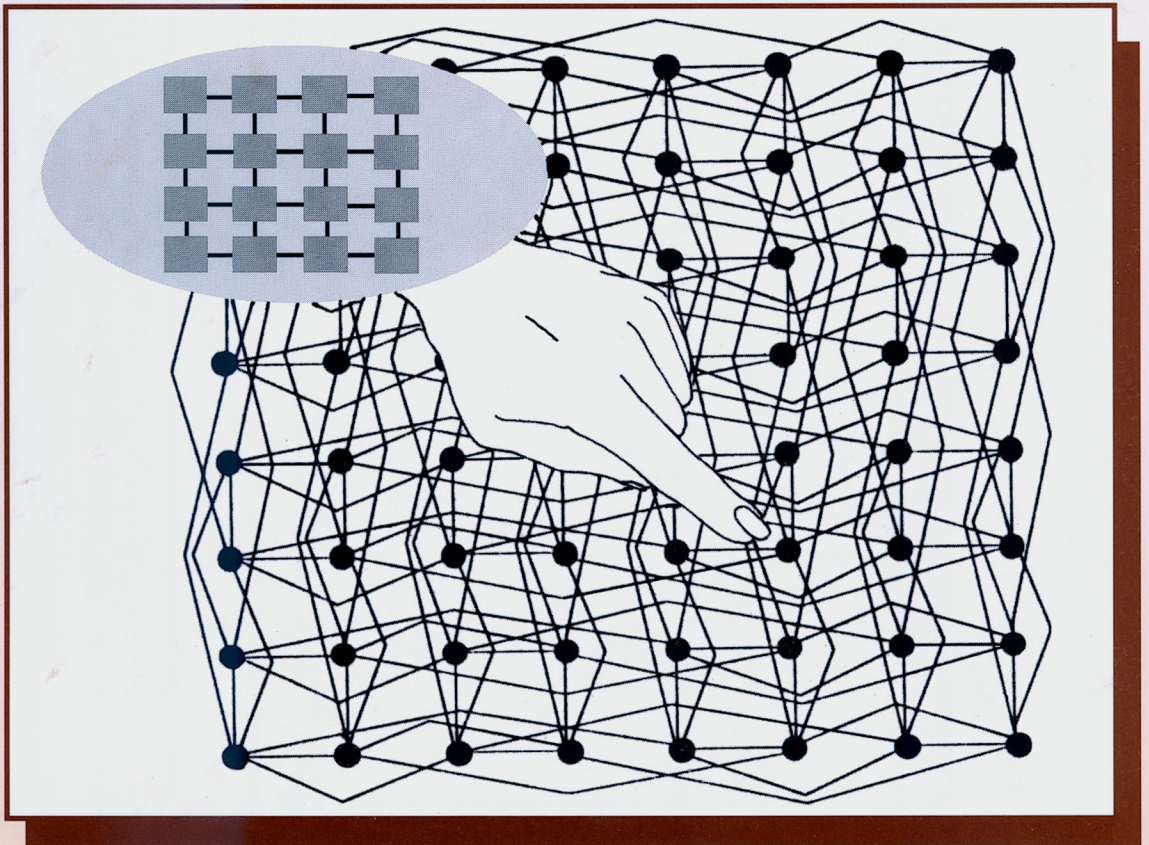


Ingeniería

Revista de la Universidad de Costa Rica
Julio/Diciembre 1996 VOLUMEN 6 Nº 2



INGENIERIA

Revista Semestral de la Universidad de Costa Rica
Volumen 6, Julio/Diciembre 1996 Número 2

DIRECTOR

Rodolfo Herrera J.

CONSEJO EDITORIAL

Víctor Hugo Chacón P.

Ismael Mazón G.

Domingo Riggioni C.

CORRESPONDENCIA Y SUSCRIPCIONES

Editorial de la Universidad de Costa Rica
Apartado Postal 75
2060 Ciudad Universitaria Rodrigo Facio
San José, Costa Rica

CANJES

Universidad de Costa Rica
Sistema de Bibliotecas, Documentación e Información
Unidad de Selección y Adquisiciones-CANJE
Ciudad Universitaria Rodrigo Facio
San José, Costa Rica

Suscripción anual:

Costa Rica: ₡ 1 000,00

Otros países: US \$ 25,00

Número suelto:

Costa Rica: ₡ 750,00

Otros países: \$ 15,00



Edición aprobada por la Comisión Editorial de la Universidad de Costa Rica
© 1998 EDITORIAL DE LA UNIVERSIDAD DE COSTA RICA
Todos los derechos reservados conforme a la ley
Ciudad Universitaria Rodrigo Facio
San José, Costa Rica.

Revisión Filológica: *Lorena Rodríguez*

Diagramación:
José R. Argüello V.

Control de Calidad:
Unidad Diseño Revistas. Oficina de Publicaciones

*Impreso en la Oficina de Publicaciones
de la Universidad de Costa Rica*

Revista
620.005
I-46i

Ingeniería / Universidad de Costa Rica. —
Vol. I, no. 1 (ene./jun. 1991)— . — San José, C. R. : Editorial
de la Universidad de Costa Rica, 1991— (Oficina de Publicaciones
de la Universidad de Costa Rica)
v. : il

Semestral.

I. Ingeniería - Publicaciones periódicas.

CCC/BUCR—250



UNA TAXONOMIA DE PAGINAS EN EL WEB

Carlos Vargas Castillo*

RESUMEN

Se plantea la necesidad de elaborar una metodología que guíe a los analistas de sistemas cuando se enfrenten con la tarea de desarrollar aplicaciones dentro del novedoso ambiente de la *Web*. Seguidamente se identifica el Diagrama de Ligas Asociativas (DLA) como un componente esencial de dicha metodología. Luego se presenta una taxonomía de los DLA basada en las distintas topologías exhibidas por ellos.

SUMMARY

It is pointed out the need for having a methodology that gives guidance to Systems Analysts when they are confronted with the task of applications development under the novelty Web environment. Next, a "Diagram of Associative Links" (DLA) is identified as an essential component of such methodology. Then, a taxonomy of the DLA's is presented based on different topologies showed by them.

1. INTRODUCCIÓN¹

La Gran Telaraña Electrónica (*World Wide Web*), ó WWW, ó simplemente la *Web*², es la rama gráfica de la red Internet. Es bien conocido que esta red se extiende por todo el mundo y que ha experimentado un crecimiento vertiginoso en los últimos años. No se conocen las cifras exactas, pero se sabe que millones de navegantes³ exploran continuamente los miles de sitios *Web* alrededor del mundo utilizando, principalmente, los visualizadores⁴ *Netscape* y *Explorer*.

La red evolucionó rápidamente de un medio para correo electrónico a una importantísima herramienta para la publicidad, los negocios, el entretenimiento y para la investigación. Nuevos usos aparecen constantemente. La importancia de la *Web* se ha incrementado aún más con la integración de las llamadas redes *Intranet*⁵, que son sitios internos

con tecnología *Web* utilizada en aplicaciones normales de la empresa.

2. NECESIDAD DE UNA METODOLOGÍA

La *Web* se ha convertido en una de las tecnologías más importantes de nuestro tiempo y se ha escrito sobre múltiples temas relacionados con ella. Hay numerosos artículos y libros que se han escrito acerca del HTML⁶, el lenguaje modelador de hipertexto (*Hypertext Modeling Language*). El HTML es el lenguaje estándar para la programación de páginas *Web*. También se ha escrito bastante sobre los distintos lenguajes (JAVA, *JavaScript*, VRML, y *Perl*) que coexisten con el HTML y que ofrecen otras capacidades. Además, se han desarrollado una gran cantidad de herramientas de ayuda durante la elaboración de páginas *Web*. Pero si bien es cierto que gran cantidad de herramientas y de lenguajes disponibles han proporcionado muchas facilidades al programador, también han

* Profesor Escuela de Ciencias de la Computación e Informática, Facultad de Ingeniería, UCR.

vuelto mas complejo el desarrollo de aplicaciones *Web*.

Aun cuando en la actualidad la *Web* presenta un ambiente altamente especializado para el desarrollo de aplicaciones, la producción literaria y la creación de herramientas se han orientado principalmente hacia la fase de Implementación descuidando casi por completo las fases de Análisis y Diseño, quizá debido a la forma tan acelerada con que se ha pasado de simples a muy complejas aplicaciones. Las fases de análisis y diseño son reconocidas por la comunidad de ingenieros de sistemas⁷, no solamente como necesarias, sino además como indispensables para la producción de *software* de calidad.

Es necesaria una "Metodología del Análisis y Diseño de Aplicaciones *Web*" que guíe tanto a analistas, como a diseñadores, a programadores y a los administradores de la *Web*, en la tarea de desarrollar y mantener aplicaciones en este contexto. Debe comprenderse que quien desarrolla aplicaciones *Web* debe resolver, además de los problemas presentes en los sistemas tradicionales, las dificultades propias de los ambientes hipertexto, los multimedios y del paradigma cliente/servidor.

En consecuencia, el analista de sistemas puede sentirse desorientado cuando requiera trabajar en un ambiente de tecnología *Web*, debido a que no cuenta con una abstracción apropiada que le permita especificar un diseño de un sitio antes de proceder a su implementación. Un modelo lógico le permitiría obtener un panorama claro de la aplicación completa y de cómo las páginas encajan entre sí para satisfacer los requerimientos previos.

3. DIAGRAMAS DE LIGAS ASOCIATIVAS

Un aspecto esencial del desarrollo de aplicaciones *Web*, que no se encuentra presente en los sistemas de información tradicionales, es la forma en que se organizan las distintas páginas⁸ que componen una presentación. Denominamos "Diagrama de Ligas

Asociativas"⁹, ó DLA, al modelo general que muestra los vínculos¹⁰ entre las distintas páginas. Un diagrama de esta naturaleza brinda una idea general de lo que debe implementarse antes de la verdadera creación de las páginas, independientemente de cómo éstas se confeccionen posteriormente. El contenido de las páginas individuales puede representarse con plantillas prototipo. La ventaja de elaborar un DLA es que permite discutir, y modificar el diseño, si fuera necesario, antes de iniciar su implementación. En la siguiente sección se muestra una clasificación de los diferentes tipos de diagramas DLA junto con recomendaciones sobre su utilización apropiada.

4. TAXONOMÍA DE LOS DLAS

Se presenta una taxonomía de los Diagramas de Ligas Asociativas basada en las distintas topologías exhibidas por estos diagramas. Las distintas estructuras se representan gráficamente mediante íconos, cuyo significado se describe en la figura 1.

La topología se refiere a la conexión establecida mediante ligas locales. Se pueden establecer conexiones a otros sitios localizados en cualquier lugar de la *Web*, simplemente a través de una liga externa, lo que implica que se debe conocer el "URL"¹¹ destino, y además no se puede predecir si el navegante regresará al punto de partida. Sitios externos se distinguen con un ícono diferente al de las páginas locales, y se puede visitar desde cualquier página local dentro de cualquier estructura.



Figura No. 1. Símbolos Gráficos.

4.1 Estructura jerárquica

La estructura de árbol invertido (ver figura 2), la raíz arriba y las hojas abajo, quizá sea el tipo de arquitectura mas ampliamente utilizada, no solo en el campo de la computación, sino también en múltiples áreas del quehacer humano.

La estructura de árbol, aplicada a la organización de páginas *Web*, ofrece una jerarquía que se ajusta bastante bien a los documentos de hipertexto. La navegación se caracteriza por proceder únicamente hacia abajo o hacia arriba en el árbol. Estrictamente hablando, esta estructura es un tipo de grafo simple, en el que la navegación hacia arriba está implícita ya que no interesa sobrecargar visualmente el diagrama en aras de la claridad.

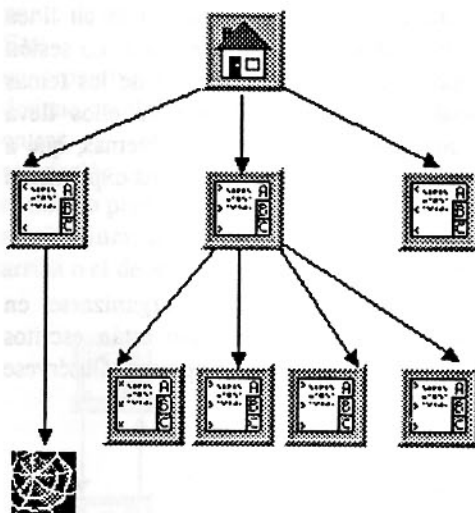


Figura No. 2. Estructura jerárquica.

Normalmente el árbol es desbalanceado y con varios niveles de profundidad. Sin embargo, la arquitectura de un sitio podría representarse de manera simplificada mediante un árbol con únicamente dos niveles. En este caso, la página base tiene una liga a todas y cada una de las páginas en el sitio, y a su vez cada página tiene una liga directa hacia la página base.

Una organización jerárquica de las páginas les facilita a los navegantes deducir el nivel adonde se hallan dentro de la estructura. La página base es la raíz que proporciona el resumen más general del contenido del nivel subsiguiente. También define las principales ligas hacia las páginas en el nivel inferior. Un navegante puede escoger entre desplazarse hacia un nivel superior si busca información general o hacia uno inferior si lo que desea es obtener información más específica. El incluir explícitamente una liga de regreso al nivel superior provee al navegante un medio rápido y fácil de volver a una posición que ya le es conocida.

Un ejemplo representativo del uso de la estructura de árbol es el de la ayuda en línea (*Help*) para una aplicación particular. La sesión de ayuda comienza con un menú de los temas principales. El seleccionar uno de ellos lleva inmediatamente a una lista de subtemas, que a su vez conducen gradualmente a una explicación detallada de un tema específico.

4.2 Estructura lineal

Las páginas pueden organizarse en secuencia, similarmente a como están escritos los documentos y las obras impresas. Obsérvese la figura 3.

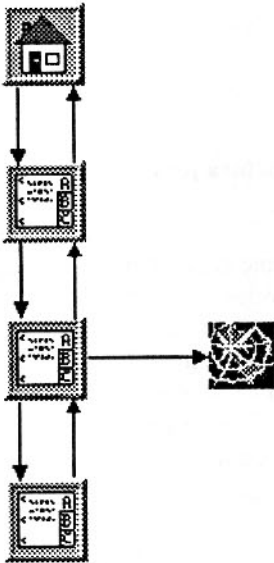


Figura No. 3. Estructura Lineal

Esta es una estructura de lista doblemente enlazada en la que la página base introduce el tema o simplemente presenta el título del tema tratado. Esta organización se caracteriza por el desplazamiento en la navegación que se da únicamente hacia adelante o hacia atrás.

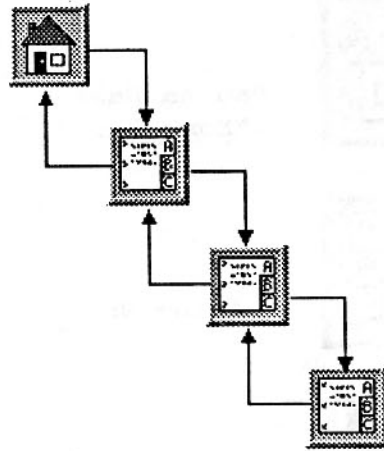


Figura No. 4. Cascada.

También puede conceptualizarse esta estructura como una cascada, mostrada en la figura 4, en la que las páginas están vinculadas en un orden predefinido de modo que únicamente existe un camino a través de todas las páginas del sitio. La organización lineal resulta muy rígida al limitar la libertad de los navegadores para realizar exploraciones y la del analista o diseñador para presentar la información. Sin embargo, la ventaja de dicha estructura radica en que el contexto es deducido con facilidad (no da lugar a que se pierda el navegante) ya que hay muy pocos lugares a donde pueda desplazarse desde su posición.

La estructura lineal es adecuada cuando la información que se presenta tiene una naturaleza secuencial inherente, tal como sucede con los instructivos, los cuentos cortos, la enseñanza programada o los recetarios de cocina.

La estructura lineal puede hacerse un tanto más flexible si se le agrega una liga extra a cada página en la lista que permita regresar de modo directo a la página base, desde cualquier página, tal como se muestra en la figura 5.

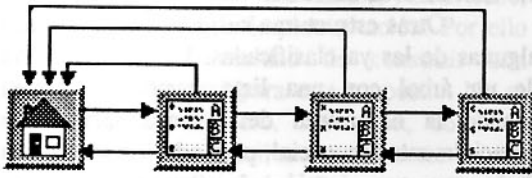


Figura No. 5. Lista con retorno directo.

4.3 Estructura multilínea

Para flexibilizar aún más la estructura lineal, puede permitirse al navegante que se aparte de la ruta o cuerpo principal de la lista. Destacamos dos variantes principales.

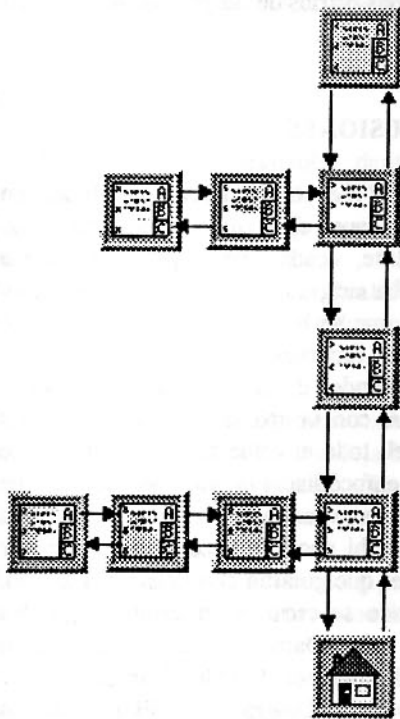


Figura No. 6. Ascensor.

4.3.1 Multilista con ramificaciones controladas

Esta estructura (ver figura 6) permite salidas desde un número reducido de puntos hacia otras

sublistas que retornan exactamente al mismo punto de salida.

Este tipo de organización puede conceptualizarse como un ascensor que se detiene en algunos pisos en los cuales se puede entrar y desplazarse (las sublistas). La restricción es que debe abordarse únicamente en el mismo piso donde se salió para continuar solo al piso inmediatamente contiguo, ya sea el de arriba o el de abajo.

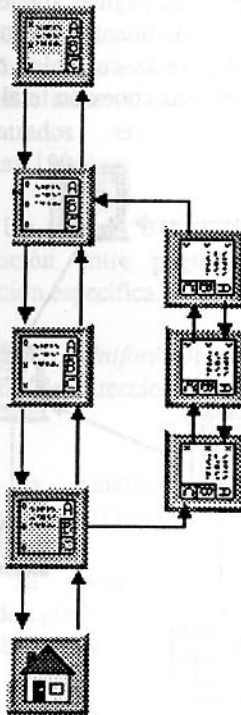


Figura No. 7. Ascensor con saltos.

4.3.2 Multilista con ramificaciones con salto

Similar a la estructura anterior, esta estructura (ver figura 7) permite salidas en un número reducido de puntos hacia otras sublistas. Pero la diferencia consiste en que el retorno no es al mismo punto desde donde se partió, sino que se une con el cuerpo principal en algún otro lugar, saltándose así algunas páginas del. Esta estructura también puede conceptualizarse como un ascensor del que se

sale en un piso pero se aborda en otro distinto (se supone que utilizó las escaleras).

4.4 Estructura telaraña

La estructura que denominamos telaraña, como la mostrada en figura 8, aparece con poca estructura a los ojos del navegante, pero realmente se trata de una estructura de grafo dirigido. Toda la *World Wide Web* es una inmensa estructura de este tipo. La estructura es proporcionada únicamente por las ligas. El navegador se mueve entre las páginas siguiendo las ligas establecidas. Virtualmente, se podría visitar cualquier página desde cualquier otra, pero nos es usual requerir una conexión total.

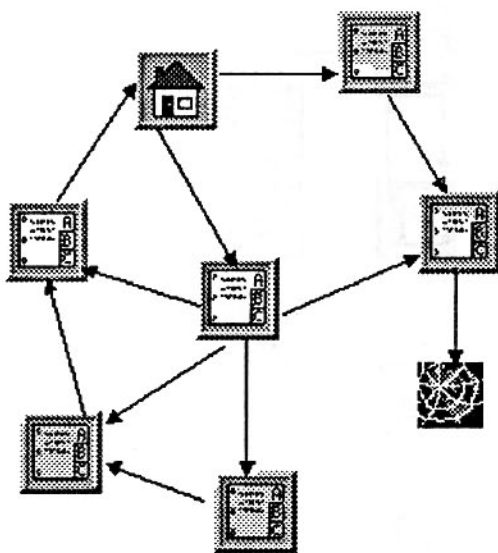


Figura No. 8. Telaraña.

Las estructuras de grafo tienden a ofrecer una navegación muy libre, permitiendo que se transite sin rumbo fijo por todo el sitio. Los grafos deben utilizarse con textos que no guardan una relación estrecha entre sí, o cuando se trata de motivar la exploración por parte del navegante.

4.5 Estructuras híbridas

Otras estructuras surgen al combinarse algunas de las ya clasificadas. La combinación de un árbol con una lista, puede utilizarse cuando la naturaleza de la información es esencialmente secuencial, pero además se quiera darle una organización de acuerdo a algún atributo o criterio particular. Por ejemplo, una compañía productora de *software* podría organizar sus productos en forma jerárquica de acuerdo con sus versiones mientras que la descripción de las "pulgas" encontradas se presentarían de forma natural en secuencia.

Otro ejemplo de estructura híbrida se obtiene al combinar un árbol de dos niveles con un grafo, lo que propicia que se puedan acceder rápidamente todas las páginas desde la página base y, además, varias de las páginas se vinculan entre sí.

5. CONCLUSIONES

La tecnología *Web* es un nuevo paradigma en computación que cada día se afianza más. Históricamente, cada vez que un nuevo paradigma ha surgido, su utilización irreflexiva ha causado que se incurra en altos costos y en desperdicio de recursos.

Debemos aprender de las lecciones del pasado. Notemos que con la aparición de los lenguajes de alto nivel, todo el esfuerzo del desarrollo de sistemas se enfocó hacia la programación, lo que condujo muchas veces a la creación de sistemas pésimos. De ahí nació la necesidad de elaborar abstracciones que guiaran el análisis y el diseño. De este modo se crearon en primer lugar las Estructuras de Datos y la Programación Modular, después el Diseño Estructurado, y posteriormente los Diagramas de Flujo de Datos y los Diagramas de Relaciones entre Entidades¹².

De igual modo, luego de la aparición de la Programación Orientada a Objetos, se elaboraron metodologías para el Análisis y el Diseño de aplicaciones basadas en Objetos¹³.

El desarrollo de aplicaciones *Web* no escapa a la necesidad de que se consideren las

fases de Planeamiento, de Análisis, y de Diseño, complementarias a la Implementación. Por ello enfatizamos la necesidad de construir una metodología integral para este propósito. No se requiere partir de cero, más bien, conviene rescatar y ajustar aquellos componentes de las metodologías tradicionales al nuevo medio, así como considerar los elementos propios de la *Web*, tales como hipertexto y multimedios.

En este artículo hemos introducido un componente importante: el Diagrama de Ligas Asociativas. Sin embargo, una metodología integral deberá considerar muchos otros aspectos, entre ellos: la utilización de plantillas prototipo, Bases de Datos, seguridad, rendimiento, planeamiento de la capacidad, métricas de evaluación, perfiles de los desarrolladores, y elementos de diseño gráfico.

6. REFERENCIAS

(1) Este artículo se desarrolló dentro de los proyectos de Investigación VD-PIE-7-95 Desarrollo de Fascículos y *Software* Académico en Graficación Avanzada, y 326-97314 Desarrollo y Utilización de la Realidad Virtual en la Internet Gráfica, dirigidos por Carlos Vargas Castillo.

(2) Se utiliza el término *Web* sin traducción por cuanto la traducción literal no refleja el concepto representado por este vocablo. Es un caso similar a *Hardware* o *Software*.

(3) Navegante (*Web Surfer*) es la persona que busca información en distintos sitios de la *Web*.

(4) Visualizador, (*Browser*) es el *software* que corre en el Cliente y que permite navegar en la *Web*.

(5) *Intranets* son sitios internos, con tecnología *Web* utilizada en aplicaciones normales de la empresa [Bickel,1996].

(6) HTML es el mecanismo estandar de codificación utilizado por los programadores *Web*.

(7) La justificación en [Taylor, 1987]

(8) Una página *Web* es un documento que consta de textos y gráficos, que conforman un cuerpo informático. La página base (*Home page*) se distingue porque es el inicio de un grupo de páginas.

(9) El modelo propuesto de DLAs y su clasificación se basa en la experticia del autor de tres años inmerso en la *Web* y en los conceptos encontrados en [Shafran,1995] y en [Lemay,1995].

(10) Un vínculo o liga (*hypertextlink*) es una asociación entre páginas que persigue una intención específica.

(11) URL (*Uniform Resource Locator*) es el sistema de direccionamiento para las páginas *Web*.

(12) La temática es muy extensa, ver [Aho,1988] y [Pressman,1993]

(13) El paradigma de Objetos es una metodología bastante utilizada en la actualidad [Rumbaugh,1991]

7. BIBLIOGRAFÍA

1. Aho, Alfred V. et. al. Estructuras de Datos y Algoritmos. Adison-Wesley Iberoamericana, México. 1988.

2. Bickel, Robert. "Web Intranets Boggle the Minds of Traditional Information System Professionals". Internet World, March 1996. pp. 74-76.

3. Lemay, Laura; Aprendiendo HTML para Web en una semana. Editorial Prentice Hall, México, 1995.

4. Pressman, Roger S.; Ingeniería del Software. un Enfoque Práctico. McGraw-Hill/Interamericana de España, S.A., Tercera Edición, 1993.
5. Rumbaugh, James et. al. "Object-Oriented Modeling and Design". Prentice Hall, New Jersey, 1991.
6. Shafran, Andrew B.; "Creating Your Own Nescape Web Pages". Que Corporation, 1995.
7. Taylor, R.N. et. al. "Next Generation Software Environments: Principles, Problems, and Research". Directions Technical Report 87-16, University of California, Irvine, July 1987.