar la *lista de palabras reconocidas* —un pequeño número de palabras con alta probabilidad de aparición en cualquier texto— supondría una mejora significativa en el rendimiento del reconocimiento aunque no se añadieran con posterioridad las palabras no cargadas que se analizaran. Se han realizado tres pruebas: 1) precargar la *lista de palabras reconocidas* con las 250 palabras más frecuentes en el corpus de prueba y analizar sin cargar las palabras nuevas, 2) repetir la operación, pero utilizando las 62 palabras más frecuentes en el estudio de [ALAM95] —se ha elegido este número porque el autor hace referencia al mismo como aquél a partir del cual empiezan a aparecer palabras no funcionales dentro de las más frecuentes de su

estudio— y 3) realizar la misma precarga que en la primera prueba y permitir incorporar a la lista las nuevas palabras reconocidas. La figura 22 muestra los resultados de las dos primeras pruebas en relación con la velocidad del reconocedor morfológico. La línea 0 representa la velocidad promedio de reconocimiento

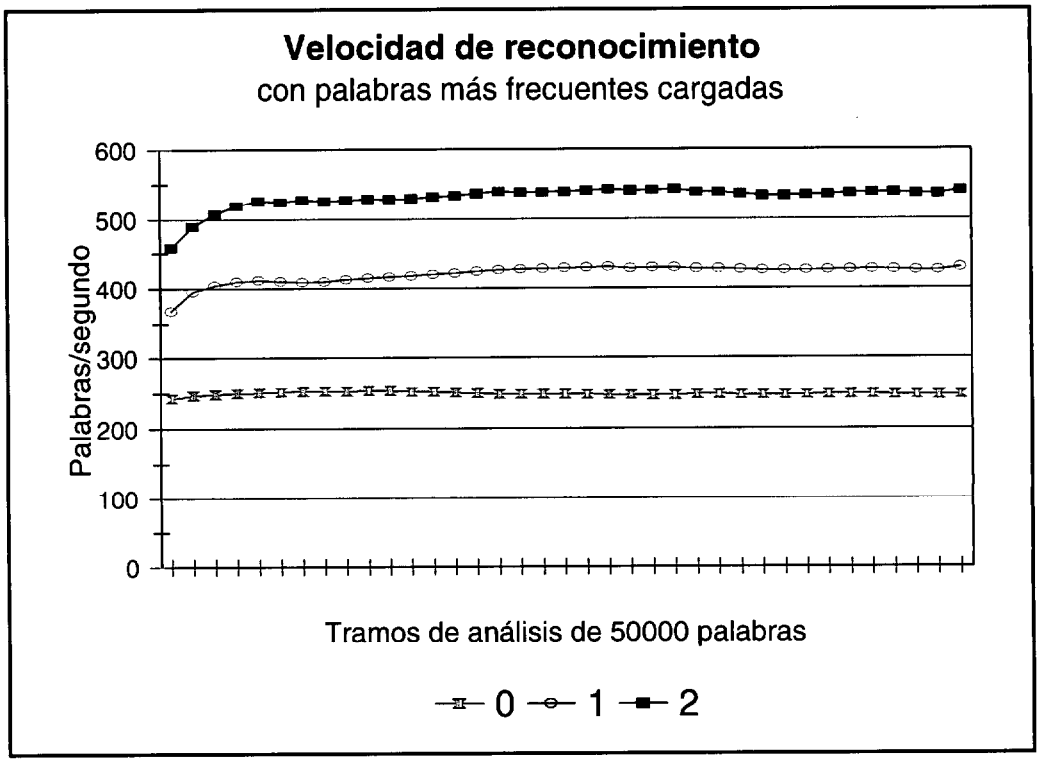



**Figura 22** Velocidad de reconocimiento con precarga

morfológico sin ninguna lista, la línea 1 muestra la velocidad cuando se empieza con las 250 más frecuentes de este conjunto de datos y la línea 2 representa la velocidad cuando se tiene cargado el reconocimiento de las 62 palabras más frecuentes de [ALAM95]. Claramente la velocidad promedio en el caso 1 dobla a la velocidad

inicial —línea 0—, lo que está en consonancia con que las 250 palabras supongan más del 55% del total de las ocurrencias; cuando se utilizan palabras de [ALAM95], la velocidad se multiplica por un factor de 1,76, lo que se explica porque ellas sólo representan el 43,8% del total de ocurrencias en el corpus estudiado.

No se ha incluido ninguna gráfica para la tercera prueba —en la que se permite añadir las palabras encontradas—; la razón es que la gráfica resulta muy similar a la de la prueba inicial —sin precarga—: con un cambio de escala podría apreciarse que cuando hay precarga, la velocidad con que arranca es ligeramente superior dado que las 250 palabras que no hay que reconocer empiezan a aprovecharse desde el principio al ser muy frecuentes; pero la realidad es que mantener almacenado el reconocimiento de esas 250 palabras tiene escasa relevancia en el comportamiento global, ya que sólo evita 250 llamadas al reconocedor de un total de 94 104 que hay que realizar —menos del 0,3%. La conclusión lógica es que precargar el reconocimiento de un conjunto selecto de palabras resulta irrelevante para DAWeb —usa el módulo optimizador de búsquedas morfológicas a pleno rendimiento—; sin embargo NAWeb —que en ninguna utilización llegará a manejar un volumen de documentos que le permita siquiera acercarse a los resultados de DAWeb— puede beneficiarse de un incremento de la velocidad base de análisis que con las palabras adecuadas puede alcanzar el 100% —seleccionar a priori cuáles son

las palabras más adecuadas constituye un tema interesante, aunque escapa a los objetivos de esta tesis.

Diversas pruebas realizadas alterando el orden entre los documentos a analizar dieron los resultados esperados: curvas de evolución de la velocidad promedio de análisis en esencia similares a la presentada en la figura 20 —el orden entre los documentos no tiene influencia relevante sobre la eficiencia. Es un resultado lógico, dado que los documentos comparten el núcleo de palabras más frecuentes —suponen una importante aportación a la evolución de la curva— y se diferencian en un pequeño número de palabras específicas que siempre hay que reconocer. La única manera de obtener variaciones sustanciales en el perfil de la curva aunque no en el valor final alcanzado —técnicamente estable para un conjunto de palabras dado— es mediante distribuciones especiales de las palabras en el texto —por ejemplo, que aparezcan en orden decreciente de frecuencia—, pero este tipo de distribuciones no son propias de textos "naturales" con sentido.

El hecho de que el orden en que se analizan los documentos no modifique sensiblemente la eficiencia del optimizador constituye un resultado digno de tener en cuenta, ya que el propio diseño de DAWeb implica que los documentos se recuperan y analizan en un orden casi aleatorio —determinado por la respuesta de la red. También conviene resaltar tal ventaja en el diseño de NAWeb dado que, aunque

se supone que trabaja con pocos documentos —sobre todo en comparación con DAWeb—, también se obtienen sin un orden predecible —determinado por la iniciativa y curiosidad del usuario y no por la red.

# 6.- Interfaz de DAWeb.

La interfaz de DAWeb, figura 23, consta de un menú y un área de trabajo dividida en tres secciones: 1) *Configuración*, 2) *Proceso* y 3) *Programación*. El menú presenta sólo dos opciones: 1) *Proyecto* permite crear un proyecto nuevo, abrir un proyecto previamente almacenado, guardar un proyecto recién creado, guardar un proyecto con un nombre distinto del que tenía y salir de la aplicación; 2) *Acerca de* visualiza información acerca de DAWeb. DAWeb trabaja con el concepto de proyecto definible como la especificación de un conjunto de opciones para realizar el análisis de un determinado grupo de documentos web de una determinada manera; tal especificación recibe un nombre —nombre del proyecto— y se puede almacenar para ser utilizada en cualquier momento —incluye la posibilidad de activarse automáticamente en tiempos preprogramados.

La sección de configuración es donde el usuario de DAWeb define las características del proyecto en cuanto a descarga y análisis que plantea. De arriba abajo, el primer elemento que se encuentra sirve para atribuir un título a un proyecto de nueva creación o visualizar —y cambiar si se desea— el título de un proyecto —todo proyecto debe tener un título.

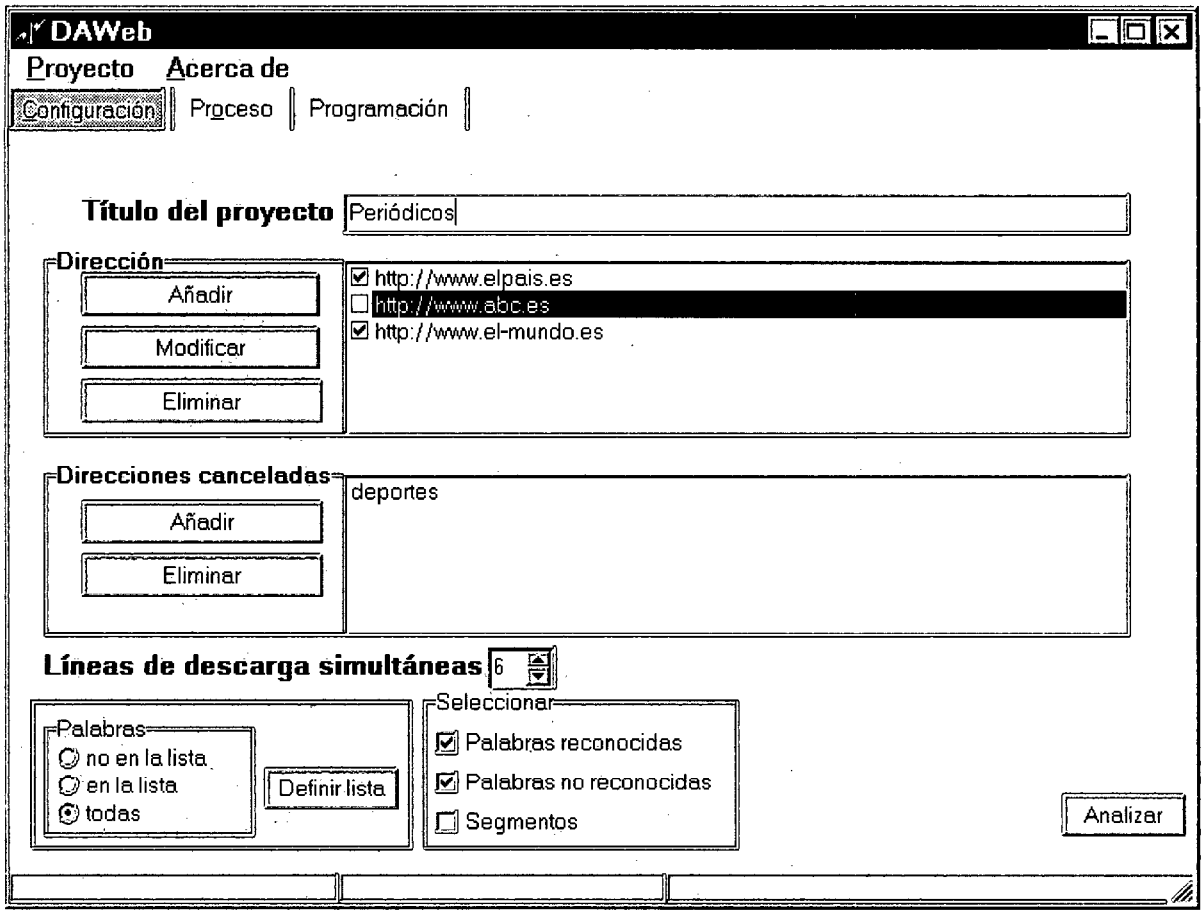



**Figura 23** Interfaz de DAWeb con la pestaña de clasificación activa



Por debajo del título se muestra una zona destinada a configurar las direcciones cuyo contenido se quiere analizar; consta de cuatro elementos: la lista de direcciones introducida por el usuario que permite seleccionar las que se quieren utilizar en cada momento —no es necesario navegar siempre por todas las direcciones incluidas en un proyecto— y tres botones para añadir, modificar o eliminar una dirección de la lista. Las opciones *Añadir* y *Modificar* permiten, como

muestra la figura 24, indicar o corregir una dirección y seleccionar un fichero para almacenar los datos recogidos —se pueden elegir ficheros diferentes para direcciones diferentes. La dirección especificada no designa únicamente una página web, sino que sirve como punto de partida para analizar el conjunto de páginas del mismo dominio accesibles por los hiperenlaces que contenga —con las restricciones que se impongan en la zona de direcciones canceladas. Tanto el botón *Modificar* como el *Eliminar* requieren que haya una dirección seleccionada en la lista de direcciones.

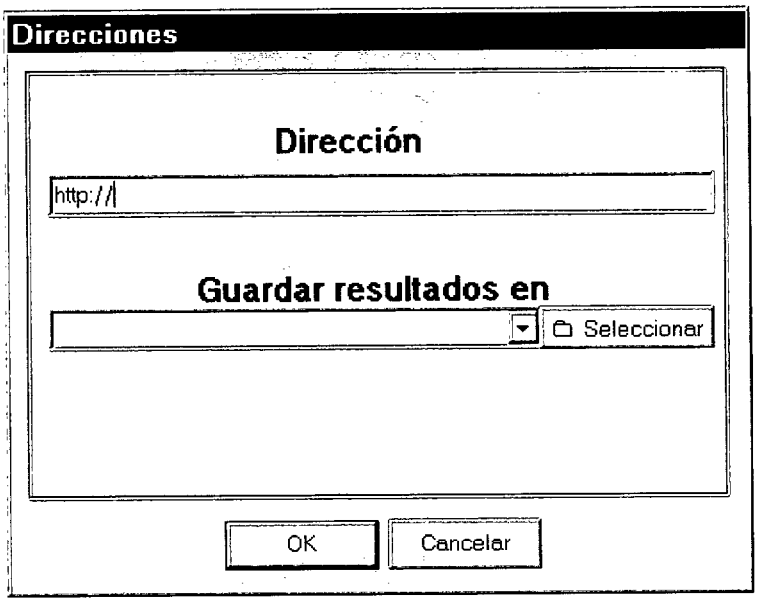



**Figura 24** Diálogo de *Añadir* y *Modificar*

La zona de direcciones canceladas se utiliza para designar una lista de direcciones que no se quieren analizar, aunque formen parte del dominio de alguna

de las direcciones de partida activas. Se proporciona un botón *Añadir* —lanza un cuadro de diálogo con tal propósito, figura 25— y otro *Eliminar* que descarta la entrada seleccionada en la lista de direcciones canceladas. Se pueden indicar tanto direcciones completas como cadenas de caracteres que puedan formar parte de una dirección; por ejemplo, puede que no se desee analizar la sección de deportes de un periódico, la cual está distribuida en varias páginas con direcciones diferentes pero todas contienen la palabra "deportes": en lugar de especificar todas las direcciones diferentes de las páginas de deportes, se puede indicar "deportes" —no se accede a ninguna dirección que contenga esta secuencia de caracteres.

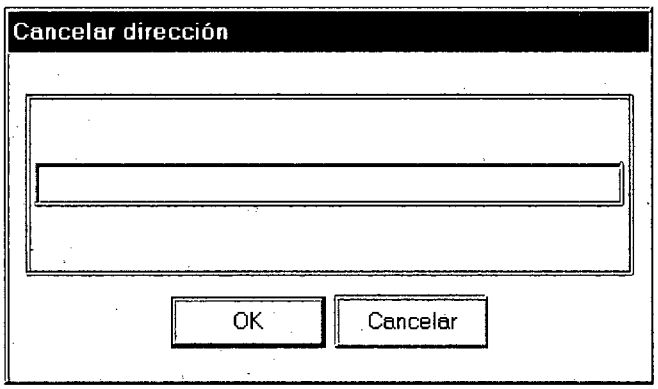



**Figura 25** Diálogo para añadir a la zona de direcciones canceladas

En la opción *Líneas de descarga simultáneas*, se configura el número de hilos de ejecución que se activarán con módulos recuperadores en paralelo; inicialmente

está puesta a 5 y puede bajarse hasta 1 —produce una descarga lineal, aunque lenta— o subirse hasta 10 —el máximo razonable según los experimentos realizados.

La parte inferior de la sección de configuración se dedica a definir cómo se llevará a cabo el análisis de los documentos accedidos según las direcciones fijadas en las secciones anteriores. Se puede definir una lista de "palabras vacías" o una lista de "palabras significativas"; en el primer caso —*no en la lista*— las palabras no entran en el análisis y en el segundo —*en la lista*— el análisis se restringe a esas palabras. Ambas listas se configuran en un cuadro de diálogo —se lanza en el botón *Definir lista*—, figura 26, que permite editarlas, guardarlas para su uso en otros proyectos y recuperarlas de un fichero.

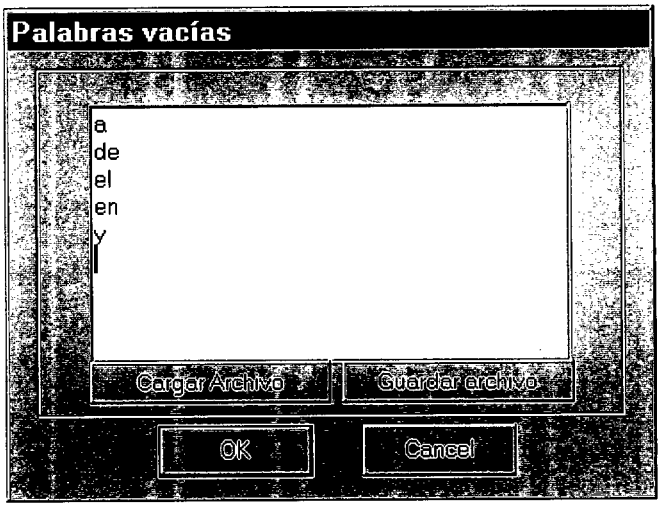



**Figura 26** Diálogo *de Definir lista*

Se puede configurar la descarga de información que se efectúa: las palabras reconocidas, las no reconocidas —suficiente si sólo se están buscando neologismos— y las secuencias frecuentes de palabras (*Segmentos*) —útil en estudios más generales.

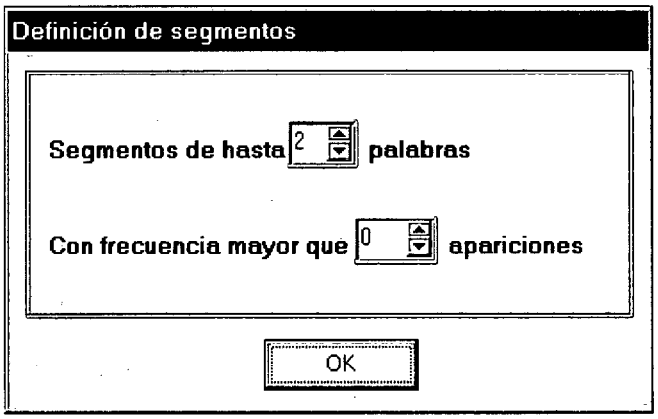



**Figura 27** Diálogo *Definición de segmentos*

En la esquina inferior derecha de la pestaña de configuración se encuentra el botón *Analizar* que pone en marcha el análisis configurado y cambia la presentación de la aplicación —activa la sección de proceso mientras dure.

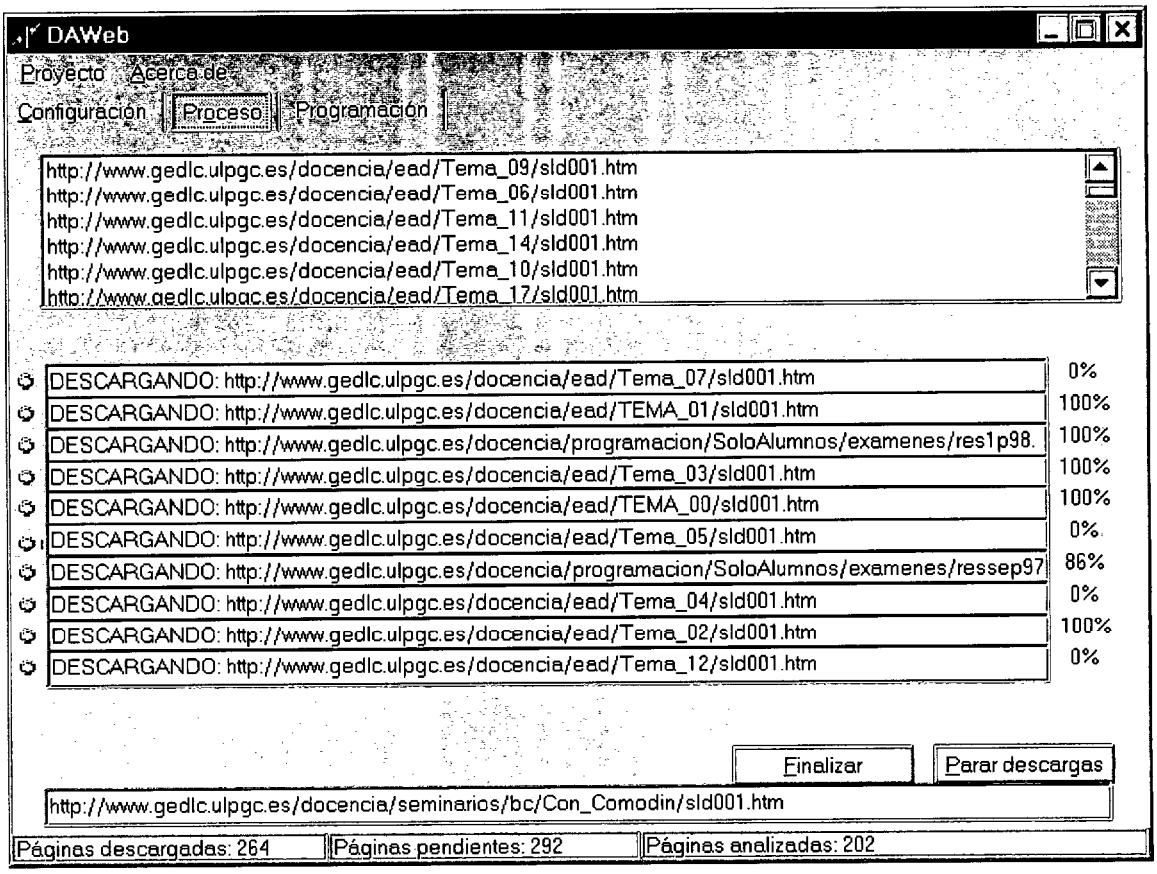



**Figura 28** Sección de proceso durante la ejecución de un proyecto



La sección de proceso se activa cuando se pone en marcha la ejecución de un proyecto —puede ser accedida siempre, pero no da información si no se está ejecutando un proyecto. La primera área que muestra contiene la *lista de direcciones pendientes*, se modifica a medida que se toman direcciones para intentar su acceso y que se incluyen las que se encuentran en los documentos accedidos —dentro de los límites establecidos en la configuración del proyecto. Debajo del área de direcciones aparecen tantas líneas de información como módulos recuperadores se activen; cada

línea muestra el estado del recuperador —preparado, descargando, redireccionado, terminado, falló la dirección, falló la recuperación, no existe el host y se necesita autentificación—, la dirección a la que está accediendo o intentando acceder y el porcentaje de la respuesta que se ha obtenido. Una línea de información situada más abajo informa acerca del documento que está en curso de análisis —de entre los que ya se han recuperado. En la parte inferior, la línea de estado de la aplicación muestra información general acerca de cuántas direcciones se han accedido, cuántas están pendientes de ser accedidas y cuántas han sido analizadas —el primer y el tercer número son siempre crecientes, mientras que el segundo fluctúa en función de las direcciones que aporte cada documento accedido y del ritmo de trabajo de los recuperadores. Los dos de acción que ofrece esta sección tiene por objeto detener la ejecución del proyecto en marcha si se considera necesario; esta parada no es en ningún caso inmediata, ya que se han de abortar las conexiones en curso, terminar los análisis iniciados y vaciar las listas de direcciones y documentos pendientes. La diferencia entre *Parar descargas* y *Finalizar* es que el primero analiza las páginas que ya estén descargadas antes de terminar y el segundo no lo hace.

La sección de programación, figura 29, tiene por objeto la selección de una lista de proyectos a los que se les asigna una hora para su ejecución diferida. No hay problema por asignar la misma hora a varios proyectos, pero estos no se ejecutan en

paralelo. Se ejecuta el primer proyecto que alcance o sobrepase su hora de ejecución, y el resto queda pendiente hasta que éste haya terminado; si mientras tanto se alcanza la hora de ejecución de algún otro, aquel se iniciará en cuanto el que está en ejecución acabe.

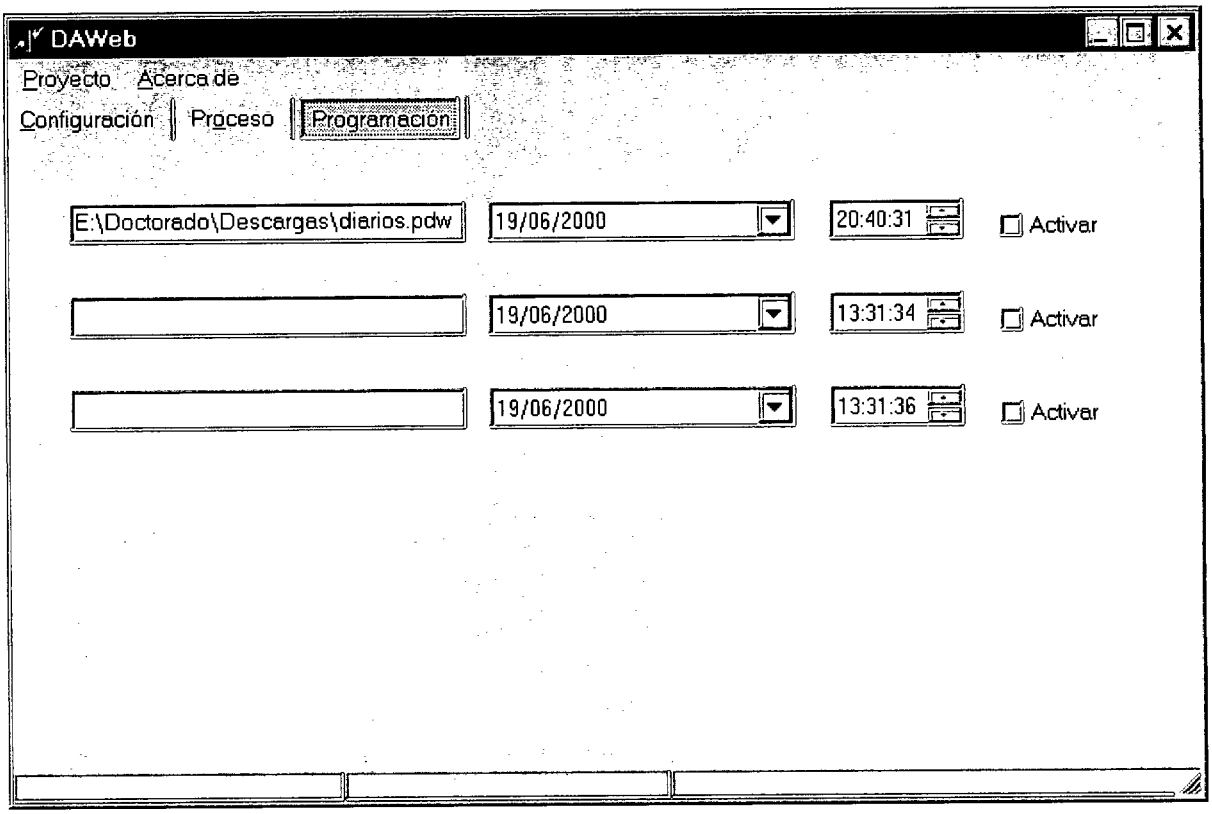



**Figura 29** Sección de programación

# 7.- Interfaz de NAWeb.

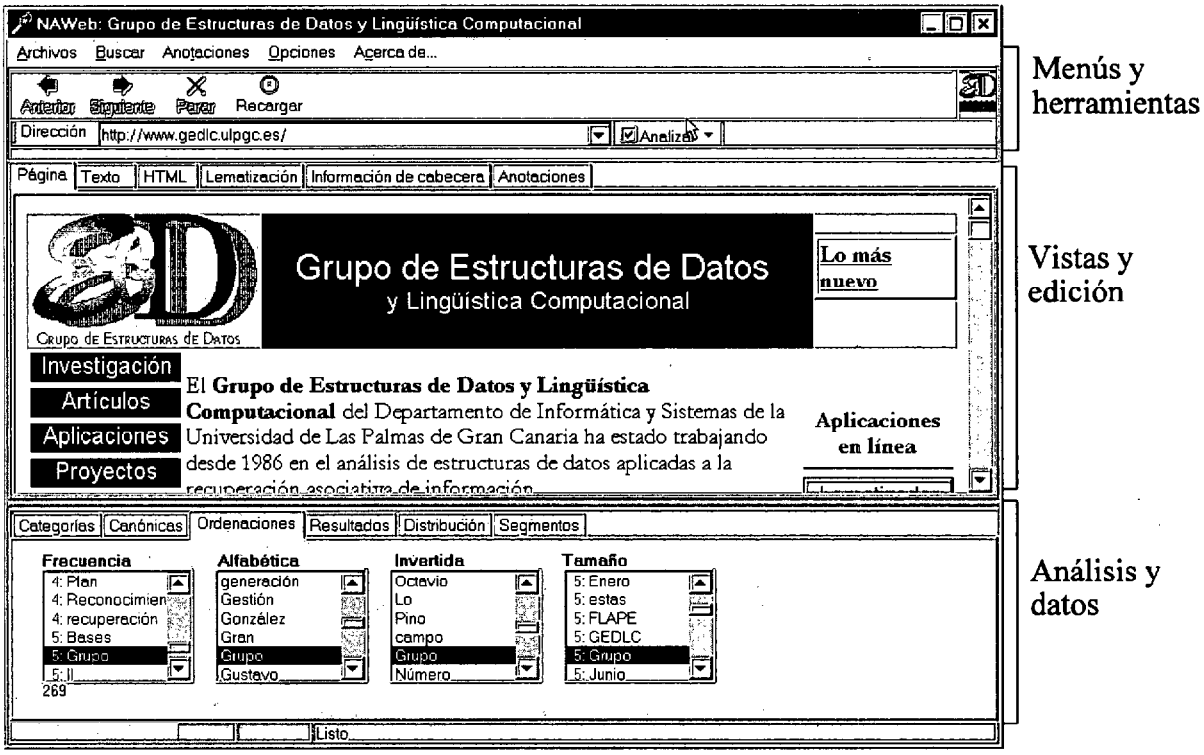



**Figura 30** Aspecto general de NAWeb

En NAWeb se aprecia como característica más destacada la división de la ventana de la aplicación en tres franjas horizontales de arriba abajo: 1) *zona de menús y barras de herramientas*,2) *zona de vistas y edición* y 3) *zona de análisis y datos*. Los distintos elementos de la interfaz se corresponden casi directamente con la arquitectura interna de la aplicación, tal paralelismo es lógico, porque dado su

carácter interactivo requiere una intervención atenta del usuario, a la que debe contribuir de forma prioritaria la interfaz.

## 7.1.- Zona de menús y barras de herramientas.

La zona de menús y barras de herramientas muestra: 1) el menú principal de la aplicación y sus submenús y 2) dos barras con botones típicos de navegación y un área para introducir la dirección a la que se quiere navegar. El conjunto formado por los menús, barras de herramientas y el contenido de la primera pestaña de la zona de vistas y edición conforman el aspecto típico de un navegador de Internet. La posibilidad de activar o desactivar el análisis —*Analizar*— representa una novedad del navegador NAWeb.

El menú principal de NAWeb ofrece cinco opciones: 1) *Archivos*, 2) *Buscar*, 3) *Anotaciones*, 4) *Opciones* y 5) *Acerca de*. El submenú *Archivos*, figura 31, ofrece cinco posibilidades: 1) *Abrir* permite navegar a un fichero local seleccionado mediante un diálogo por el que se accede a la estructura del sistema de información de la máquina personal o iniciar la ejecución de otra instancia de NAWeb para

realizar navegaciones paralelas sobre diferentes documentos, 2) *Guardar* almacena localmente la página visualizada —permite diferir su estudio a una ocasión posterior sin necesidad de reconexión a la red—, 3) *Guardar como* se usa para almacenar una copia con un nombre distinto al guardado previamente, 4) *Imprimir* proporciona una versión en papel del documento y 5) *Salir* termina la ejecución de NAWeb —de la instancia en que se utilice, si hubiera varias abiertas.

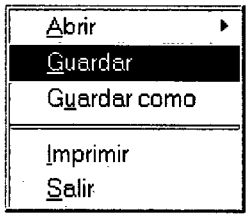



**Figura 31** Submenú *Archivos*

El submenú *Búsquedas*, figura 32, se encuentra dividido en cuatro apartados. El primero ofrece dos opciones para la localización de ocurrencias exactas de palabras o de formas canónicas —se buscan las palabras que corresponden a una forma canónica determinada. El segundo presenta tres opciones de búsqueda de elementos complejos tales como colocaciones simples, perífrasis o regímenes preposicionales —en todos los casos se puede configurar mediante un diálogo la frecuencia mínima con que deben aparecer las ocurrencias que se desea considerar.

El tercer apartado ofrece dos opciones para localizar las palabras más parecidas a una dada, según el tipo de distancia a aplicar —subsecuencia común más larga no contigua (SCML) o distancia de Levenshtein (DL), en [COR90], [GUS97], [WEB06] y [WEB07] se detallan sus definiciones y algoritmos para su cálculo. El cuarto apartado consta de una única opción conmutable *Marcar/Desmarcar* que activa el efecto de las anteriores sobre la vista del texto: cada una de las opciones de búsqueda de los tres primeros apartados actúa resaltando con un color las ocurrencias que encuentra en un proceso acumulativo —permite al usuario estudiar la distribución y correlación de las mismas—; como la vista del texto también se modifica por los mecanismos de sincronización cuando el usuario se mueve por las otras zonas de información, la opción *Marcar* bloquea la sincronización para que no interfiera con el resultado de las búsquedas —se activa automáticamente siempre que se inicia una búsqueda y el usuario puede activarla o desactivarla en cualquier momento.



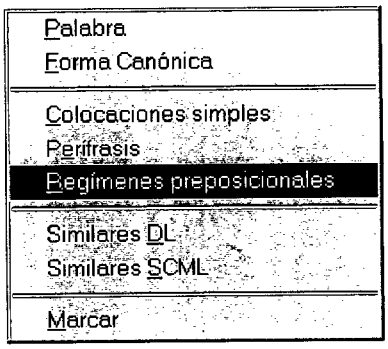



**Figura 32** Submenú *Búsquedas*

El submenú *Anotaciones*, figura 33, está pensado para extraer información de contexto acerca de los elementos seleccionados en la vista del texto y traspasarla a la vista de anotaciones donde puede ser modificada con apuntes hechos por el usuario y almacenada en un fichero. Ofrece tres opciones: 1) *Anotar contexto*, añade el contexto —región de 80 caracteres en torno a la palabra— a la vista de anotaciones, 2) *Limpiar*, vacía la vista de anotaciones y 3) *Guardar*, ofrece un cuadro de diálogo con el que se puede seleccionar un nombre de fichero y una ubicación para almacenar la información mostrada en la vista de anotaciones.

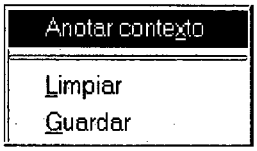



**Figura 33** Submenú *Anotaciones*

El submenú *Opciones* permite configurar la transferencia de documentos, especifica la carga de imágenes, vídeos, animaciones y sonidos; son elementos que tienen un importante coste de transmisión y suelen carecer de interés cuando se trata de hacer un estudio lingüístico del texto contenido en un documento —no de su diseño gráfico—; el descartar su carga acelera el proceso de transferencia, con lo que se gana un tiempo valioso que puede ser aprovechado para el estudio propiamente dicho.

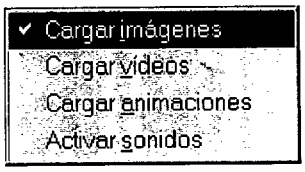



**Figura 34** Submenú *Opciones*

La primera barra de herramientas, figura 35, está formada por cuatro botones que se activan y se desactivan según las circunstancias de la navegación y representan los cinco movimientos básicos de una navegación por Internet: 1) *Anterior*, retrocede a la página anterior en el orden de descarga, 2) *Siguiente*, avanza a la página siguiente en el orden de descarga, 3) *Parar*, detiene la descarga de una página y 4) *Recargar*, vuelve a cargar una página.

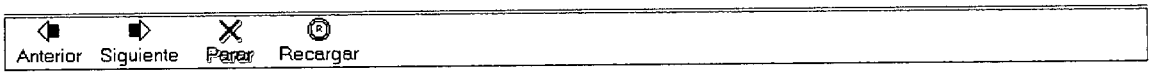



**Figura 35** Primera barra de herramientas: barra de navegación

La segunda barra, figura 36, contiene: 1) un campo de entrada para teclear una URL que al desplegarse permite renavegar a una página accedida con anterioridad y 2) un cuadro de selección, *Analizar*, para activar o desactivar el análisis de los documentos accedidos. El proceso desencadenado por *Analizar* está gobernado por un submenú de opciones conmutables, figura 37, que permite elegir los análisis que

se realizarán según la complejidad y dependencia entre ellos: siempre se realizan los estudios métricos, se pueden seleccionar los morfológicos —activados por defecto—, los de distribución y las ordenaciones de las palabras por parecido a una dada.


**Figura 36** segunda barra de herramientas: barra de dirección y análisis

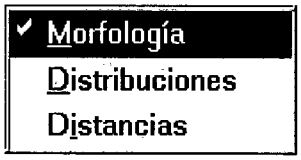



**Figura 37** Submenú de análisis



## 7.2.- Zona de vistas y edición.

La zona de vistas y edición se halla organizada por pestañas, figura 38, cada una de las cuales presenta una forma distinta de ver el documento accedido. La pestaña inicialmente activa muestra la visión "web" del documento tal como se

presenta en cualquier navegador —con colores, gráficas, navegación a través de los hiperenlaces que contiene, etc.

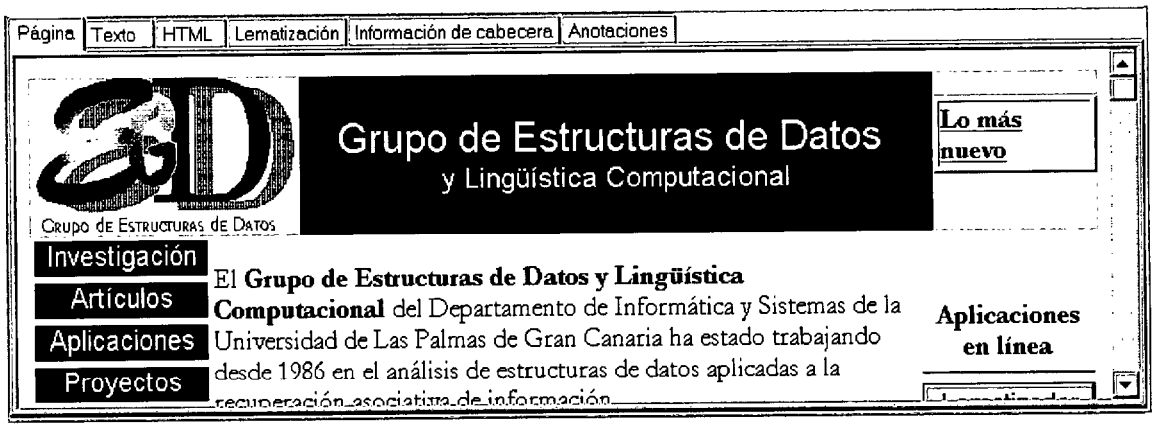



**Figura 38** Zona de vistas y edición

La segunda pestaña muestra el contenido del texto del documento —la vista del texto—, desprovisto de sus aspectos gráficos, figura 39. El texto se obtiene extrayendo el contenido del componente TWebBrowser como se describe en la sección 5.1. El texto que aparece en esta pestaña constituye el punto de partida para todos los análisis y búsquedas: la información que se muestra en las otras partes de la interfaz se mantiene sincronizada con este texto.

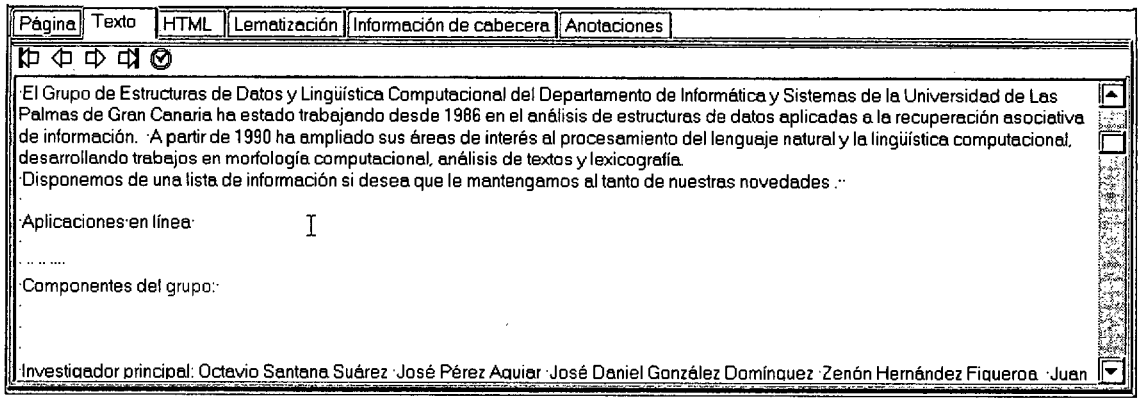



**Figura 39** Vista del texto

En la parte superior se incluye una barra de herramientas local con cinco botones; los cuatro primeros sirven para desplazarse secuencialmente adelante —siguiente y última— y atrás —anterior y primera— por la lista de ocurrencias destacadas en el texto; el quinto botón tiene el mismo efecto que la opción *Marcar/Desmarcar* del submenú *Búsquedas*.

Un menú flotante permite copiar al "clipboard" la parte seleccionada en la vista del texto y seleccionar todo el texto contenido en esta vista. Cuando esté activa la opción *Marcar*, se puede seleccionar un trozo de texto con el ratón o con las teclas del cursor; cuando esté activa *Desmarcar*, cualquier acción del ratón o del teclado pone en marcha los mecanismos de sincronización que impiden mantener la selección.

La tercera pestaña presenta el contenido del documento en lenguaje HTML —vista HTML—; cumple con la función básica de ofrecer información complementaria en cuanto a la estructura del documento, figura 40. Responde al mismo menú flotante que la vista del texto, pero sin restricciones.

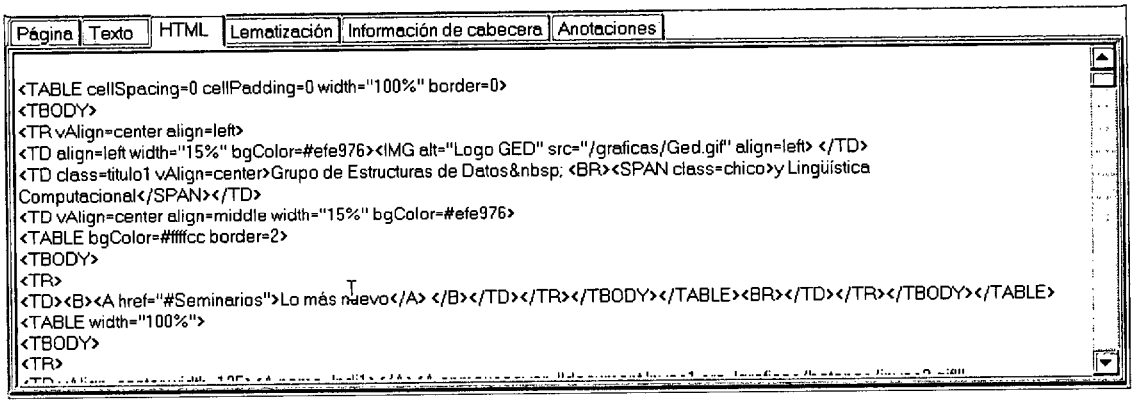



**Figura 40** Vista HTML

La cuarta pestaña muestra la vista lematizada del texto —vista de la lematización—, figura 41. Como resultado del análisis del contenido de la vista del texto, el primer recuadro por la izquierda presenta una lista de todas las palabras en el orden en que aparecen en el texto —un código de colores[3] identifica el grado de

3 Negro para las no ambiguas, rojo para las ambiguas o no reconocidas, azul para las desambiguadas automáticamente, azul marino para las desambiguadas manualmente y oliva para las no reconocidas a las que se las ha asignado una categoría..

ambigüedad de cada palabra—; el segundo recuadro muestra el resultado de la lematización de la palabra seleccionada en el primero: formas canónicas y categorías gramáticales. Si la opción de análisis morfológico está desactivada, el segundo recuadro mostrará un mensaje indicando que las palabras no han sido lematizadas —desaparecerá en cuanto se active la lematización en el submenú de análisis.

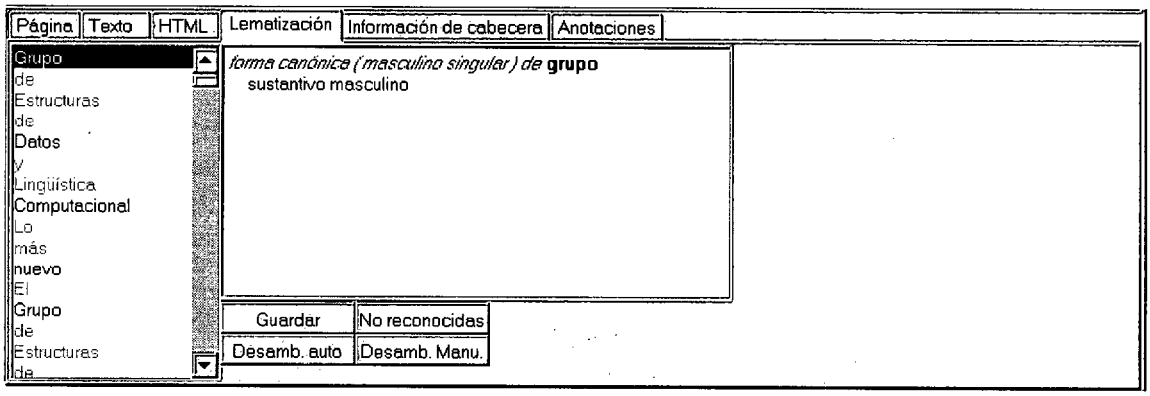



**Figura 41** Vista de la lematización



Debajo del segundo recuadro aparece un grupo de botones —inactivos si no se ha realizado análisis morfológico— que sirven para: 1) guardar el resultado de la lematización en un fichero con formato HTML que puede consultarse cuando se desee, 2) asignar manualmente categorías a las palabras no reconocidas, 3) desambiguar automáticamente las palabras con reconocimiento múltiple y 4) desambiguar manualmente. Las dos opciones manuales —2 y 4— operan por

medio de un cuadro de interacción con el usuario que se despliega en el área vacía de la derecha y ofrece una lista de las posibles interpretaciones para la palabra no reconocida o ambigua, figura 42; el usuario puede elegir una de las opciones para aplicarla a la aparición actual de la palabra; también cabe la posibilidad de dejar la ambigüedad —y sus siguientes ocurrencias— sin resolver por el momento.

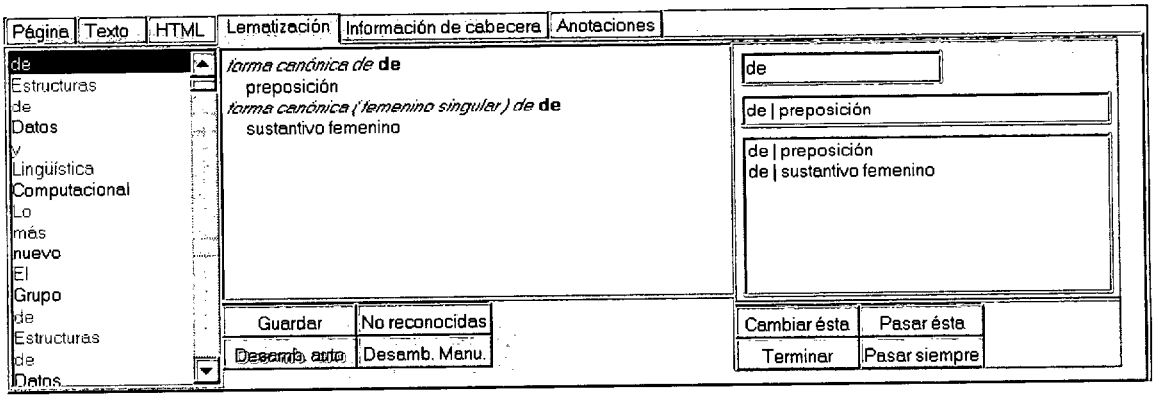



**Figura 42** Vista de la lematización, despliegue operación manual

La sección Información de cabecera ofrece información técnica sobre el servidor accedido —sin relevancia lingüística. La sección de Anotaciones es un área de edición en la que el usuario puede actuar libremente; también las anotaciones de contexto solicitadas mediante el submenú de la figura 33.

## 7.3.- Zona de análisis y datos.

La zona de análisis y datos se emplea para organizar según distintos criterios los resultados obtenidos en el análisis del texto, figura 43; se compone de seis pestañas: 1) *Categorías*, 2) *Canónicas*, 3) *Ordenaciones*, 4) *Resultados*, 5) *Distribución* y 6) *Segmentos*.

La pestaña *Categorías* muestra las palabras agrupadas según su categoría gramatical, figura 43. Aparecen seis cajas con listas de verbos, sustantivos, adjetivos, adverbios, otras formas, palabras no reconocidas y secuencias no alfabéticas. La lista de secuencias no alfabéticas incluye tanto números y fechas como cualquier otra secuencia formada por letras y caracteres "extraños" —en la web del diario "El País" aparece "ciberp@is". Una misma palabra puede aparecer en más de una caja —salvo las no reconocidas o secuencias no alfabéticas— bien porque aparezca en el texto cumpliendo funciones diferentes o porque sus apariciones no hayan sido desambiguadas.

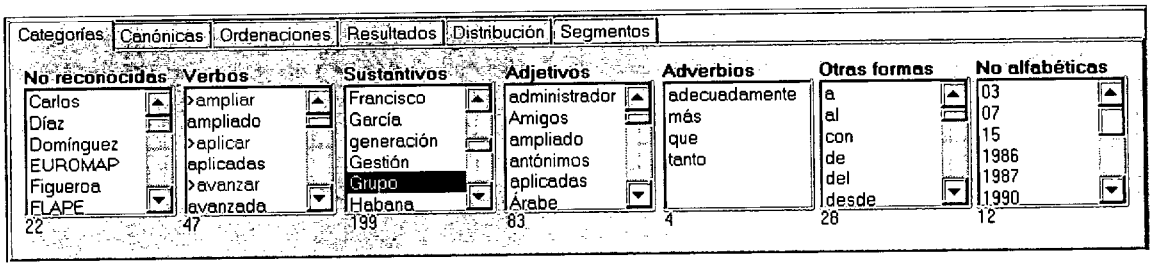



**Figura 43** Zona de análisis y datos. Vista por categorías

La pestaña *Canónicas*, figura 44, muestra las formas canónicas correspondientes a las palabras del texto en cuatro listas diferentes ordenadas según: 1) relación *Alfabética*, 2) *Frecuencia*, 3) relación *Alfabética inversa* y 4) *Tamaño*.

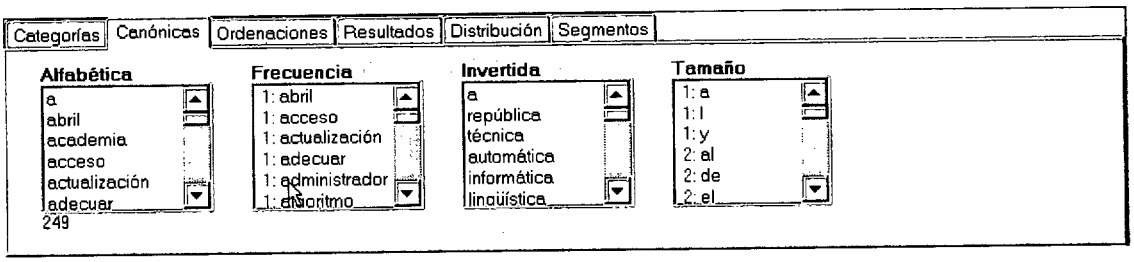



**Figura 44** Vista por formas canónicas

La pestaña *Ordenaciones*, figura 45, muestra cuatro listas de las palabras del texto, ordenadas según su *Frecuencia*, relación *Alfabética*, relación *Alfabética inversa* y *Tamaño*; una quinta lista aparece cuando se activa la opción *Distancias* del

submenú vinculado a la opción *Analizar* de la barra de herramientas, muestra las palabras ordenadas de menor a mayor distancia a la que se halle seleccionada en las otras listas —se puede seleccionar el tipo de distancia entre la distancia de Levenshtein (*DL*) y la que se obtiene en función del cáculo de la subsecuencia común más larga o contigua (*SCML*).

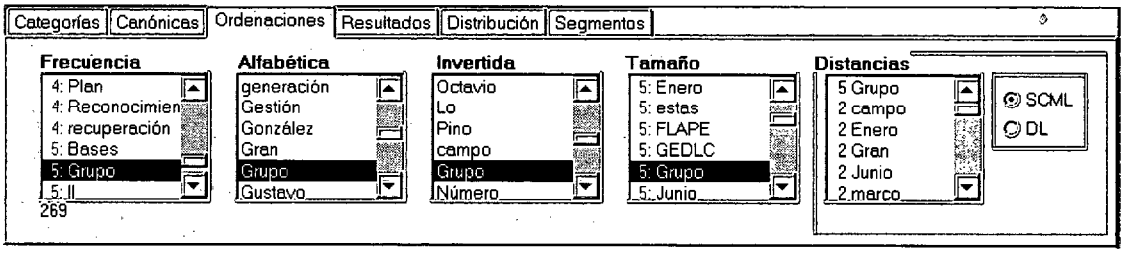



**Figura 45** Vista por ordenaciones

La pestaña *Resultados*, figura 46, muestra información de carácter general sobre el número de palabras del documento, cuántas son distintas y cómo se distribuyen numéricamente por categorías gramaticales —estos números también aparecen bajo cada una de las listas de la pestaña *Categorías*. A la derecha se representa un perfil gráfico que ilustra el ritmo de aparición de nuevas palabras en el documento.

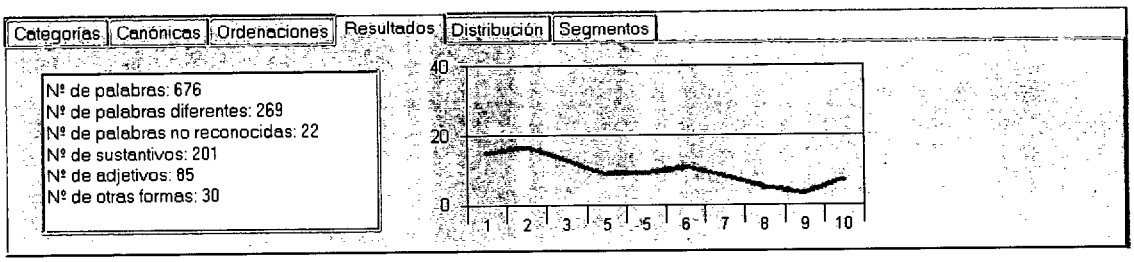
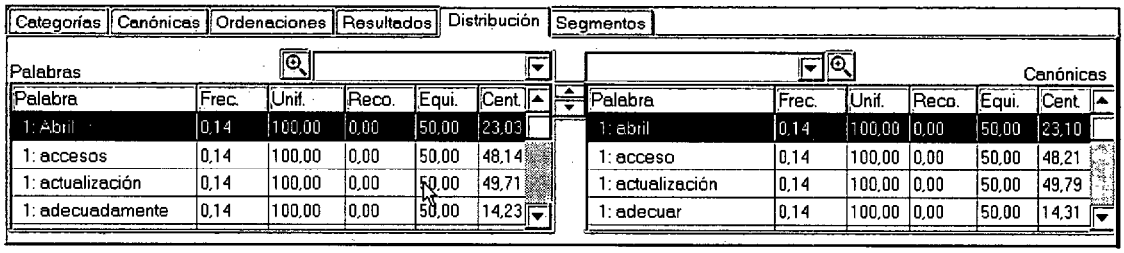



**Figura 46** Vista de resultados

La pestaña *Distribución*, figura 47, muestra los factores de distribución de las palabras cuando se activa *Distribución* en el submenú de análisis; la información se presenta tanto para las palabras presentes en el texto como para sus formas canónicas. En ambos casos —calculadas tal como se describió en la sección 2— aparecen: 1) la palabra o forma canónica acompañada de su frecuencia absoluta, 2) la frecuencia relativa de aparición, 3) la uniformidad de la distribución, 4) el recorrido, 5) el equilibrio interno del recorrido y 6) la centralidad del recorrido en el texto. La información se muestra ordenada por frecuencias crecientes de aparición de las palabras a partir de un umbral —configurable con los botones de puntas de flecha que se hallan entre las cajas de distribución de palabras y formas canónicas.




| Palabra | Frec. | Unif. | Reco. | Equi. | Cent. |
|---|---|---|---|---|---|
| 1: Abril | 0,14 | 100,00 | 0,00 | 50,00 | 23,03 |
| 1: accesos | 0,14 | 100,00 | 0,00 | 50,00 | 48,14 |
| 1: actualización | 0,14 | 100,00 | 0,00 | 50,00 | 49,71 |
| 1: adecuadamente | 0,14 | 100,00 | 0,00 | 50,00 | 14,23 |

| Palabra | Frec. | Unif. | Reco. | Equi. | Cent. |
|---|---|---|---|---|---|
| 1: abril | 0,14 | 100,00 | 0,00 | 50,00 | 23,10 |
| 1: acceso | 0,14 | 100,00 | 0,00 | 50,00 | 48,21 |
| 1: actualización | 0,14 | 100,00 | 0,00 | 50,00 | 49,79 |
| 1: adecuar | 0,14 | 100,00 | 0,00 | 50,00 | 14,31 |

**Figura 47** Vista de distribución

Encima de cada una de las cajas de distribución aparece una lista combinada. Pulsar el botón *lupa* implica encontrar en el texto una lista de palabras o formas canónicas con distribución similar a la que se halle en ese momento seleccionada en la caja de distribución correspondiente, figura 48. La similitud de la distribución entre dos palabras se calcula en función de los mínimos cuadrados de las diferencias de cada uno de los cuatro factores de distribución considerados y de las frecuencias relativas de aparición de cada una —permite obtener pistas interesantes sobre la correlación y la relevancia con que aparecen en el texto determinadas palabras. Se interpreta como una aproximación "al vuelo" a la afinidad entre palabras, de la que se pueden obtener medidas cuantitativas en función de la correlación de las coocurrencias de las palabras en el texto[RODR99]; en este caso, la información que se proporciona es más cualitativa que cuantitativa: obvia el problema de la definición de lo que se considera coocurrencia —no se definen ventanas u otra clase de ámbito

de aparición, sino que se trabaja combinando las medidas de distribución de las palabras obtenidas de manera independiente para cada una— y diferencia entre palabras y clases de palabras —están claramente separadas en la interfaz.

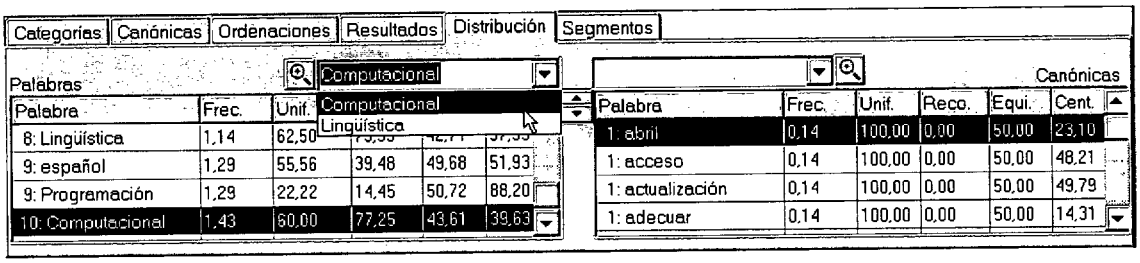



**Figura 48** Despliegue de similares en distribución

La pestaña *Segmentos*, figura 49, sirve para localizar todas las secuencias de palabras —segmentos— que cumplen unas determinadas restricciones —configurables— en cuanto a número de palabras que la integran y frecuencia mínima de aparición en el texto.

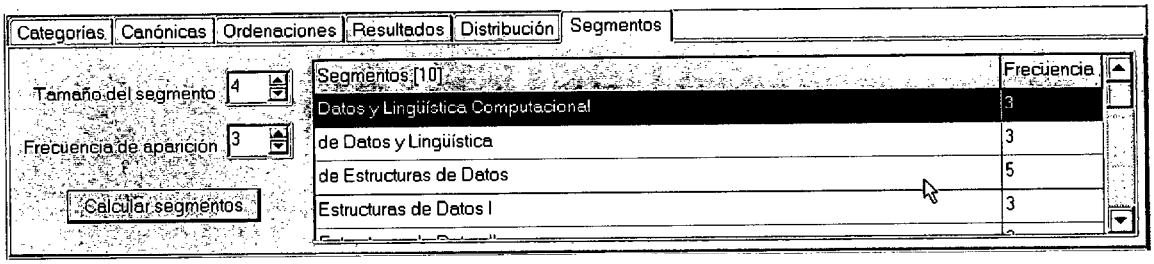



**Figura 49** Pestaña Segmentos

## 7.4.- Sincronización de la información mostrada.

En lo posible, la información de las diferentes zonas de la interfaz de NAWeb se mantiene sincronizada cuando el usuario "se mueve" por ellas, de manera que pueda observar la correlación desde diferentes "vistas" de un fenómeno lingüístico particular en lugar de datos aislados; por ejemplo, a medida que cambia la selección de las palabras en la lista alfabética, se actualizarán las restantes que muestran las palabras según diferentes órdenes o categorías, se resaltará la palabra en la vista del texto, y se sincronizará la vista de la lematización. Las figuras 50 y 51 esquematizan cómo se lleva a cabo la sincronización: los rectángulos oscuros con bordes redondeados indican estímulos de inicio de acción, los rectángulos blancos indican acciones terminales, las flechas continuas indican acciones ejecutadas como reacción a estímulos y las líneas discontinuas indican interferencias en la cadena de acciones. La figura 50 ilustra la sincronización entre los elementos de la vista del texto, la vista de la lematización, la vista de la distribución y las listas de ordenaciones y categorías.

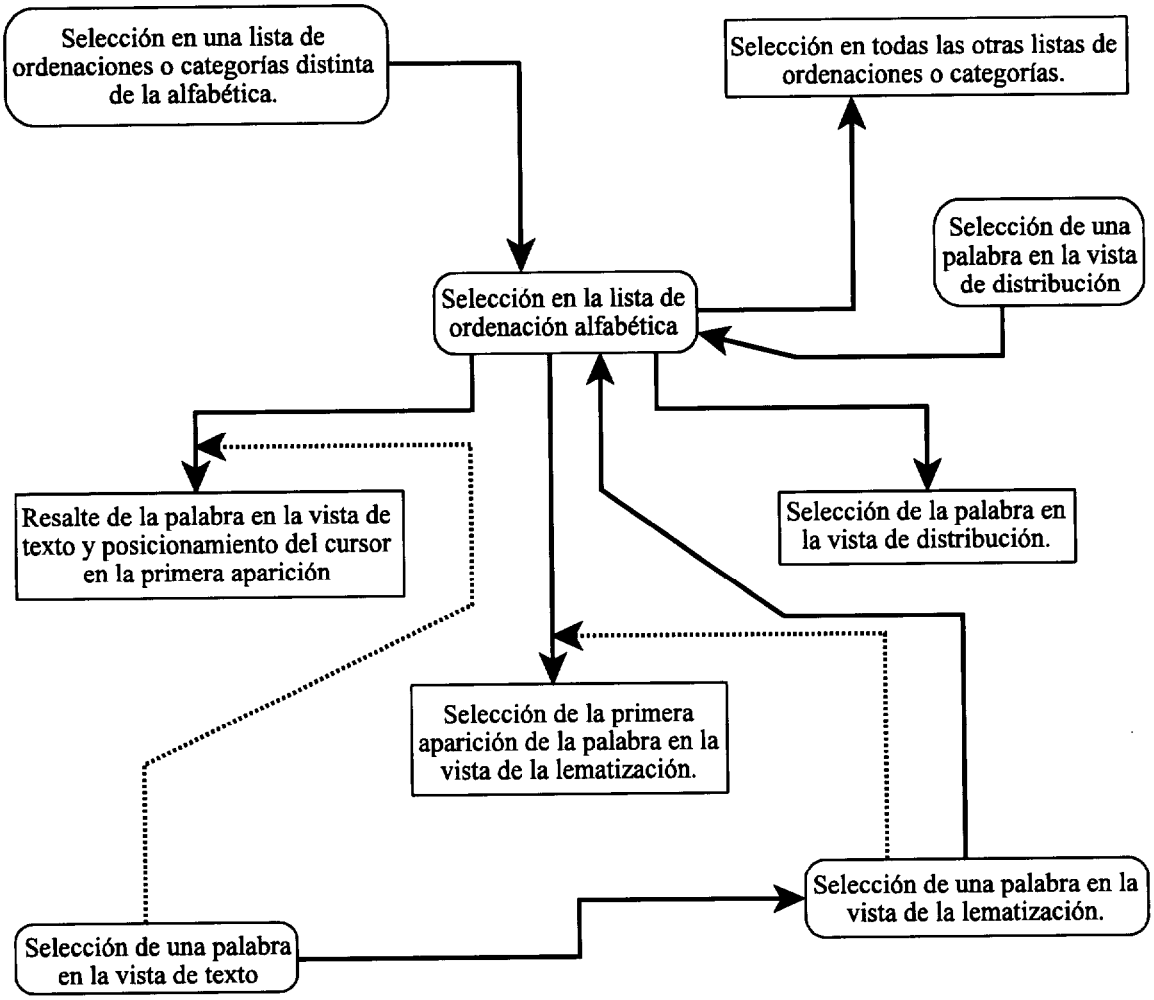



**Figura 50** Esquema de sincronización. Palabras

Excepto en la lista que muestra la ordenación alfabética, la selección explícita de una palabra en cualquier lista de ordenaciones o categorías desencadena un evento de selección en la lista de ordenación alfabética —centro coordinador de la sincronización. La selección de una palabra en la lista de la ordenación alfabética

provoca: 1) la selección de la misma palabra en todas las demás listas de ordenaciones y categorías, 2) la búsqueda y resalte de todas las apariciones de la palabra en la vista del texto y el posicionamiento del cursor en la primera aparición, 3) la selección de la primera aparición de la palabra en la vista de la lematización y 4) la selección de la palabra en la vista de la distribución.

La selección explícita de una palabra en la vista de la distribución provoca un evento de selección de la palabra en la lista de la ordenación alfabética. La selección de una palabra en la vista de la lematización también lanza un evento de selección de la palabra en la lista de la ordenación alfabética, pero acompañado de otro que interfiere la selección de la primera aparición de la palabra en la vista de la lematización que se produciría como respuesta al evento de selección de la palabra en la lista de la ordenación alfabética —de no ser por esta interferencia, nunca se podría seleccionar una palabra directamente en la vista de la lematización, ya que desencadenaría la selección de la primera aparición de la misma en lugar de la deseada. La selección de una palabra en la vista del texto se transforma en un evento de selección en la vista de la lematización y en un evento de interferencia que actúa sobre la reacción de la lista de ordenación alfabética de forma que permite el resalte de todas las apariciones de la palabra en la vista de texto, pero impide que se

reposicione el cursor —apartaría al usuario de la zona en la que ha centrado su atención.

Se establece un segundo circuito de coordinación en relación con las formas canónicas, figura 51: entran en juego las listas alfabética y por frecuencias de las formas canónicas, la vista del texto, la vista de la lematización y la vista de la distribución de formas canónicas.

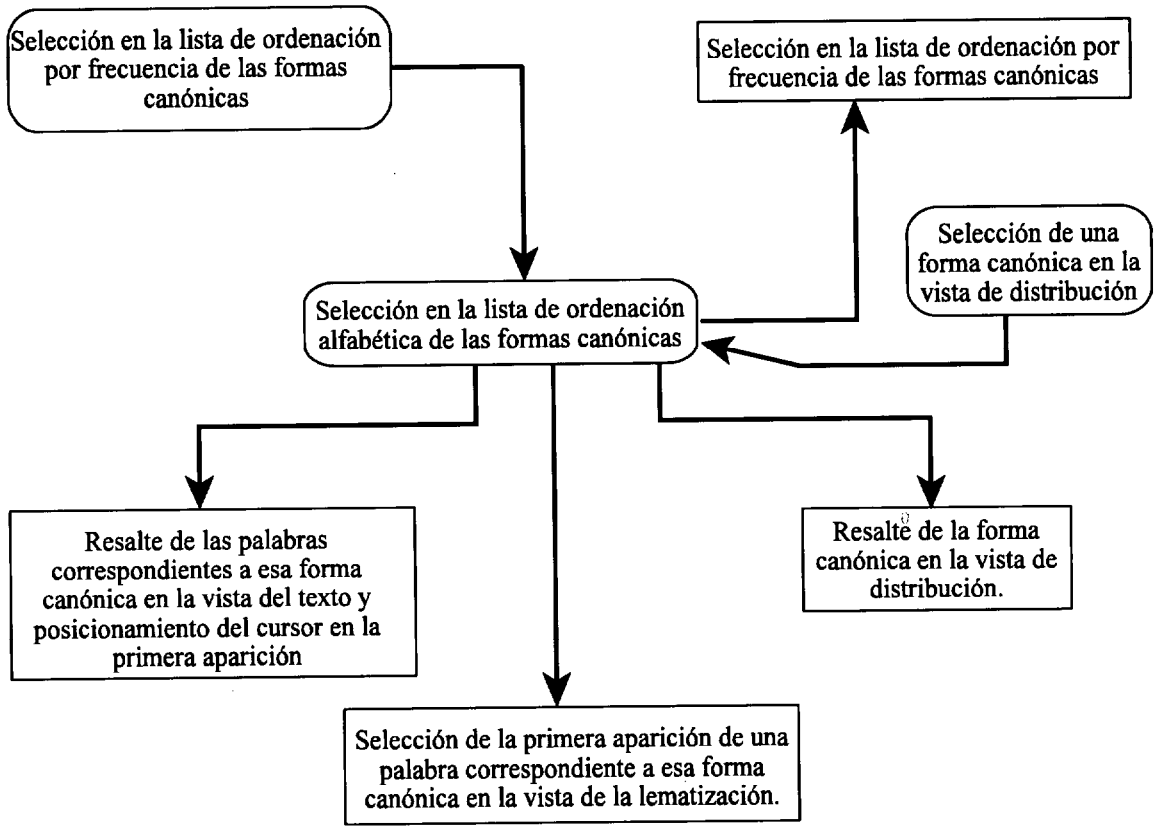



**Figura 51** Esquema de sincronización. Formas canónicas.

La diferencia con el primer circuito es que en el caso de las formas canónicas no se puede sincronizar desde la vista del texto o desde la vista de la lematización hacia las listas de ordenación, a menos que se disponga de una completa desambiguación del texto —a una palabra podría corresponderle más de una forma canónica.

# 8.- Conclusiones y perspectivas futuras.

Esta tesis se inscribe en el interés de la informática hacia todo lo relacionado con el lenguaje. Tal inquietud ha producido y continúa produciendo técnicas y herramientas de ayuda importantes en diversas facetas del trabajo del lingüista. Así mismo, el campo de la informática relacionado con el desarrollo de técnicas de procesamiento del lenguaje natural se ha beneficiado y sigue beneficiándose de la atracción que estas herramientas ejercen en muchos filólogos y de la consiguiente mejora en el conocimiento de las lenguas que semejantes actividades implican.

Esta tesis:

1. Incide en el aprovechamiento lingüístico de un fenómeno sociológico multifacético, Internet, relativamente reciente —en su expansión, porque sus orígenes se remontan a tres décadas— y ya muy consolidado. Dado su origen anglosajón por un lado y el tradicional desfase tecnológico del mundo hispánico por otro, la presencia del Español en la Red se mantiene inferior a la que le correspondería en consonancia con su presencia en el mundo no virtual —tal desventaja se da también con las otras lenguas distintas del inglés—. Se prevé un

importante aumento de la presencia del Español en la red en vista de los enormes beneficios potenciales que ofrece un sustrato tecnológico relativamente tan barato —principalmente, lo fomentarán y aprovecharán los internautas hispanohablantes.

2. Muestra que la disponibilidad de herramientas adecuadas constituye el camino necesario para obtener el debido rendimiento de la metarred como fuente lingüística de caudal nunca antes imaginado. Existe el precedente de la propia expansión de Internet: aunque los elementos físicos estaban disponibles, no empezó a convertirse en un fenómeno de masas hasta que se diseñó el World Wide Web y aparecieron los navegadores adecuados para que los usuarios sin una cualificación informática o tecnológica específica pudieran acceder con facilidad a la información. Ahora ya no se trata de elaborar herramientas básicas, sino unas de carácter más complejo y especializado que permitan descubrir espacios de utilidad en el aprovechamiento de la información que abrieron las primeras aproximaciones. Precisamente, las herramientas relacionadas con el lenguaje auguran la mayor proyección de todas las posibles, ya que trabajan en la base de comunicación de cualquier clase de información.

3. Determina que las enormes posibilidades de aprovechamiento lingüístico de Internet no pueden ser abarcadas por una única herramienta —se volvería

demasiado compleja— y en consecuencia desarrolla dos herramientas con perfiles claramente diferenciados e inspirados en las formas clásicas de acceso a webs:

3.1 La navegación interactiva —NAWeb—, pensada para el estudio detenido de documentos individuales.

3.2 La descarga desasistida —DAWeb, orientada al estudio general de grandes conjuntos de documentos.

4. Aunque NAWeb y DAWeb han sido desarrolladas con el propósito de facilitar el análisis lexicológico de los documentos accesibles por Internet.

4.1. NAWeb resulta útil.

4.1.1 Sin necesidad de acceso externo, como herramienta de autoaprendizaje que permite estudiar y corregir el estilo literario de quien la utilice con ese fin; la posibilidad de analizar documentos en formato MS-WORD resulta de interés.

4.1.2. Con conexión externa, porque facilita el estudio del estilo de aquellos autores cuyos textos sean accesibles. Aproxima el conocimiento de la estructura del léxico: de un texto, autor, época, etc., de la lengua, con las ulteriores aplicaciones, por ejemplo, en la enseñanza de español a extranjeros.

4.2 DAWeb no resulta tan útil en estas facetas porque su carácter no interactivo resta agilidad a la realimentación de las observaciones que se lleven a cabo.

5. A la vista de los grandes volúmenes de palabras que se llegan a encomendar al analizador morfológico, se han estudiado y desarrollado mecanismos capaces de aumentar de forma importante su rendimiento en NAWeb y DAWeb.

5.1. El módulo optimizador de búsquedas morfológicas se implementó como una entidad independiente que ha racionalizado el uso del módulo de análisis morfológico invocándolo sólo cuando es estrictamente preciso —parece lógico usar el procesador morfológico conjuntamente con el optimizador

5.2. En el futuro habría que estudiar la conveniencia de integrar en el módulo de análisis morfológico los mecanismos de optimización, de tal manera que fuese posible configurar con flexibilidad su uso mediante parámetros.

6. Para potenciar el carácter masivo de DAWeb se podría plantear su emplazamiento en una plataforma distribuida.

6.1. La opción más simple sería disponer de dos máquinas enlazadas por una conexión de alta velocidad —las mejoras que se conseguirían con la separación estarían condicionadas por la bondad de la intercomunicación. En una de las máquinas se mantendrían los módulos de descarga, mientras que a la otra se trasladaría el análisis; de esta manera, se sustraería su carga a la fuerte competencia de los hilos de recuperación en paralelo —de máxima prioridad dado que su eficacia tiene un efecto directo en el coste de la recuperación.

6.2. En un planteamiento mucho más ambicioso cabe pensar en una agrupación de máquinas estrechamente vinculadas; varias de ellas ejecutando hilos de recuperación y otras ejecutando módulos de análisis. El estudio empírico de los pros y contras de tal configuración

requiere la aportación de recursos significativos no disponibles en el contexto de desarrollo de esta tesis.

7. La evolución más probable de NAWeb y DAWeb implicará un esfuerzo de: estilización, especialización y extensión. Ambas herramientas han sido diseñadas para servir como prototipos que expongan una amplia paleta de posibilidades en el aprovechamiento de los recursos lingüísticos de la metarred.

8. Ciertamente se obtuvieron dos herramientas potentes con una alta densidad interna y externa, pero convendría darles un aspecto más ligero y adaptado a las distintas modalidades de estudio; en esa dirección, valdría la pena simplificar la abundante información sobre aspectos dispares del documento analizado que proporciona la interfaz de NAWeb. Resulta obvio que no todas las informaciones son necesarias siempre: por ejemplo, si se está llevando a cabo un estudio sobre algún tipo de coocurrencia, la información referida a la ordenación de las palabras por diferentes criterios puede ser irrelevante.

9. En aras de conseguir mejores resultados con arquitecturas más flexibles, DAWeb tenderá a diversificarse en un conjunto de agentes independientes —programas especializados—, pero con capacidad de cooperación mutua que se

activarían conjuntamente sobre el flujo de datos entrante en función de las tareas que se van a realizar —unos se ocuparán del análisis, otros de la búsqueda de coocurrencias, otros de la búsqueda de neologismos, etc.—, de forma que cada uno se encargaría de una tarea diferente y simple —aunque puedan haber varios trabajando en un mismo tipo de tarea. Por ejemplo, en el caso de sólo buscar neologismos se dedicarían 10 agentes a esa labor, mientras que si interesaran además coocurrencias de categorías gramaticales se podrían dedicar 6 y 4 o 3 y 7 agentes o cualquier configuración que aumentase la eficiencia por dedicar más recursos a la tarea que más los necesite —la configuración podría alterarse dinámicamente a medida que evolucionara la situación.

10. En breve, las posibilidades de análisis en red quedarán ampliadas con la incorporación de nuevos desarrollos del Grupo de Estructuras de Datos que se hallan en fase avanzada: mejoras en la desambiguación, herramientas de análisis sintáctico, búsquedas de palabras que pertenezcan a una misma familia, etc. A medio plazo se manejarán también aspectos semánticos.

# 9.- Anexo I: Correspondencia entre secuencias alfabéticas y caracteres.

| Código | Valor | Carácter |
|---|---|---|
| Á | Á | A mayúscula, acento agudo |
| á | á | a minúscula, acento agudo |
| Â | Â | A mayúscula, acento circunflejo |
| â | â | a minúscula, acento circunflejo |
| ´ | ´ | acento agudo |
| Æ | Æ | AE en mayúscula |
| æ | æ | ae en minúscula |
| À | À | A mayúscula, acento grave |
| à | à | a minúscula, acento grave |
| ℵ | ℵ | símbolo alef |
| Α | Α | Letra griega mayúscula alfa |
| α | α | Letra griega minúscula alfa |
| & | & | signo & |
| ∧ | ⊥ | y lógica |
| ∠ | ∠ | ángulo |
| Å | Å | A mayúscula, círculo |
| å | å | a minúscula, círculo |
| ≈ | ≈ | casi igual a |
| Ã | Ã | A mayúscula, tilde |

| Código | Valor | Carácter |
|---|---|---|
| ã | ã | a minúscula, tilde |
| Ä | Ä | A mayúscula, diéresis |
| ä | ä | a minúscula, diéresis |
| „ | „ | comillas dobles y bajas |
| Β | Β | letra griega mayúscula beta |
| β | β | letra griega minúscula beta |
| ¦ | ¦ | barra partida (vertical) |
| • | • | viñeta, pequeño círculo negro |
| ∩ | ∩ | intersección |
| Ç | Ç | C mayúscula, cedilla |
| ç | ç | c minúscula, cedilla |
| ¸ | ¸ | cedilla |
| ¢ | ¢ | signo de céntimo |
| Χ | Χ | Letra griega mayúscula chi |
| χ | χ | letra griega minúscula chi |
| ˆ | ˆ | modificador del acento circunflejo de una letra |
| ♣ | ♣ | trébol |
| ≅ | ≅ | aproximadamente igual a |
| © | © | copyrigth |
| ↵ | ↵ | flecha hacia abajo con esquina hacia la izquierda |
| ∪ | ∪ | unión |
| ¤ | ¤ | signo genérico de moneda |
| ⇓ | ⇓ | flecha doble hacia abajo |

| Código | Valor | Carácter |
|---|---|---|
| ‡ | ‡ | cruz doble |
| † | † | cruz |
| ↓ | ↓ | flecha hacia abajo |
| ° | ° | signo de grado |
| Δ | Δ | letra griega mayúscula delta |
| δ | δ | letra griega minúscula delta |
| ♦ | ♦ | palo de picas |
| ÷ | ÷ | signo de división |
| É | É | E mayúscula, acento agudo |
| é | é | e minúscula, acento agudo |
| Ê | Ê | E mayúscula, acento circunflejo |
| ê | ê | e minúscula, acento circunflejo |
| È | È | E mayúscula, acento grave |
| è | è | e minúscula, acento grave |
| ∅ | ∅ | conjunto vacío |
|   |   | espacio em |
|   |   | espacio en |
| Ε | Ε | letra griega mayúscula épsilon |
| ε | ε | letra griega minúscula épsilon |
| ≡ | ≡ | idéntico a |
| Η | Η | Letra griega mayúscula eta |
| η | η | letra griega minúscula eta |
| Ð | Ð | eth mayúscula, islandés |

| Código | Valor | Carácter |
|---|---|---|
| ð | ð | eth minúscula, islandés |
| Ë | Ë | E mayúscula, diéresis |
| ë | ë | e minúscula, diéresis |
| ∃ | ∃ | existe |
| ƒ | ƒ | f latina minúscula con gancho |
| ∀ | ∀ | para todo |
| ½ | ½ | fracción un medio |
| ¼ | ¼ | fracción un cuarto |
| ¾ | ¾ | fracción tres cuartos |
| ⁄ | ⁄ | barra de fracción |
| Γ | Γ | letra griega mayúscula gamma |
| γ | γ | letra griega minúscula gamma |
| ≥ | ≥ | signo de mayor o igual que |
| > | > | signo de mayor que |
| ⇔ | ⇔ | flecha doble a derecha e izquierda |
| ↔ | ↔ | flecha a derecha e izquierda |
| ♥ | ♥ | palo de corazones |
| … | … | puntos suspensivos horizontales |
| Í | Í | I mayúscula, acento agudo |
| í | í | i minúscula, acento agudo |
| Î | Î | I mayúscula, acento circunflejo |
| î | î | i minúscula, acento circunflejo |
| ¡ | ¡ | signo de exclamación invertido |

| Código | Valor | Carácter |
|---|---|---|
| Ì | Ì | I mayúscula, acento grave |
| ì | ì | i minúscula, acento grave |
| ℑ | ℑ | I mayúscula en letras negras |
| ∞ | ∞ | infinito |
| ∫ | ∫ | integral |
| Ι | Ι | letra griega mayúscula iota |
| ι | ι | letra griega minúscula iota |
| ¿ | ¿ | signo de interrogación invertido |
| ∈ | ∈ | elemento de |
| Ï | Ï | I mayúscula, diéresis |
| ï | ï | i minúscula, diéresis |
| Κ | Κ | letra griega mayúscula kappa |
| κ | κ | letra griega minúscula kappa |
| ⇐ | ⇐ | flecha doble hacia la izquierda |
| Λ | Λ | letra griega mayúscula lambda |
| λ | λ | letra griega minúscula lambda |
| ⟨ | 〈 | paréntesis en ángulo hacia la izquierda |
| « | « | comilla en ángulo, izquierda |
| ← | ← | flecha hacia la izquierda |
| ⌈ | ⌈ | tope superior izquierdo |
| “ | “ | comillas dobles, izquierda |
| ≤ | ≤ | menor o igual que |
| ⌊ | ⌊ | tope inferior izquierdo |

| Código | Valor | Carácter |
|---|---|---|
| ∗ | ∗ | operador asterisco |
| ◊ | ◊ | rombo |
| ‎ | ‎ | marca de izquierda a derecha |
| ‹ | ‹ | comilla simple en ángulo, izquierda |
| ‘ | ‘ | comilla simple izquierda |
| < | < | menor que |
| ¯ | ¯ | macro |
| — | — | barra em |
| µ | µ | micro |
| · | · | punto medio |
| − | − | signo de menos |
| Μ | Μ | letra griega mayúscula mu |
| μ | μ | letra griega minúscula mu |
| ∇ | ∇ | nabla, diferencia hacia atrás |
|   |   | espacio irrompible |
| – | – | barra en |
| ≠ | ≠ | distinto de |
| ∋ | ∋ | lo contiene como miembro |
| ¬ | ¬ | signo de negación |
| ∉ | ∉ | no es un elemento de |
| ⊄ | ⊄ | no es un subconjunto de |
| Ñ | Ñ | N mayúscula, tilde |
| ñ | ñ | n minúscula, tilde |

| Código | Valor | Carácter |
|---|---|---|
| Ν | Ν | letra griega mayúscula nu |
| ν | ν | letra griega minúscula nu |
| Ó | Ó | O mayúscula, acento agudo |
| ó | ó | o minúscula, acento agudo |
| Ô | Ô | O mayúscula, acento circunflejo |
| ô | ô | o minúscula, acento circunflejo |
| Œ | Œ | Ligadura latina mayúscula de OE |
| œ | œ | ligadura latina minúscula de oe |
| Ò | Ò | O mayúscula, acento grave |
| ò | ò | o minúscula, acento grave |
| ‾ | ‾ | línea elevada |
| Ω | Ω | letra griega mayúscula omega |
| ω | ω | letra griega minúscula omega |
| Ο | Ο | letra griega mayúscula omicrón |
| ο | ο | letra griega minúscula omicrón |
| ⊕ | ⊕ | signo más con círculo |
| ∨ | ⊦ | o lógica |
| ª | ª | indicador ordinal femenino |
| º | º | indicador ordinal masculino |
| Ø | Ø | O mayúscula barrada |
| ø | ø | o minúscula barrada |
| Õ | Õ | O mayúscula, tilde |
| õ | õ | o minúscula, tilde |

| Código | Valor | Carácter |
|---|---|---|
| &otimes; | &#8855; | horas rodeadas por círculo |
| &Ouml; | &#214; | O mayúscula, diéresis |
| &ouml; | &#246; | o minúscula, diéresis |
| &para; | &#182; | signo de párrafo |
| &part; | &#8706; | diferencial parcial |
| &permil; | &#8240; | signo de por milla |
| &perp; | &#8869; | ortogonal a, perpendicular |
| &Phi; | &#934; | letra griega mayúscula phi |
| &phi; | &#966; | letra griega minúscula phi |
| &Pi; | &#928; | letra griega mayúscula pi |
| &pi; | &#960; | letra griega minúscula pi |
| &piv; | &#982; | símbolo griego pi |
| &plusmn; | &#177; | signo de más o menos |
| &pound; | &#163; | signo de libra esterlina |
| &Prime; | &#8243; | primo doble, segundos, pulgadas |
| &prime; | &#8242; | primo, minutos, pies |
| &prod; | &#8719; | signo de producto |
| &prop; | &#8733; | proporcional a |
| &Psi; | &#936; | letra griega mayúscula psi |
| &psi; | &#968; | letra griega minúscula psi |
| &quot; | &#34; | comilla |
| &rArr; | &#8658; | flecha doble hacia la derecha |
| &radic; | &#8730; | raíz cuadrada |

| Código | Valor | Carácter |
|---|---|---|
| ⟩ | 〉 | paréntesis en ángulo apuntando hacia la derecha |
| » | » | comilla en ángulo derecha |
| → | → | flecha hacia la derecha |
| ⌉ | ⌉ | tope superior derecho |
| ” | ” | comillas dobles hacia la derecha |
| ℜ | ℜ | R mayúscula en letra negra |
| ® | ® | signo de registrado |
| ⌋ | ⌋ | tope inferior derecho |
| Ρ | Ρ | letra griega mayúscula rho |
| ρ | ρ | letra griega minúscula rho |
| ‏ | ‏ | marca de derecha a izquierda |
| › | › | comilla simple en ángulo apuntando hacia la derecha |
| ’ | ’ | comilla simple hacia la derecha |
| ‚ | ‚ | comilla simple baja |
| Š | Š | letra latina mayúscula s con caron |
| š | š | letra latina minúscula s con caron |
| ⋅ | ⋅ | operador de punto |
| § | § | signo de sección |
|  |  | guión suave |
| Σ | Ω | letra griega mayúscula sigma |
| σ | σ | letra griega minúscula sigma |
| ς | ς | letra griega minúscula final sigma |
| ∼ | ∼ | similar a |

| Código | Valor | Carácter |
|---|---|---|
| ♠ | ♠ | palo de picas |
| ⊂ | ⊂ | subconjunto de |
| ⊆ | ⊆ | subconjunto de o igual a |
| ∑ | ∑ | sumatorio |
| ⊃ | ⊃ | superconjunto de |
| ¹ | ¹ | superíndice 1 |
| ² | ² | superíndice 2 |
| ³ | ³ | superíndice 3 |
| ⊇ | ⊇ | superconjunto de o igual a |
| ß | ß | s aguda minúscula, alemán |
| Τ | Τ | letra griega mayúscula tau |
| τ | τ | letra griega minúscula tau |
| ∴ | ∴ | por lo tanto |
| Θ | Θ | letra griega mayúscula theta |
| θ | θ | letra griega minúscula theta |
| ϑ | ϑ | símbolo de letra griega minúscula theta |
|   |   | espacio estrecho |
| Þ | Þ | thorn mayúscula, islandés |
| þ | þ | thorn minúscula, islandés |
| ˜ | ˜ | tilde minúscula |
| × | × | signo de multiplicar |
| ™ | ™ | signo de marca registrada |
| ⇑ | ⇑ | flecha doble hacia arriba |

| Código | Valor | Carácter |
|---|---|---|
| Ú | Ú | U mayúscula, acento agudo |
| ú | ú | u minúscula, acento agudo |
| ↑ | ↑ | flecha hacia arriba |
| Û | Û | U mayúscula, acento circunflejo |
| û | û | u minúscula, acento circunflejo |
| Ù | Ù | U mayúscula, acento grave |
| ù | ù | u minúscula, acento grave |
| ¨ | ¨ | signo de diéresis |
| ϒ | ϒ | símbolo griego upsilon con gancho |
| Υ | Υ | letra griega mayúscula upsilon |
| υ | υ | letra griega minúscula upsilon |
| Ü | Ü | U mayúscula, diéresis |
| ü | ü | u minúscula, diéresis |
| ℘ | ℘ | P mayúscula con guión |
| Ξ | Ξ | letra griega mayúscula xi |
| ξ | ξ | letra griega minúscula xi |
| Ý | Ý | Y mayúscula, acento agudo |
| ý | ý | y minúscula, acento agudo |
| ¥ | ¥ | signo del yen |
| Ÿ | Ÿ | Letra latina mayúscula Y con diéresis |
| ÿ | ÿ | y minúscula, diéresis |
| Ζ | Ζ | letra griega mayúscula zeta |
| ζ | ζ | letra griega minúscula zeta |

| Código | Valor | Carácter |
|---|---|---|
| &zwj; | &#8205; | ensamblador de extensión cero |
| &zwnj; | &#8204; | desensamblador de extensión cero |

# 10.- Anexo II: Etiquetas HTML.

| Etiqueta | Significado | Acción |
|---|---|---|
| <A> | Anclaje | Quitar etiqueta |
| <ABBR> | Abreviatura | Quitar todo |
| <ACRONYM> | Acrónimo | Quitar todo |
| <ADDRESS> | Dirección | Quitar todo y marcar fin de sección |
| <APPLET> | Aplicación elemental | Quitar todo y marcar fin de sección |
| <AREA> | Área | Quitar todo y marcar fin de sección |
| <B> | Negrita | Quitar etiqueta |
| <BASE> | Base | Quitar todo y marcar fin de sección |
| <BASEFONT> | Tipo de letra base | Quitar etiqueta |
| <BDO> | Anulación bidireccional | Quitar etiqueta |
| <BGSOUND> | Sonido de fondo | Quitar etiqueta |
| <BIG> | Texto grande | Quitar etiqueta |
| <BLINK> | Parpadeo | Quitar etiqueta |
| <BLOCKQUOTE> | Bloques de notas | Quitar todo y marcar fin de sección |
| <BODY> | Cuerpo | Quitar etiqueta |
| <BR> | Corte | Quitar etiqueta |

| Etiqueta | Significado | Acción |
|---|---|---|
| <BUTTON> | Botón | Quitar etiqueta |
| <CAPTION> | Encabezamiento | Quitar etiq. y marcar fin de sección |
| <CENTER> | Centrado | Quitar etiq. y marcar fin de sección |
| <CITE> | Cita | Quitar etiqueta |
| <CODE> | Texto de código | Quitar todo y marcar fin de sección |
| <COL> | Columna | Quitar etiq. y marcar fin de sección |
| <COLGROUP> | Grupo de columnas | Quitar etiq. y marcar fin de sección |
| <DD> | Definición | Quitar etiqueta |
| <DEL> | Texto borrado | Quitar etiq. y marcar fin de sección |
| <DFN> | Definición | Quitar etiq. y marcar fin de sección |
| <DIR> | Directorio | Quitar etiq. y marcar fin de sección |
| <DIV> | División | Quitar etiq. y marcar fin de sección |
| <DL> | Lista de definición | Quitar etiq. y marcar fin de sección |
| <DT> | Término en definición | Quitar etiq. y marcar fin de sección |
| <EM> | Énfasis | Quitar etiqueta |

| Etiqueta | Significado | Acción |
|---|---|---|
| <EMBED> | Contenido empotrado | Quitar todo y marcar fin de sección |
| <FIELDSET> | Conjunto de campos | No debe aparecer |
| <FONT> | Tipo de letra | Quitar etiqueta |
| <FORM> | Formulario | Quitar todo y marcar fin de sección |
| <FRAME> | Marco | Quitar todo y marcar fin de sección |
| <FRAMESET> | | Quitar todo y marcar fin de sección |
| <HEAD> | Sección de cabecera | Quitar todo y marcar fin de sección |
| <H1> | Cabecera nivel 1 | Quitar etiqueta |
| <H2>, <H3>, <H4>, <H5>, <H6> | Cabeceras nivel 2 a nivel 6 | Quitar etiqueta |
| <HTML> | HTML | Quitar etiqueta |
| <I> | Texto en cursiva | Quitar etiqueta |
| <IFRAME> | Marco en línea | Quitar todo y marcar fin de sección |
| <ILAYER> | Capa en línea | Quitar todo y marcar fin de sección |
| <IMG> | Imagen | Quitar etiq. y marcar fin de sección |
| <INPUT> | Entrada | No debe aparecer |

| Etiqueta | Significado | Acción |
|---|---|---|
| <INS> | Texto insertado | Quitar etiq. y marcar fin de sección |
| <ISINDEX> | Isindex | Quitar etiq. y marcar fin de sección |
| <KBD> | Texto de teclado | Quitar etiqueta |
| <LABEL> | Etiqueta | Quitar etiqueta |
| <LAYER> | Capa | Quitar etiqueta |
| <LEGEND> | Leyenda | Quitar etiqueta |
| <LI> | Tema de lista | Quitar etiqueta |
| <LINK> | Vínculo | Quitar etiqueta |
| <LISTING> | Texto de listado | Quitar etiqueta |
| <MAP> | Mapa | Quitar todo y marcar fin de sección |
| <MENU> | Lista de menú | Quitar todo y marcar fin de sección |
| <META> | Metainformación | Quitar etiq. y marcar fin de sección |
| <NOEMBED> | contenido no empotrado | Quitar todo y marcar fin de sección |
| <NOFRAMES> | Contenido sin marcos | Quitar todo y marcar fin de sección |
| <NOLAYER> | Contenido sin capas | Quitar etiq. y marcar fin de sección |
| <NOSCRIPT> | Contenido sin script | Quitar todo y marcar fin de sección |

| Etiqueta | Significado | Acción |
|---|---|---|
| <OBJECT> | Contenido del objeto | Quitar todo y marcar fin de sección |
| <OL> | Lista ordenada | Quitar etiqueta |
| <OPTION> | Opción | No debe aparecer |
| <OPTGROUP> | Grupo de opciones | No debe aparecer |
| <P> | Párrafo | Quitar etiq. y marcar fin de sección |
| <PARAM> | Parámetro | Quitar todo y marcar fin de sección |
| <PERSON> | Persona | Quitar etiqueta |
| <PLAINTEXT> | Texto plano | Quitar etiq. y marcar fin de sección |
| <PRE> | Texto preformateado | Quitar etiq. y marcar fin de sección |
| <Q> | Cita | Quitar etiqueta |
| <S> | Tachado | Quitar etiqueta |
| <SAMP> | Texto ejemplo | Quitar etiqueta |
| <SCRIPT> | Script | Quitar todo y marcar fin de sección |
| <SELECT> | Elección | No debe aparecer |
| <SMALL> | Texto pequeño | Quitar etiqueta |
| <SPAN> | Extensión | Quitar etiqueta |
| <STRIKE> | Texto tachado | Quitar etiqueta |
| <STRONG> | Énfasis fuerte | Quitar etiqueta |

| Etiqueta | Significado | Acción |
|---|---|---|
| <STYLE> | Def. De hoja de estilo | Quitar todo y marcar fin de sección |
| <SUB> | Subíndice | Quitar etiqueta |
| <TABLE> | Tabla | Quitar etiq. y marcar fin de sección |
| <TBODY> | Sección de cuerpo de tabla | Quitar etiq. y marcar fin de sección |
| <TD> | Datos de tabla | Quitar etiq. y marcar fin de sección |
| <TEXTAREA> | Área de texto | Quitar etiq. y marcar fin de sección |
| <TFOOT> | Sección de notas al pie de una tabla | Quitar etiq. y marcar fin de sección |
| <TH> | Cabecera de la tabla | Quitar etiq. y marcar fin de sección |
| <THEAD> | Sección de cabecera de una tabla | Quitar etiq. y marcar fin de sección |
| <TITLE> | Título | No debe aparecer |
| <TR> | Fila de tabla | Quitar etiq. y marcar fin de sección |
| <TT> | Texto de teletipo | Quitar etiq. y marcar fin de sección |
| <U> | Subrayado | Quitar etiqueta |
| <UL> | Lista desordenada | Quitar etiq. y marcar fin de sección |

| Etiqueta | Significado | Acción |
|---|---|---|
| <VAR> | Variable | Quitar todo y marcar fin de sección |
| <XMP> | Ejemplo | Quitar etiqueta |

# 11.- Referencias.

## 11.1.- Libros y artículos.

[COR90] *Introduction to Algorithms*. Tomas H. Cormen; charles E. Leiserson; Ronald L. Rivest. The MIT Press, 1990.

[ZAMP91] *Hacia bases multifuncionales de datos léxicos*. Antonio Zampolli. Las industrias de la lengua. Ed.: Fundación Germán Sánchez Ruipérez, 1 991. 185/202.

[BALL93] *Recuperación de Información en Diccionarios*. Ballester Monzón, A.; Díaz Roca, M.; Santana Pérez, F.; Santana, O. Boletín de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN). Febrero, 1 993.Nº 13, 423/430.

[SANT93a] *Frectext: Una Aplicación de Ayuda a la Elaboración de Documentos.* Santana, O.; Rodríguez del Pino, J. C.; González Domínguez, J. D. Boletín de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN). Febrero, 1 993. N° 13, 451/462.

[SANT93b] *Conjugaciones Verbales.* Santana, O.; Hernández, Z. J.; Rodríguez, G. Boletín de la Sociedad Española para el Procesamiento del Lenguaje Natural (SEPLN). Febrero, 1 993. N° 13, 443/450.

[DÍAZ93] *Distancia Dependiente de la Subsecuencia Común Más Larga entre Cadenas de Caracteres.* Díaz, M.; Pérez, J.; Santana, O. Anales de las II Jornadas de Ingeniería de Sistemas Informáticos y de Computación, Quito (Ecuador). Abril, 1 993. 117/123.

[RODR93] *Agrupaciones de Tiempos Verbales en un Texto.* Rodríguez, G.; Hernández, Z.; Santana, O. Anales de las II Jornadas de Ingeniería de Sistemas Informáticos y de Computación, Quito (Ecuador). Abril, 1 993. 132/137.

[SANT93c] *Información Textual: Línea de Investigación y Proyectos de Desarrollo*. Santana, O.; Díaz, M.; Rodríguez, J. C.; González, D.; Rodríguez, G.; Hernández, Z.; Ballester, A. Español Actual. Ed.: Arco/Libros, S. L. Nº 59/1 993. 31/37.

[SANT94] *Reconocedor de conjugación en formas verbales que trata los pronombres enclíticos*. Santana, O.; Hernández, Z.; Rodríguez, G.; Pérez, J.; Carreras, F.; Bogliani, S. Lingüística Española Actual. Ed.: Arco/Libros, S. L. 1 994, Nº 16-1. 125/133.

[VETT94] *Mosaic an the World-Wide Web*. Ronald J. Vetter; Chris Spell; Charles Ward. Computer, Vol. 27 Nº 10, octubre 1 994. 49/57.

[SANT95a] *Proyecto SOTA: Sistema de Organización de Texto Abierto*. Santana, O.; Hernández, Z.; Rodríguez, G.; Rodríguez, J. C.; González, J. D. Procesamiento de Lenguaje Natural, Revista nº 16. Ed.: SEPLN. Abril, 1 995. Nº 16, 92/94.

[SANT95b] *Proyecto GEISA: GEstión Integrada de Sinónimos y Antónimos.* Santana, O.; Pérez, J.; Santos, S.; Rodríguez, G.; Hernández, Z. Procesamiento de Lenguaje Natural, Revista nº 16. Ed.: SEPLN. Abril, 1 995. Nº 16, 79/81.

[ALAM95] *Diccionario de frecuencias de las unidades lingüísticas del castellano.* José Ramón Alameda; Fernando Cuetos. Servicio de publicaciones de la Universidad de Oviedo. 1 995.

[SANT96] *Diccionarios en soportes informáticos.* Santana, O.; Hernández, Z.; Pérez, J.; Rodríguez, G.; Carreras, F.. Cuadernos Cervantes de la Lengua Española, nº 11 Noviembre - Diciembre, 1 996 68/77.

[SANT97a] *Herramienta para el manejo de diccionarios ideológicos.* Santana, O.; Rodríguez, G.; Hernández, Z. Lingüística Española Actual XIX, 1, 1 997. Ed. Arco/Libros, S.L. 127/136.

[SANT97b] *GEISA: Un diccionario de sinónimos en formato electrónico*. Santana, O.; Pérez, J.; Carreras, F.; Santos, S.; Rodríguez, G.; Hernández, Z. Revista de Lexicografía, Volumen III. Universidade da Coruña. 1 996-1 997. 111/134.

[SANT97c] *FLAVER: Flexionador y lematizador automático de formas verbales*. Santana, O.; Pérez, J.; Hernández, Z.; Carreras, F.; Rodríguez, G. Lingüística Española Actual XIX, 2, 1 997. Ed. Arco/Libros, S.L. 229/282.

[GUS97] *Algorithms on strings, trees, and sequences. Computing Science and Computacional Biology*. Dan Gusfield. Cambridge University Press, 1997.

[ALVA98] *La redacción lexicográfica asistida por ordenador: dificultades y deseos*. Diccionarios e informática, 1 998. Publicaciones de la Universidad de Jaén. 3/22.

[MALD98] *Problemas reales en la elaboración de un diccionario: historia de los diccionarios SM*. Concepción Maldonado González.. Diccionarios e informática, 1 998. Publicaciones de la Universidad de Jaén. 43/55.

[SANT98] *Reconocedor y generador automático de formas nominales*. Santana, O.; Pérez, J.; Carreras, F.; Duque, J.D.; Hernández, Z.; Rodríguez, G. Diccionarios e informática, 1 998. Publicaciones de la Universidad de Jaén. 57/74.

[SANT99a] *De un reconocedor y generador morfológico del español en Internet*. Santana, O.; Pérez, J.; Carreras, F.; Hernández, Z.; Rodríguez, G.; Duque, J.D. Publicado Mayo, 1 999, Lexicon Planet Ltd.

[SANT99b] *FLANOM: Flexionador y lematizador automático de formas nominales*. Santana, O.; Pérez, J.; Carreras, F.; Duque, J.; Hernández, Z.; Rodríguez, G. Lingüística Española Actual XXI, 2, 1 999. Ed. Arco/Libros, S.L. 253/297.

[ABBA99] *Inventing the Web*. J. Abbate. Proceedings of the IEEE, Vol. 87 Nº 11, noviembre 1 999. 1 999/2 002.

[RODR99] *Técnicas estadísticas en el tratamiento del lenguaje natural*. Horacio Rodríguez Hontoria. Filología e Informática. Seminario de filología e informática de la Universidad Autónoma de Barcelona. 1 999. 111/140.

[MILL99] *Estaciones filológicas*. José Antonio Millán. Filología e Informática. Seminario de filología e informática de la Universidad Autónoma de Barcelona. 1 999. 143/164.

[MORR99] *Informática y crítica textual: realidades y deseos*. María Morrás. Filología e Informática. Seminario de filología e informática de la Universidad Autónoma de Barcelona. 1 999. 143/164.

[SANT00] *Generación automática de respuestas en análisis morfológico*. Santana, O.; Pérez, J.; Losada, L. Estudios de lingüística. Universidad de Alicante, 14, 2 000. Departamento de Filología Española, Lingüística General y Teoría de la Literatura. 245/257.

[DAIL01] Translation technologiy tries to hurdle the language barrier. Linda Dailey Paulson. Computer, Vol. 34 Nº 9, septiembre 2 001. 12/15.

**11.2.- Páginas web.**

[WEB01] "A Brief History of the Internet". Internet Society (ISOC). http://www.isoc.org/internet-history/brief.html

[WEB02] "History of the Internet". http://www.isoc.org/internet/history/cerf.html

[WEB03] "History of the Internet". http://showtheplanet.com/history.shtml

[WEB04] *Anuario 2000 del Español en el Mundo.* Centro virtual Cervantes. http://cvc.cervantes.es/obref/anuario/anuario_00/

[WEB05] Centro virtual Cervantes. http://cvc.cervantes.es/

[WEB06] Levenshtein Distance, in Three Flavors. http://www.merriampark.com/ld.htm

[WEB07] Distance Between Strings.

http://www.cut-the-knot.com/do_you_know/Strings.html

[WEB08] Biblioteca Virtual Miguel de Cervantes.

http://www.cervantesvirtual.com/