

Generación de sitios Web paramétricos — adaptativos

**Gabriele Facciolo
Gabriel Gesto
Alex Siler
Daniel Srulevich**

Tutor: Eduardo Fernández

**Proyecto de Taller 5
Universidad de la R.O. del Uruguay
Facultad de Ingeniería – In.Co.**

Resumen:

En los últimos años el uso de la Internet y las intranets ha surgido como una herramienta poderosa y de múltiples aplicaciones. A medida que su uso se extiende, los contenidos aumentan y la búsqueda de los intereses de los usuarios se hace cada vez más compleja. Continuamente se encuentran nuevos usos y posibilidades, dando como resultado el crecimiento de los sitios.

Para facilitar la tarea de edición de estos sitios se han desarrollado herramientas que permiten adaptarlos y convertirlos en sitios dinámicos. Sin embargo adaptar los sitios a las necesidades de cada usuario es un problema complejo.

En la mayoría de los casos un diseñador de sitios Web está capacitado para crear páginas informativas, pero no cuenta con el conocimiento suficiente para adaptar el sitio y simplificar la navegación a cada usuario en particular.

Por lo tanto el objetivo de este proyecto es investigar, proponer e implementar tecnologías que ayuden a diseñadores de sitios Web a crear sitios que cambien su estructura y apariencia dependiendo del usuario que lo accede, adaptándose a las características y hábitos de conducta de dicho usuario.

ÍNDICE

1.	Definición del Problema	6
1.1.	Introducción	6
1.1.1	Motivación	6
1.2.	Objetivos	7
1.2.1	Objetivo General	7
1.2.2	Objetivos Particulares	7
1.3.	Conceptos	8
1.3.1	Conceptos generales sobre sitios Web	8
1.3.2	Web Mining	9
1.4.	Sitios Dinámicos Paramétricos Adaptativos	11
1.5.	Herramientas existentes	12
2.	Planteo de la Solución	13
2.1.	Introducción	13
2.2.	Análisis preliminar	15
2.2.1	Sitios Estáticos	15
2.2.2	Sitios Dinámicos	15
3.	Especificación del Análisis	17
3.1.	Análisis de Requerimientos	17
3.1.1	Edición de páginas para incluir Links Dinámicos	17
3.1.2	Configuración general del sistema	18
3.1.3	Generación del sitio con comportamiento paramétrico adaptativo	18
3.2.	Arquitectura del Sistema	19
3.2.1	Diagrama de contexto	19
3.2.2	Entidades externas	20
3.2.3	Proceso Sistema	20
3.2.4	Diagrama de Contexto bajo las condiciones del Taller	21
3.2.5	Entidad externa "Motor de Recomendación"	21
3.2.6	Proceso Generación de sitios Web paramétricos adaptativos	22
3.2.7	Diagrama de Flujo de Datos	23
3.2.8	Procesos principales	24
3.3.	Análisis de la solución	25
3.3.1	Selección de la forma de resolución de páginas dinámicas	25
3.3.2	Método de resolución de páginas dinámicas	27
3.4.	Arquitectura de la solución	30
3.4.1	Expansión del administrador	31
3.4.2	Editor HTML	33
3.4.3	Expansión del generador de comportamiento	35
4.	Diseño del Sistema	37
4.1.	Actores involucrados en el sistema	37
4.1.1	Usuarios del sitio	37
4.1.2	Administradores del sitio	37
4.1.3	Diseñadores del sitio	37
4.2.	Casos de Uso Generales	38
4.3.	Conceptos del diseño de la solución	38
4.3.1	Jerarquización de grupos de páginas	38
4.3.2	Parámetros de Usuarios	39
4.3.3	Reglas de definición del comportamiento dinámico	41
4.3.4	Lenguaje de Consultas para las reglas de comportamiento	44
4.3.5	Texto e Imágenes Asociadas a una URL	45

4.4.	Módulos del Sistema	46
4.5.	Módulo Administrador	48
4.5.1	Casos de uso del Administrador	48
4.5.2	Mantenimiento de Usuarios	49
4.5.3	Mantenimiento de Páginas	52
4.5.4	Configuración del sistema	55
4.5.5	Traductor	56
4.5.6	Dynamix	57
4.6.	Módulo Editor HTML	60
4.5.1	Dynamix	60
4.7.	Módulo Generador de Comportamiento PHP	60
4.6.1	Página de Registro	61
4.6.2	Modulo Data Mining	61
4.6.3	Usuario	61
4.6.4	Acceso a Base de Datos	61
4.6.5	Link Dinámico	61
4.8.	La base de datos	62
4.8.1	MER	62
4.8.2	Tablas	63
4.9.	Portabilidad	65
4.9.1	Utilización de múltiples sitios	65
4.9.2	Ambientes requeridos	65
4.9.3	Comentarios sobre las plataformas soportadas	66
5.	Informe del Proyecto	67
5.1.	Evolución de la Solución	67
6.	Conclusiones finales	69
6.1.	Contribución del proyecto	69
6.2.	Trabajos futuros	69
6.2.1	Extender las posibilidades de elementos dinámicos	69
6.2.2	Extender las posibilidades del Administrador	70
6.2.3	Trabajos futuros sobre los motores estadísticos	70
6.2.4	El Link Dinámico	71
6.2.5	El lenguaje de traducción	71
6.2.6	El lenguaje de consultas DQL	71
Apéndice A		72
Análisis de tecnologías de páginas dinámicas		72
Análisis de opciones		72
Selección del lenguaje		74
Apéndice B		77
Descripción del módulo PHP de Dynamix		77
Introducción		77
Clases del sistema		77
Acotaciones		83
Mejoras futuras		86
Apéndice C		88
Interconexión con el Motor de Recomendación		88
Introducción		88
Descripción de la Componente off-line		88
Descripción de la Componente on-line		89
Intercambio de paquetes para el protocolo on-line		91
Implementaciones		91

<u>Consideraciones</u>	92
<u>Apéndice D</u>	93
<u>Acceso a datos</u>	93
<u>MS Access</u>	94
<u>Oracle</u>	95
<u>MS SQL Server</u>	96
<u>Apéndice E</u>	100
<u>Especificación del lenguaje DQL</u>	100
<u>Sintaxis del lenguaje</u>	100
<u>Semántica del lenguaje</u>	101
<u>Apéndice F</u>	104
<u>Dos ejemplos de especificación de Guardas</u>	104
<u>Apéndice G</u>	108
<u>Configuración del servidor Web con PHP</u>	108
<u>Configuración del DSN</u>	108
<u>Configuración del PHP</u>	109
<u>Configuración del Servidor Web</u>	109
<u>Apéndice H</u>	115
<u>Descripción de la Base de Datos</u>	115
<u>Apéndice I</u>	120
<u>Casos de prueba de la aplicación</u>	120
<u>Apéndice J</u>	144
<u>Glosario de Términos</u>	144
<u>REFERENCIAS</u>	146

1. Definición del Problema

1.1. Introducción

1.1.1 Motivación

En los últimos años el uso de la Internet y las intranets ha surgido como una herramienta poderosa y de múltiples aplicaciones. A medida que su uso se extiende, los contenidos aumentan y la búsqueda de los intereses de los usuarios se hace cada vez más compleja. Continuamente se encuentran nuevos usos y posibilidades, dando como resultado el crecimiento de los sitios.

La World Wide Web (WWW) permite que millones de usuarios, expertos e inexpertos accedan fácilmente a sitios publicados por organizaciones de diversos tamaños, tipos y fines. En los últimos años, los sitios comenzaron a tener miles de accesos diarios, impulsando aún más el crecimiento de la WWW.

En un principio los sitios representaban un medio más de publicación, como libros, revistas, etc. El crecimiento de los sitios hizo que los usuarios encuentren cada vez más difícil hallar contenidos de su interés. Faltaba dar un paso más, conocer los intereses de los usuarios y ayudarlos en la utilización de los sitios. Para esto deben adaptarse los sitios a las necesidades de cada usuario.

Adaptar los sitios a las necesidades de cada usuario es un problema complejo. Actualmente existen muchas organizaciones que cuentan con un sitio en la WWW. Muchas de estas cuentan con personal capacitado en tecnologías de Internet como por ejemplo Java, Java Script, VB Script, ASP, CGI, etc., pero muchas otras tan solo cuentan con un diseñador HTML.

En la mayoría de los casos un diseñador de sitios *Web* está capacitado para crear páginas informativas, pero no cuenta con el conocimiento suficiente para adaptar el sitio y simplificar la navegación a cada usuario en particular. Incluso las organizaciones que cuentan con programadores para su sitio, no suelen tener profesionales capacitados en las técnicas de programación adaptativa.

Generación de sitios Web paramétricos adaptativos

Definición del problema

1.2. Objetivos

1.2.1 Objetivo General

Investigar, proponer e implementar tecnologías que ayuden a diseñadores de sitios *Web* a crear sitios que cambien su estructura y apariencia dependiendo del usuario que lo accede, adaptándose a las características y hábitos de conducta de dicho usuario.

Estas tecnologías, permitirán que los usuarios que accedan a un sitio puedan utilizarlo de forma eficiente, accediendo de una manera más sencilla a los contenidos de su interés.

Para poder desarrollar tecnologías como las descritas, se necesitan considerar dos aspectos. El primero es la recolección y análisis de la información disponible de los usuarios del sitio, el segundo es la forma en que los diseñadores puedan hacer uso de estas tecnologías.

El proyecto se dividió en dos grupos de taller. El primero grupo, encargado de la recolección y análisis de la información, y el segundo grupo, esta encargado de generar un producto que interactúe con los diseñadores para facilitar la generación de un sitio *Web* como el descrito anteriormente.

El primer taller está titulado "*Análisis de patrones de búsqueda con Web Mining*" y el segundo "*Generación de sitios Web paramétricos adaptativos*" ¹

1.2.2 Objetivos Particulares

Nuestro grupo de taller es el encargado de la interacción con los diseñadores. El objetivo es implementar una herramienta que ayude a diseñadores de sitios *Web*, a crear o editar sitios que se adapten a los usuarios que los acceden, de forma que éstos puedan encontrar más fácilmente contenidos de su interés.

La herramienta debe estar orientada a diseñadores que no necesariamente cuenten con conocimientos de programación. Dicha herramienta, debe poder ayudar al diseñador a parametrizar el sitio según las características de los usuarios que lo acceden. Esto implica que el diseñador pueda definir aquellas características que interesan de los usuarios. También es importante proveer facilidades de edición gráfica, presentes en la mayoría de las herramientas de edición actuales. Además, una vez que el sitio este construido y publicado, la herramienta debe proporcionar una forma de modificar la manera en que éste se adapta a los usuarios.

¹ Más adelante se detallan los conceptos que aparecen en los nombres de los talleres.

Generación de sitios Web paramétricos adaptativos

Definición del problema

1.3. Conceptos

1.3.1 Conceptos generales sobre sitios Web

En la sección anterior se mencionaron conceptos como Internet, Sitios, páginas, diseño de un sitio. A continuación profundizaremos en estos términos a los efectos de definir claramente a que se refiere cada uno de estos. No se pretende dar una definición formal de cada concepto, sino simplemente fijar las ideas más importantes.

Internet es una red de alcance global y de acceso público, últimamente muy utilizada por personas y organizaciones para publicar información. La información se organiza en páginas, y las páginas a su vez en sitios. El lenguaje utilizado para escribir las páginas es HTML (Hyper Text Markup Language) [1] [10]. Este es interpretado por aplicaciones llamadas navegadores, que muestran el contenido en la pantalla. Las páginas se han vuelto muy populares ya que permiten desplegar imágenes, videos, sonidos, y una importante gama de contenidos multimedia.

HTML está conformado por texto y Marcas (*tags*). Estas marcas indican la manera en que se despliegan los distintos elementos de una página. Todas las marcas tienen una instancia de comienzo y una de fin. Por ejemplo indica que el texto comienza a desplegarse en negritas (Bold) y indica que el texto deja de desplegarse en negritas. Hay *tags* para incluir imágenes, audio, video, etc. Otro factor que hace que HTML sea un lenguaje poderoso es la posibilidad de incluir hipervínculos. Los hipervínculos permiten mediante un simple clic del botón del mouse desplazarse hacia otras páginas. Esta acción es llamada "navegar".

Cada documento y recurso en Internet tiene una única dirección conocida como URL (Uniform Resource Locator)[2]. Como las direcciones URL pueden ser usadas para representar casi cualquier recurso en Internet, estas pueden ser de distinto tipo: http, javascript, ftp, file, news, nntp, malito, telnet o gopher. Por ejemplo <http://www.fing.edu.uy/index.html> es una dirección http univoca a una página ubicada en el servidor de la facultad de ingeniería.

Los sitios están almacenados en servidores HTTP (Hyper Text Transfer Protocol) [3] y son accedidos por los navegadores. Los servidores envían a los navegadores los archivos HTML mediante el protocolo HTTP.

Como mencionamos antes, al desarrollarse Internet, cada vez se hace más difícil encontrar información. Como primera solución a éste problema se desarrollaron aplicaciones a través de la WWW que permiten encontrar páginas que contienen la información buscada. Estas se denominan buscadores.

Generación de sitios Web paramétricos adaptativos

Definición del problema

Otra solución al problema fue mejorar cada vez más HTML para incluir nuevas funcionalidades. A medida que los estándares de HTML evolucionaron, se fueron incluyendo mejoras al lenguaje que permitieron la incorporación a las páginas de nuevos objetos y aplicaciones. Entre ellos se encuentran por ejemplo Java Applets [4] y ActiveX [5]; estas son aplicaciones que se ejecutan en la computadora donde se encuentra el navegador.

A pesar de las mejoras incluidas, por la forma en que fue concebido HTML, las páginas escritas en HTML son estáticas, o sea que siempre que se accede a la página se ve lo mismo. Muchas veces esto no resulta conveniente, por lo que se han desarrollado varios mecanismos para poder modificar el contenido de las páginas, dando a los usuarios la sensación de dinamismo. Algunos de ellos permiten "mejorar" HTML incluyendo lenguajes tales como ASP [6], PHP [7], otros utilizan aplicaciones CGI (Common Gateway Interface) [8] que generan el código HTML correspondiente a la página que se quiere mostrar. También se puede utilizar Java-Servlets [9] similares a CGI pero ejecutando Java. [Apéndice A]

El protocolo HTTP permite definir cookies [3], estas vienen a resolver un inconveniente del HTTP, que es un protocolo sin estados, es decir no guarda información sobre anteriores conexiones del usuario. Las cookies son datos que quedan almacenados en los navegadores de los usuarios y que el servidor puede recuperar cuando la página vuelve a ser accedida desde el navegador.

Los servidores dejan además archivos de registro (*logs*) con información referente a la navegación que se efectúa. Los *logs* entre otras cosas registran las cookies y por supuesto el acceso a las páginas del sitio. Estos *logs* permiten acceder a la información de navegación capturada a medida que el usuario interactúa con el sitio. Luego analizando esta información se pueden obtener datos relevantes, que interpretados correctamente revelan información a partir de las cuales se pueden hacer deducciones sobre hábitos de comportamiento.

1.3.2 Web Mining

Data Mining

A medida que la información almacenada en forma electrónica aumenta exponencialmente, se hace cada vez más difícil de procesar y analizar manualmente. Es por esta razón que últimamente ha tomado gran fuerza el estudio de técnicas de minería de datos (Data Mining en inglés). La minería de datos se enfoca en la obtención de patrones de información en forma automática, lo cual nos permite descubrir relaciones no triviales en los datos almacenados.

Web Mining

Una de las variantes de Data Mining que ha tomado más fuerza en los últimos tiempos es lo que se llama *Web Mining* [11] [12]. *Web Mining* es la búsqueda y análisis automático de información disponible en la WWW.

Existen 3 principales enfoques referentes al *Web Mining*:

Generación de sitios Web paramétricos adaptativos

Definición del problema

- *Web Content mining*
- *Web Structure Mining*
- *Web Usage Mining*

Web Content Mining

Se basa en la idea de buscar información y extraer conocimiento automáticamente a partir del contenido y descripciones de los documentos de la WWW. Esto se hace cada vez más necesario ya que el sistema de hipervínculos hace complicado el descubrimiento de la información en una red de las dimensiones actuales. Se trata de una tecnología nueva en la que aún hay mucho por avanzar. Algunas de las técnicas aplicadas son:

- Agentes de búsqueda inteligente
- Bases de Datos de contenido

Web Structure Mining

Es el proceso de inferir conocimiento a partir de la organización y referencias entre links de la WWW. La cantidad de links hacia un documento puede dar una idea de la importancia y la popularidad de un documento. Algunos buscadores utilizan esta técnica para ordenar los documentos encontrados en las búsquedas.

Web Usage Mining

Web Usage Mining es un tipo de *Web Mining* que se enfoca al descubrimiento y análisis de patrones de acceso (o hábitos) de los usuarios desde uno o más sitios *Web*, mediante la extracción de información implícita en la navegación.

Generalmente se basan en el análisis de los *logs* generados por los servidores los cuales contienen la información de navegación de los usuarios que acceden a un sitio. De esta manera las organizaciones pueden obtener información útil para el mercadeo en base al perfil de los usuarios que acceden a su sitio, y sus hábitos de navegación. Esta información puede ser utilizada luego para mejorar la estructura del sitio y facilitar la navegación ofreciendo al usuario lo que seguramente este más interesado en ver. [4]

Generación de sitios Web paramétricos adaptativos

Definición del problema

1.4. Sitios Dinámicos Paramétricos Adaptativos

Un sitio *Web* es un conjunto de páginas. Dentro de las páginas hay referencias a direcciones URL (hipervínculos, imágenes, etc.). Es posible definir un grafo dirigido donde los nodos representan las direcciones URL y las aristas representan las referencias entre direcciones. Llamaremos a dicho grafo, Grafo de Navegación asociado a un sitio.

Normalmente el grafo de navegación asociado a los sitios *Web* son estáticos y conexos. A partir de una página inicial, los usuarios navegan por el sitio en búsqueda de las páginas que sean de su interés. Estos sitios obligan al usuario a encontrar la información que busca dentro del sitio, lo que suele ser lento y tedioso. Muchas veces en sitios de gran tamaño los usuarios no encuentran la información que buscan. Una alternativa a este esquema de sitios, son los sitios dinámicos paramétricos adaptativos.

Un sitio es dinámico cuando se modifica su apariencia automáticamente según ciertos criterios. Una posibilidad es que cambie su apariencia entre los distintos usuarios que lo acceden en el mismo momento. Por lo tanto en un sitio dinámico tanto la estructura como el contenido de las páginas pueden variar con cada acceso.

Un sitio adaptativo, es un sitio dinámico que adapta su estructura y el contenido de sus páginas de acuerdo a las necesidades del usuario. Al adaptar el sitio al usuario, se intenta minimizar el tiempo que el usuario debe dedicar a la búsqueda de la información de su interés y maximizar el beneficio que puede obtener del sitio.

Un sitio paramétrico, es un sitio dinámico en el cual existen una serie de parámetros de usuario y reglas que sirven para definir la estructura del sitio y el contenido de sus páginas de acuerdo a sus necesidades.

Generación de sitios Web paramétricos adaptativos

Definición del problema

1.5. Herramientas existentes

Se realizó un relevamiento de herramientas existentes en el mercado, que cumplan con objetivos similares a los definidos para este proyecto. Para esto se realizó una búsqueda de productos en Internet, enfocada hacia aplicaciones que provean una combinación de funciones de Data Mining y ayuda en la creación de sitios adaptables al usuario aplicando Data Mining.

Se encontraron varios productos con características de Data Mining y otros con ayuda para creación de sitios dinámicos pero ninguno de ellos cumple exactamente con los objetivos buscados. La razón para esta conclusión se debe a que varios productos ofrecen motores de Data Mining muy avanzados y otras facilidades muy interesantes para la creación de sitios, pero no se ha encontrado un producto que integre estos dos puntos enfocándolos hacia el diseñador gráfico. En varios productos se ofrecen informes y estadísticas muy detalladas sobre el comportamiento de los usuarios del sitio, pero no se permite indicar un comportamiento automático basado en esta información.

En el Apéndice B se describen algunos de los productos más relevantes que fueron analizados:

- Accrue G2 y Accrue Insight 5
- Microsoft Commerce Server 2000
- Knowledge SERVER
- Knowledge Access Suite y Data Mining Suite
- *Web Analyst*
- *Web mining Genius*, Genius Server
- Virtual Miner
- I/PRO NETLINE

Ver [Apéndice B]

2. Planteo de la Solución

2.1. Introducción

En una página *Web*, pueden aparecer elementos de distinto tipo, como texto, imágenes, sonido, video, etc. En principio no hay forma de saber si el contenido de la página es atractivo para el usuario. Se puede suponer que las páginas más accedidas por un usuario en particular son las de mayor interés para dicho usuario. Esta suposición será la base del análisis del problema.

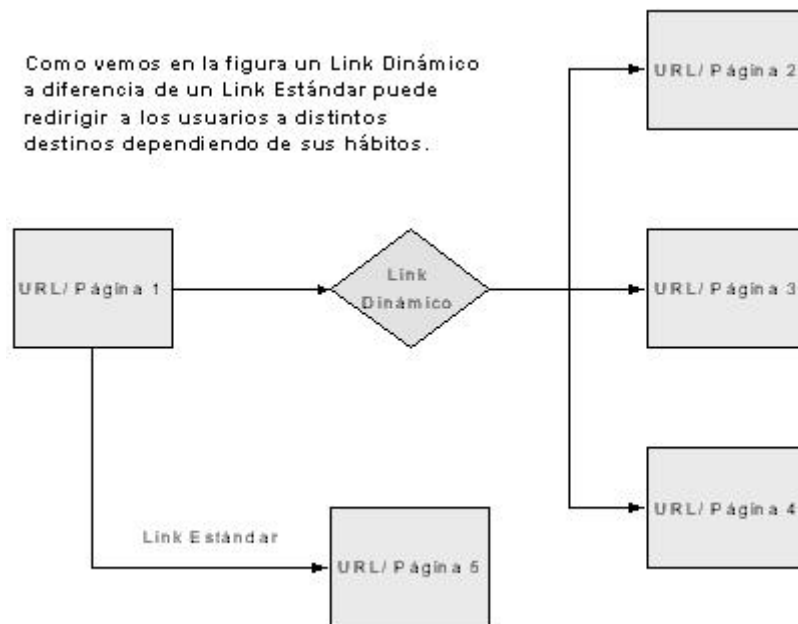
Uno de los objetivos es crear sitios que se adapten a los usuarios que los acceden, para que resulte más fácil encontrar contenidos de su interés. Una aproximación posible para lograr este objetivo, es que cuando se accede a una página del sitio, se modifiquen los hipervínculos que ésta contiene de forma que reflejen los intereses del usuario. Para modificar los hipervínculos se debe identificar los documentos de mayor interés para el usuario que accede a la página en ese momento, y cambiar la dirección URL de los hipervínculos de acuerdo a éstos.

Los hipervínculos estándar de HTML, no son capaces de modificar la dirección URL destino. Por lo tanto es necesario definir nuevos hipervínculos, que se adapten a los usuarios que acceden la página *Web*. A estos nuevos hipervínculos los llamamos "Links Dinámicos" y sobre éstos se basa la nueva herramienta de edición.

Los Links Dinámicos son similares a los hipervínculos estándar de HTML en lo que respecta a su funcionalidad de dirigir al usuario a otra dirección URL. La diferencia está en que los Links Dinámicos pueden modificar la dirección URL destino en base a ciertos parámetros y hábitos del usuario.

Generación de sitios Web paramétricos adaptativos

Planteo de la solución



Los Links Dinámicos deben poder ser configurables de manera de poder determinar la dirección URL destino según el caso. Esta configuración se realiza estableciendo una serie de reglas a aplicar para obtener la URL destino que verá el usuario. Se define además un criterio para determinar cual de las reglas hay que aplicar según las características del usuario. Por ejemplo se pueden definir dos reglas, una para hombres y otra para mujeres. Las reglas permiten en base a la información del *log* recomendar páginas.

La utilización de Links Dinámicos en un sitio, permite que los usuarios tengan que navegar por una menor cantidad de páginas hasta llegar a una de su interés. Así no solo se ahorra tiempo en la búsqueda de contenidos de interés, sino que aumenta la probabilidad de que realmente encuentre el contenido buscado. Hay que tener en cuenta que en sitios de grandes dimensiones, donde las páginas tienen varios hipervínculos, a veces resulta una tarea muy difícil y tediosa recorrerlo en busca de una información en particular.

Desde el punto de vista del diseñador del sitio, la utilización de Links Dinámicos no solo permite crear sitios que resulten más atractivos a los usuarios sino también la posibilidad de focalizar las estrategias publicitarias hacia grupos específicos de la población de usuarios.

Generación de sitios Web paramétricos adaptativos

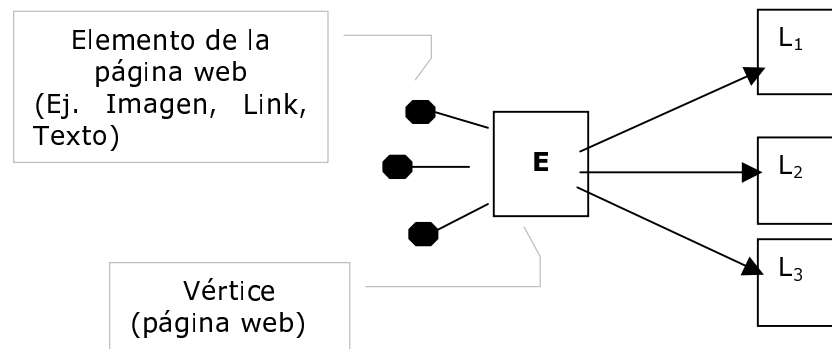
Planteo de la solución

2.2. Análisis preliminar

La siguiente sección no pretende ser un análisis formal, si no que en ella se plantea un modelo sencillo de representación de sitios *Web* para facilitar la comprensión de los conceptos que expondremos en el resto del documento.

2.2.1 Sitios Estáticos

Para representar un sitio estático se utilizará un grafo dirigido conexo. En este modelo los vértices representan a las páginas del sitio y las aristas a los hipervínculos que hay entre estas. De este modo una arista que une un nodo origen a un nodo destino indica que en la página representada por el nodo origen, hay un hipervínculo a la página representada por el nodo destino.



E = Página Web Estática

2.2.2 Sitios Dinámicos

Este modelo, no permite representar sitios dinámicos paramétricos adaptativos, ya que no hay manera de reflejar la dirección URL de un Link Dinámico. Para poder modelar esta situación, introduciremos en la representación, un nuevo elemento al que llamamos vértice decisional.

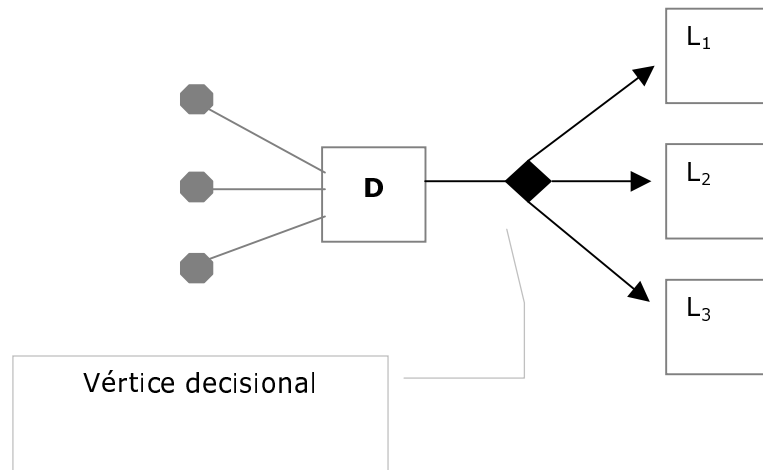
De esta manera, la representación incluye tres objetos:

- vértices
- aristas
- vértices decisionales

Generación de sitios Web paramétricos adaptativos

Planteo de la solución

Un vértice decisional une a un nodo origen con N nodos destino. Una arista que une un nodo origen a un vértice decisional indica que en la página representada por el nodo origen, hay un Link Dinámico que puede llevar a alguna de las N páginas representadas por los N nodos destino.



D = Página Web Dinámica

3. Especificación del Análisis

3.1. Análisis de Requerimientos

En esta sección analizaremos las principales funcionalidades que debe incluir el sistema. Una vez definidos los módulos que lo integran, se realiza un análisis particular detallado para cada uno de estos.

- i. Edición de páginas para incluir Links Dinámicos
 - a. Definición de la estructura del sitio
 - b. Mantenimiento y configuración de Links Dinámicos en páginas
- ii. Configuración general del sistema
 - a. Parámetros de configuración de la aplicación
 - b. Definición de propiedades de usuario
 - c. Categorización de páginas
 - d. Agrupación de usuarios
- iii. Generación del sitio con comportamiento paramétrico adaptativo
 - a. Generación de página de registro de nuevos usuarios
 - b. Generación de las páginas con el comportamiento de Links Dinámicos incluido

3.1.1 Edición de páginas para incluir Links Dinámicos

Como vimos anteriormente el Link Dinámico es la base de la solución brindada. Como primer requerimiento, se debe permitir que los diseñadores puedan incluir los Links Dinámicos en un sitio. Para esto habrá que brindar una forma de insertar, eliminar y configurar los mismos. La herramienta debe proveer funcionalidades especiales, preferiblemente gráficas, de modo que la tarea de modificar el comportamiento no represente una dificultad equivalente a especializarse en un lenguaje de programación u otra tecnología *Web*.

Es conveniente facilitar una manera de visualizar y editar la estructura del sitio en donde se identifiquen claramente los hipervínculos y Links Dinámicos que relacionan unas páginas con otras de forma de controlar la navegación, incluso en sitios de grandes dimensiones.

Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

3.1.2 Configuración general del sistema

Para que el sitio sea paramétrico se debe poder definir cuales son los parámetros que interesan de los usuarios. Estos pueden ser propiedades de los usuarios, como por ejemplo, el sexo, la nacionalidad, el rango de ingresos, el país, etc.

Mediante los parámetros se podrá categorizar a los usuarios, por ejemplo se puede crear un grupo de hombres, uno de ciudadanos uruguayos, uno de personas de alto poder adquisitivo, etc. Esta categorización podrá utilizarse para determinar la adaptación del sitio al comportamiento de los usuarios. Además, es útil poder clasificar las páginas en grupos, de forma que sea más fácil seleccionar los destinos que se adapten a los intereses de los usuarios.

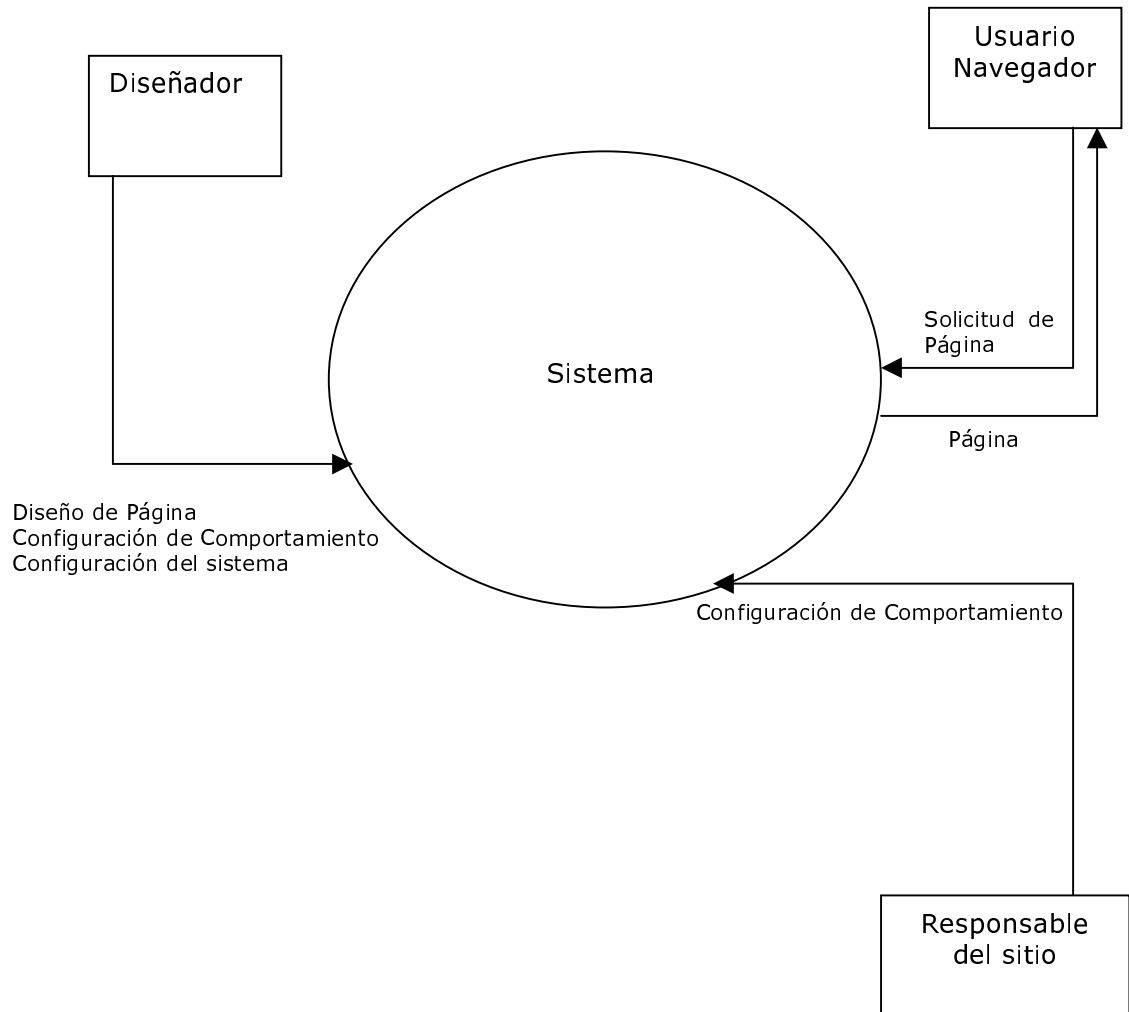
3.1.3 Generación del sitio con comportamiento paramétrico adaptativo

En base a la definición y configuración de la estructura del sitio y los Links Dinámicos, se debe generar un sitio que implemente el comportamiento especificado por el diseñador web.

Además, para el caso que el diseñador defina propiedades de usuario, se debe generar una página de registro de nuevos usuarios, donde se solicita al usuario que ingrese el valor de sus propiedades. Esta página luego podrá ser modificada por el diseñador para adaptarse a los requerimientos gráficos del sitio.

3.2. Arquitectura del Sistema

3.2.1 Diagrama de contexto



Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

3.2.2 Entidades externas

Diseñador

Es la persona encargada del diseño de las páginas del sitio. Esto incluye tanto el diseño gráfico de cada página como la construcción de la estructura general del sitio. (Puede tratarse de más de un individuo)

Responsable del sitio

Es quien se encarga de que el sitio este disponible y cumpla los objetivos buscados, así como de su administración. Su tarea no es hacer el sitio sino mantenerlo funcional. (Un individuo puede ser diseñador y responsable)

Usuario Navegador

Son aquellas personas que utilizando un navegador de Internet (por ej.: Internet Explorer, Netscape Communicator, etc.), acceden al sitio y recorren sus páginas.

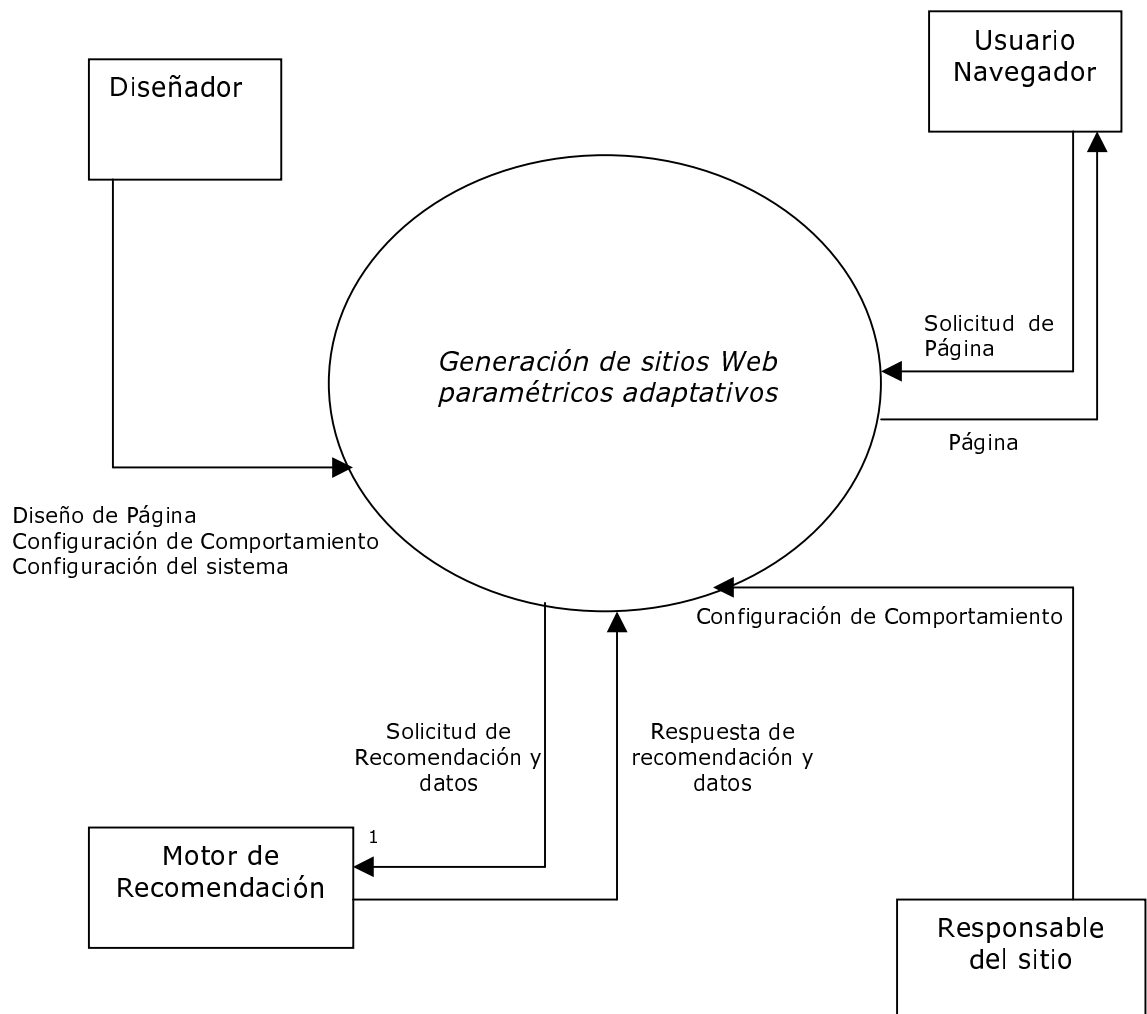
3.2.3 Proceso Sistema

Representamos aquí al sistema como un solo proceso, el que deberá ser dividido luego en los sub - procesos principales del sistema. Antes de analizar como realizar esta división hay que considerar la existencia de un módulo propio del sistema pero que es implementado por el grupo de taller "*Análisis de patrones de búsqueda con Web Mining*" [13]. Consideraremos a este módulo como una entidad externa que presta servicios a nuestro sistema, por lo que lo utilizamos como una caja negra. La tarea central de este módulo será determinar cual es la página que mejor se adecua a un usuario según la configuración de un *Link Dinámico*.

Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

3.2.4 Diagrama de Contexto bajo las condiciones del Taller



3.2.5 Entidad externa "Motor de Recomendación"

Es un proceso, que consideramos como caja negra, implementado por el grupo de taller "Análisis de patrones de búsqueda con Web Mining". Este recibe de nuestro sistema solicitudes de recomendación y datos de navegación para cada usuario, y responde a dichas solicitudes.

¹ Producto final grupo de taller "Análisis de patrones de búsqueda con Web Mining".

Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

3.2.6 Proceso Generación de sitios Web paramétricos adaptativos

Al dividir el sistema y considerar la entidad externa “Motor de Recomendación”, el proceso Generación de sitios Web paramétricos adaptativos representa todo el resto del sistema, el cual corresponde a nuestro grupo de taller.

Este proceso ofrece al usuario las funcionalidades principales explicadas en [punto 3.1]: Edición de páginas para incluir Links Dinámicos, Configuración general del sistema, Generación del sitio con comportamiento paramétrico adaptativo.

En la investigación expuesta en [punto 1.5] se determinó que no existe un editor HTML que reúna las características y funcionalidades antes descritas. Por otra parte, el desarrollo de un editor HTML extendido que permita el manejo de *Links Dinámicos* limitaría las posibilidades en cuanto a las prestaciones ofrecidas. Esto es debido a que los editores HTML actuales ofrecen funcionalidades superiores a las que se pueden alcanzar en el marco del taller, especialmente considerando que no es el objetivo central del mismo.

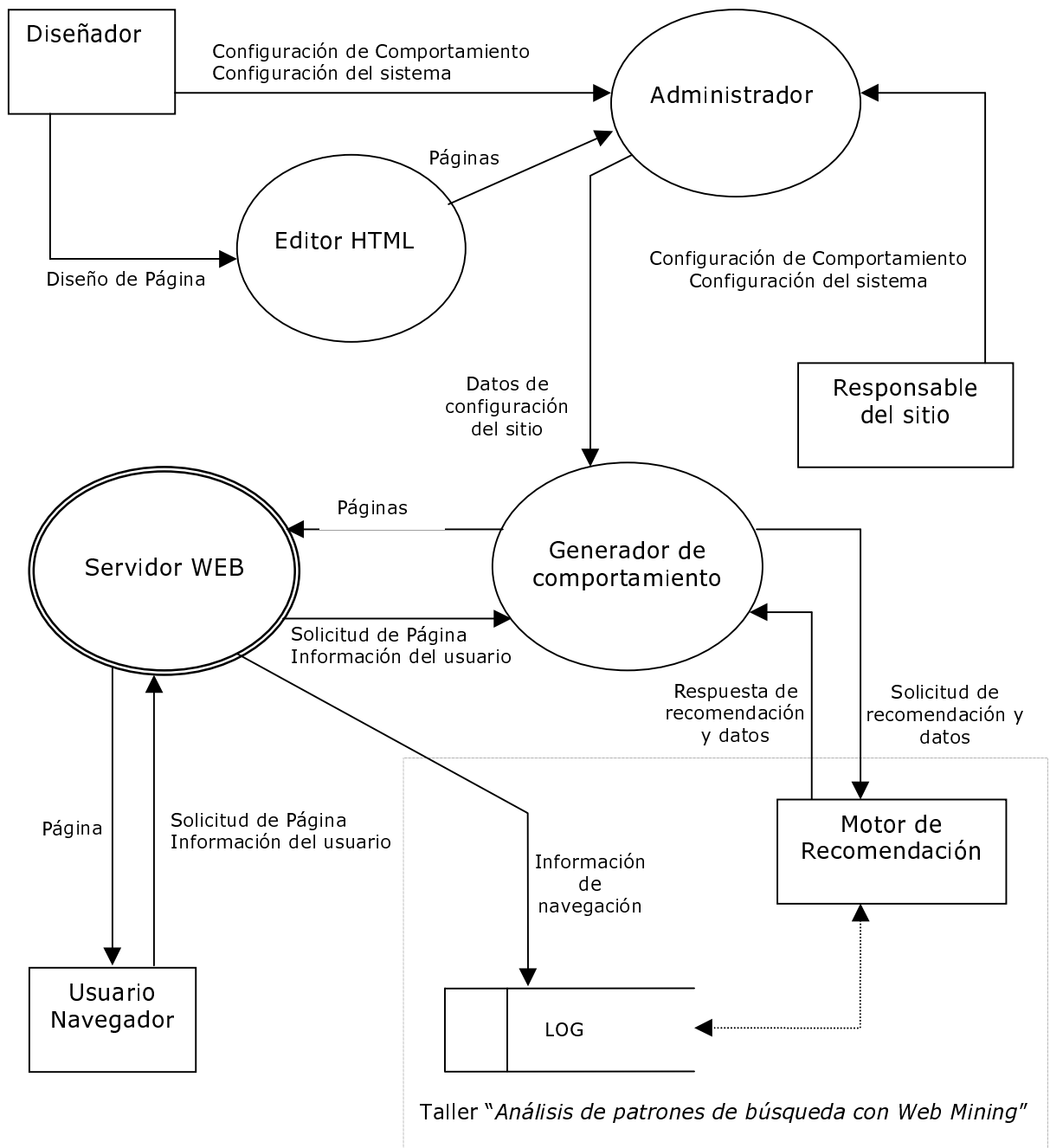
Por lo tanto se tomó la decisión de utilizar cualquier editor HTML estándar para la edición de páginas, y desarrollar una herramienta que permita al editor incorporar los *Links Dinámicos*. En [punto 3.4] se expone una justificación más detallada de esta decisión.

En el punto siguiente se expande el proceso Generación de sitios Web paramétricos adaptativos para reflejar estos conceptos.

Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

3.2.7 Diagrama de Flujo de Datos



Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

3.2.8 Procesos principales

Editor HTML

Es un programa o un conjunto de programas que permiten editar archivos con formato HTML. Puede tratarse de editores de texto o editores específicos para HTML. Estos programas deben además permitir de alguna manera la utilización de *Links Dinámicos* en las páginas que son diseñadas.

Administrador

Es el programa que permite establecer tanto los parámetros del sistema como la configuración del sitio y los *Links Dinámicos*. Algunas de sus funcionalidades sirven en la creación del sitio y otras sirven una vez que el mismo es publicado para cambiar configuraciones.

Generador de Comportamiento

Es el módulo encargado de lograr que los *Links Dinámicos* tengan el comportamiento para el cual fueron configurados.

Servidor

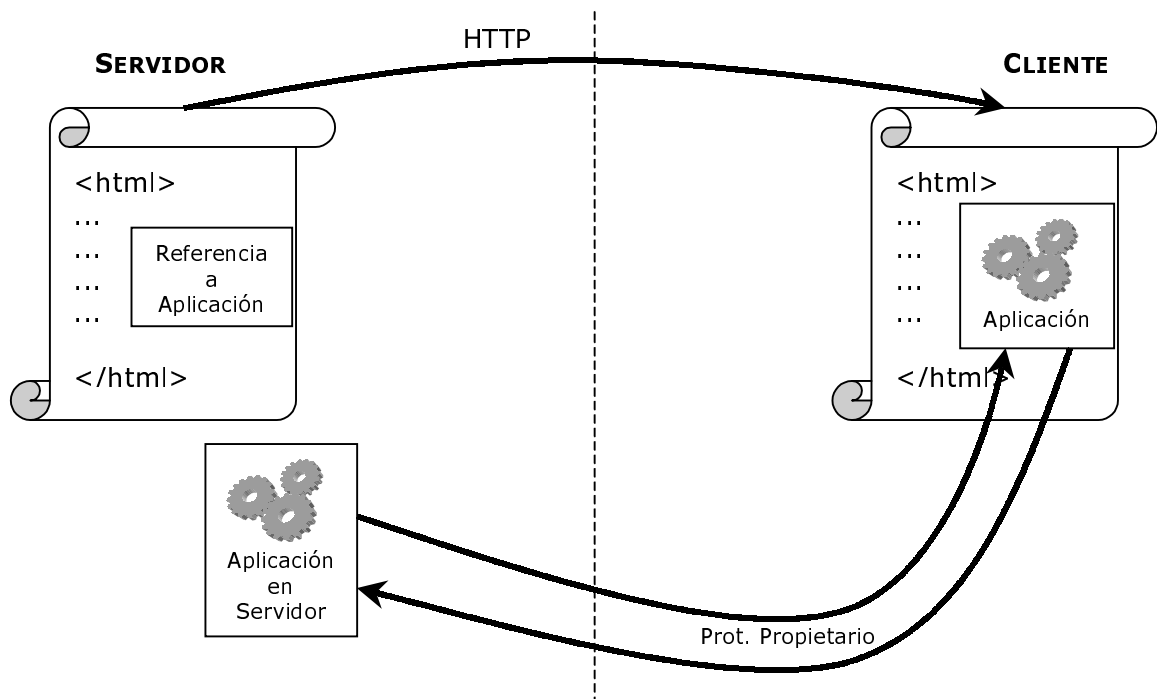
Este es un sistema necesario para la publicación de sitios en Internet que atiende solicitudes de los usuarios navegadores y envía las páginas del sitio a los navegadores que las solicitaron..

3.3. Análisis de la solución

3.3.1 Selección de la forma de resolución de páginas dinámicas

En principio se ven dos grandes métodos de generar el comportamiento dinámico. El primero es introducir en las páginas elementos como ActiveX, Java Applets o lenguajes de Scripts que se ejecutan en el cliente y el segundo es utilizar herramientas como ASP, PHP, PERL, etc. que nos permitan generar las páginas en forma dinámica desde el servidor.

Primer método

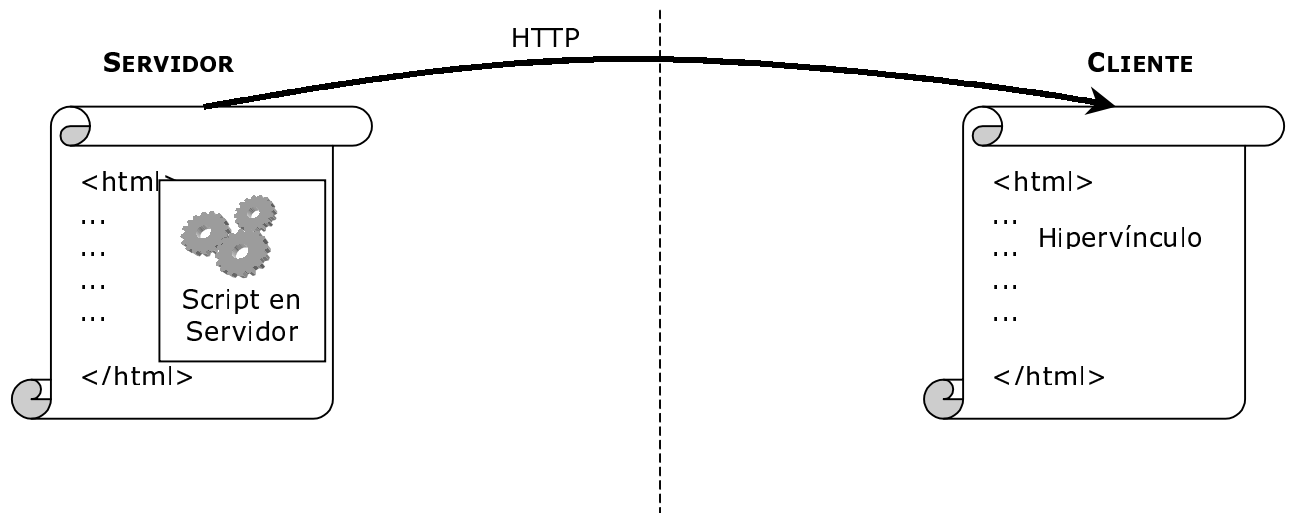


El primer método implica la ejecución de código en el cliente, es decir el navegador. Dicho código es el encargado de realizar los cambios en la página *Web* en base a información de navegación del usuario, para generar el comportamiento dinámico. Por lo tanto, se necesita información almacenada en el servidor y debería establecerse una comunicación entre el navegador y el servidor para obtenerla.

Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

Segundo método



El segundo método independiza totalmente al navegador ya que este no puede determinar que la página mostrada es dinámica. Con esta solución el comportamiento dinámico se logra mediante un programa que se ejecuta en el servidor, y por lo tanto puede acceder a la información contenida en él sin la necesidad de implementar interfaces de comunicación. De esta manera se minimiza el tráfico de datos entre servidor y cliente, ya que evita el pasaje de objetos (ActiveX, Applets, etc.) y la comunicación de éste con el servidor. Las herramientas mencionadas en esta opción (ASP, PHP, etc.) generan código HTML al ser solicitada la página, que a efectos del navegador es lo mismo que si se le enviase código estático HTML.

Entendemos que esta última opción presenta ventajas sobre la primera, por lo tanto optamos por ella. Tomada la decisión de utilizar un lenguaje Server-Side, resta determinar cual. Luego de analizar diferentes opciones, llegamos a la conclusión de utilizar el lenguaje PHP. En el Apéndice A detallamos el motivo de dicha elección.

3.3.2 Método de resolución de páginas dinámicas

Para implementar la solución utilizando un lenguaje Server-Side debemos embeber en las páginas *Web* un algoritmo que genere el HTML correspondiente a cada *Link Dinámico*. Para esto en primer lugar se debe disponer de las páginas HTML del sitio para luego poder embeber el código del algoritmo antes mencionado.

Por lo tanto, antes de introducir los *Links Dinámicos*, el diseñador debe crear páginas *Web* con algún editor (HTML o texto). Se le proveerá al diseñador de herramientas con las cuales podrá luego, insertar y configurar los *Links Dinámicos* en las distintas páginas del sitio.

```
<!--
    Página original con un comentario en la ubicación del objeto dinámico
-->

<html>
    <head>
        Página Web
    </head>

    <body>
        Página Web dinámica.
        <-- Aquí pongo un Link Dinámico -->
    </body>
</html>
```

Para insertar un *Link Dinámico* en una página, tenemos dos posibilidades. La primera es embeber directamente el código PHP que procesa al *Link Dinámico*. Otra opción es insertar un *tag* (a definir), que luego de un proceso de traducción se sustituya por el mismo código de PHP que de la primera opción.

En principio la opción más sencilla parece ser la primera, pero los editores de HTML no pueden interpretar código PHP embebido. Por lo tanto, para que un editor de HTML logre mostrar y permita editar las páginas una vez introducido algún *Link Dinámico*, debemos insertar un *tag* que los editores puedan interpretar.

Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

```
<!--
    Página con un NUEVO tag (DYNAMIX),
    para marcar la ubicación del objeto dinámico
-->

<html>
  <head>
    Página Web
  </head>

  <body>
    Página Web dinamica.
    <DYNAMIX objetoCod="2" proyecto="prueba" ... >
    </DYNAMIX>
  </body>
</html>
```

Como el *tag* debe ser interpretado por un editor de HTML, no se puede definir uno nuevo, se debe utilizar uno existente. Este *tag* además de poder ser interpretado por el editor, debe distinguirse de cualquier otro *tag* que pudiera haber en la página. Después de estudiar el estándar HTML encontramos que el *tag* <Object> reúne las características deseadas. El *tag* <Object> fue originalmente concebido para poder incluir ActiveX en páginas HTML; luego se comenzó a utilizar para incluir Java Applets. Para distinguir un ActiveX o Java Applet de otro, el *tag* incluye un atributo llamado "*ClassId*" que es único. Actualmente, muchos editores de HTML admiten la posibilidad de incluir ActiveX y Java Applets utilizando este *tag*.

También es de interés que desde el editor se pueda acceder a la configuración de los *Links Dinámicos*. Para lograr esto, desarrollamos un control ActiveX con una pantalla de propiedades que permite configurar el *Link Dinámico*. A este ActiveX le llamamos **DynamiX**.

Utilizamos el código HTML generado por los editores HTML al insertar DynamiX como el *tag* que identifica un *Link Dinámico* que es generado por los editores al insertar el ActiveX como el *tag* de *Link Dinámico*. De este modo es posible ver *Links Dinámicos* en editores HTML como una instancia de DynamiX.

Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

Así, se logró definir un *tag* que es interpretado por los editores de HTML, que se puede distinguir de cualquier otro *tag* de la página y para el cual se puede invocar un programa de configuración (página de propiedades¹ de ActiveX) desde editores que provean dicha funcionalidad². Hay que notar que DynamiX no aparecerá en las páginas publicadas, ya que será sustituido luego, simplemente es utilizado para poder acceder a los *Links Dinámicos* desde un editor HTML.

```
<!--
    Página con un tag OBJECT , para marcar la ubicación del Link Dinámico
-->

<html>
  <head>
    Página Web
  </head>

  <body>
    Página Web dinamica.
    <object
      classid="clsid:1396E47C-84E9-11D5-BB07-0BA86B50B"
      id="Link1"
      width="153"
      height="177"

      >
        <param name="_ExtentX" value="4048">
        <param name="_ExtentY" value="4683">
        <param name="ObjetoCod" value="2">
      </object>
    </body>
  </html>
```

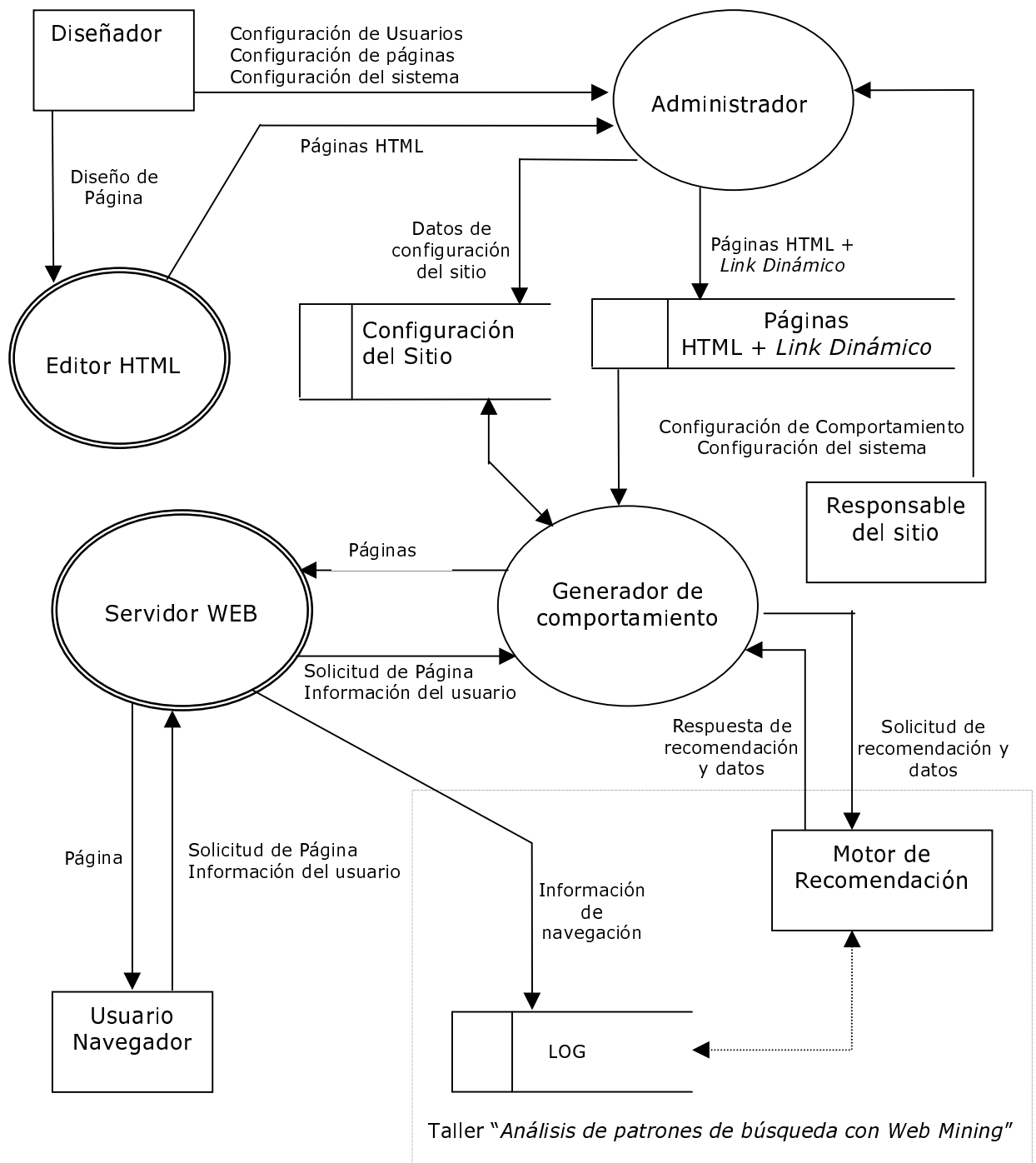
Una vez definido el *tag*, resta definir qué información debe contener. Se podría incluir la información de configuración del *Link Dinámico* en el *tag* pero optamos por asignar un identificador único a cada *Link Dinámico* y almacenar en la base de datos la configuración del mismo. Esto simplifica el manejo de *tags* y permite la modificación de la configuración sin tener que regenerar el sitio.

¹ Los Active X pueden incluir páginas de propiedades. Estos son formularios que permiten asignar valores a las propiedades del mismo. Desde estas páginas de propiedades se pueden además ejecutar acciones como acceder a una base de datos.

² Hemos comprobado que Microsoft FrontPage 2000 incluye dicha funcionalidad

3.4. Arquitectura de la solución

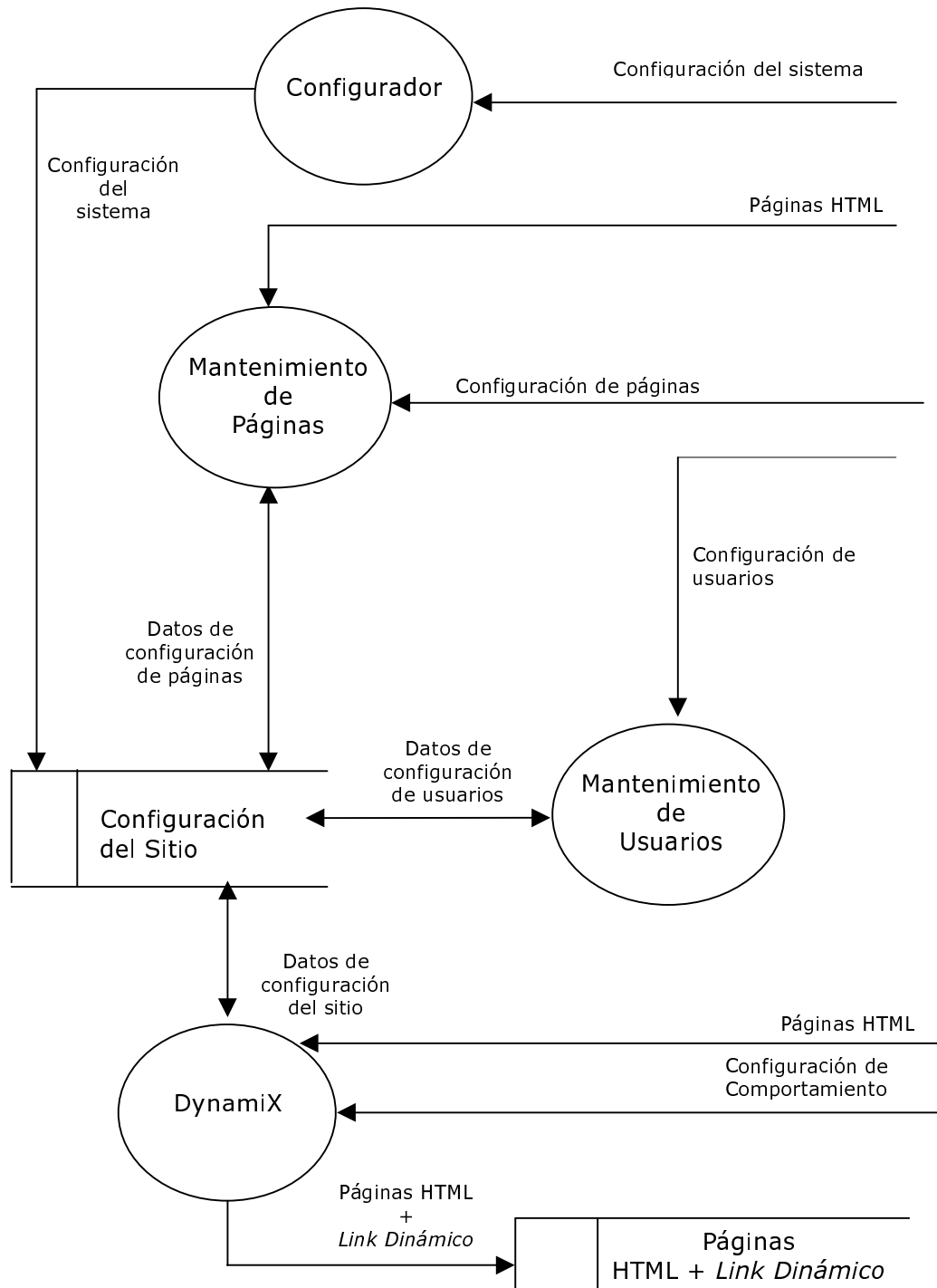
Después de realizado el análisis de la solución, llegamos a la arquitectura final que exponemos a continuación.



Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

3.4.1 Expansión del administrador



Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

A continuación detallamos los procesos y flujos de datos producto de la expansión del proceso del administrador.

Configurador

Mediante este proceso es posible establecer la configuración del sistema. Esta incluye datos como directorios por defecto, preferencias de usuario, etc. El sistema debe permitir la creación de varios sitios *Web*, por lo tanto es posible la creación de *Proyectos de sitios Web* desde este proceso.

Los *Proyectos de sitios Web* son el conjunto de datos, configuraciones del sistema y datos de generación de comportamiento dinámico, característicos de cada sitio *Web*.

Mantenimiento de Páginas

Este proceso es el encargado de manejar la información de las páginas del sitio. Aquí se determina que páginas lo conforman, y se define una categorización de las páginas en grupos. También es el encargado de mantener la jerarquía de grupos de páginas.

Mantenimiento de Usuarios

Este proceso define los parámetros de usuario del sitio. Además se encarga del manejo de grupos de usuario.

DynamiX

Se trata del proceso previamente definido en 3.3.2 encargado del manejo de la configuración de cada *Link Dinámico*.

Configuración del Sitio

Luego del análisis del sistema, la arquitectura de la solución agrega dos medios de almacenamiento destinados a guardar datos del sistema en general. Estos aparecen en el diagrama general de la arquitectura de la solución y particularmente en la expansión del Administrador y la expansión del Generador de comportamiento. Se trata del almacenamiento que describimos en este ítem, Configuración del Sitio, y del que describiremos en el siguiente ítem, Páginas HTML + *Link Dinámico*.

La Configuración del Sitio es el almacenamiento donde se guardan los datos de los *Proyectos de sitios Web*, que como describimos antes, incluyen datos de configuraciones del sistema, datos de generación de comportamiento dinámico, etc. Por este motivo, el Administrador y el Generador de Comportamiento poseen este almacenamiento en común.

Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

Páginas HTML + *Link Dinámico*

Luego de editadas las páginas con DynamiX, éstas se encuentran en un estado intermedio representado en un medio de almacenamiento hasta que sean procesadas por el Generador de Comportamiento.

3.4.2 Editor HTML

Este proceso aparece diferente que en el análisis del sistema original donde lo consideramos como parte del sistema a desarrollar. En este diagrama de arquitectura final, se representa como un proceso interno pero no a desarrollar como parte del proyecto. Esto se debe a que la edición y diseño de páginas es un proceso clave e interno a nuestro sistema, pero debido a la complejidad de desarrollo que involucra la implementación de un editor se decidió la utilización de editores de terceros.

Debemos entonces adaptar editores para permitir la inserción de *Links Dinámicos*. Encontramos tres maneras de hacer esto:

- Modificar el código un editor OPEN-SOURCE (de código abierto).
- Crear un PLUG-IN para un editor en particular.
- Crear un objeto programado que se pueda insertar en una página con varios editores HTML.

La primera opción presenta el problema de depender de una versión del editor, por lo tanto, ante un cambio de versión del editor, sería necesario readaptar y recompilar el editor para mantener las nuevas funcionalidades. Además el costo de adaptarse al estilo de programación de este editor podría ser alto.

La segunda opción considerada, si bien no tiene la dificultad de la adaptación de versiones, presenta el problema del compromiso con un único editor. Además, requiere demasiado tiempo de investigación y entrenamiento acerca de cómo programar una aplicación capaz de interactuar con un editor (PLUG-IN); sin la certeza de que efectivamente esto sea posible.

Finalmente optamos por la tercera opción. Crear componentes programados que puedan insertarse en páginas con editores existentes. Para la inserción de componentes programados, existen en el mercado dos tecnologías predominantes, ActiveX y Java Applets. La decisión entre ambas recayó sobre la experiencia de programación y el conocimiento de las tecnologías, que en este caso era mayor para los componentes ActiveX. Además recordamos que preferimos un componente que cuente con alguna aplicación de configuración programable, que en el caso de los ActiveX son las páginas de propiedades.

Esta decisión limita las posibilidades de interacción de algunos editores que no soportan la inserción de objetos ActiveX, en general los más sencillos, sin

Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

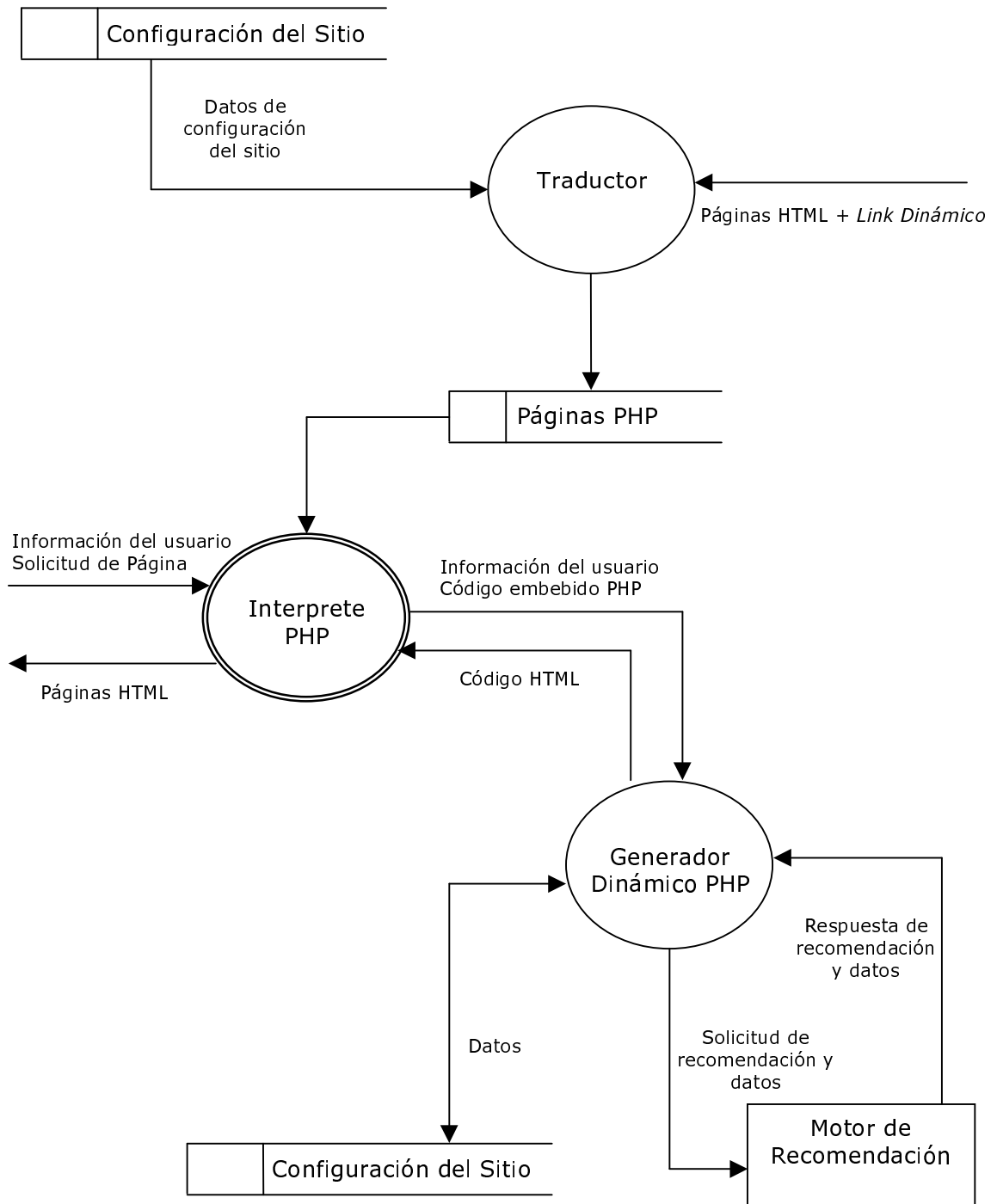
embargo algunos de los más completos si tienen soporte. Además la portabilidad de la plataforma de edición de Páginas no es un prerequisite

La utilización de ActiveX no implica que el sitio solo pueda ser accedido desde navegadores que soporten ActiveX ya que en el proceso de traducción se quitan los ActiveX y se traducen por invocaciones a procedimientos PHP. Es más, podría haberse tomado la decisión de que no se puedan incluir ni modificar *Links Dinámicos* con el editor y permitir configurar los *Links Dinámicos* únicamente desde el Administrador.

Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

3.4.3 Expansión del generador de comportamiento



Generación de sitios Web paramétricos adaptativos

Especificación del Análisis

En este diagrama se expanden todos los procesos del Generador de Comportamiento, a continuación explicamos cada uno de ellos.

Traductor

El traductor es el proceso que genera páginas PHP en base a las páginas HTML que incluyen el *Link Dinámico*.

Interprete PHP

Se trata de una aplicación del tipo "Plug – In" que es utilizada por el servidor *Web* para interpretar las páginas PHP. Como resultado, la aplicación genera la página HTML que el servidor envía y es mostrada al usuario. Para la generación del comportamiento se utiliza un proceso común a todas las páginas PHP, que está codificado también en PHP, y por lo tanto es interpretado por esta aplicación.

Generador Dinámico PHP

Es el proceso descrito anteriormente. Este proceso lee la configuración del *Link Dinámico*, y es el encargado de comunicarse con el Motor de Recomendación mediante una interfaz que definiremos más adelante. Con estos datos genera el comportamiento dinámico correspondiente a la configuración del *Link Dinámico* en código HTML.

4. Diseño del Sistema

4.1. Actores involucrados en el sistema

- Usuarios del sitio
- Administradores del sitio
- Diseñadores del sitio

4.1.1 Usuarios del sitio

Se trata de las personas que acceden a las páginas del sitio con un navegador. La acción que realizan estos pues es navegar por el sitio. En principio cualquier persona con acceso al sitio es un usuario, se puede distinguir dentro de este grupo de actores, el subgrupo constituido por los usuarios registrados al sitio.

4.1.2 Administradores del sitio

Es aquella o aquellas personas que se encargan de mantener al sitio funcional. Son en definitiva los responsables de que el sitio cumpla con su objetivo.

4.1.3 Diseñadores del sitio

Se trata de quien (o quienes) crean las páginas del sitio y su estructura. Se trata de técnicos con conocimientos de diseño gráfico y un mínimo de tecnologías de Internet.

4.2. Casos de Uso Generales

Casos de uso del Diseñador

- Agregar una página al sitio
- Quitar una página del sitio
- Diseñar una página
- Agregar un Link Dinámico a una página
- Quitar un Link Dinámico de una página

Casos de uso del Administrador y Diseñador

- Modificar la configuración de un *Link Dinámico*.
- Modificar los parámetros de configuración de la aplicación.
- Agregar un parámetro de usuario
- Quitar un parámetro de usuario
- Crear un grupo de páginas
- Eliminar un grupo de páginas
- Agregar una página a un grupo de páginas
- Quitar una página a un grupo de páginas

Casos de uso de los Usuarios

- Ingresar parámetros
- Navegar por el sitio

4.3. Conceptos del diseño de la solución

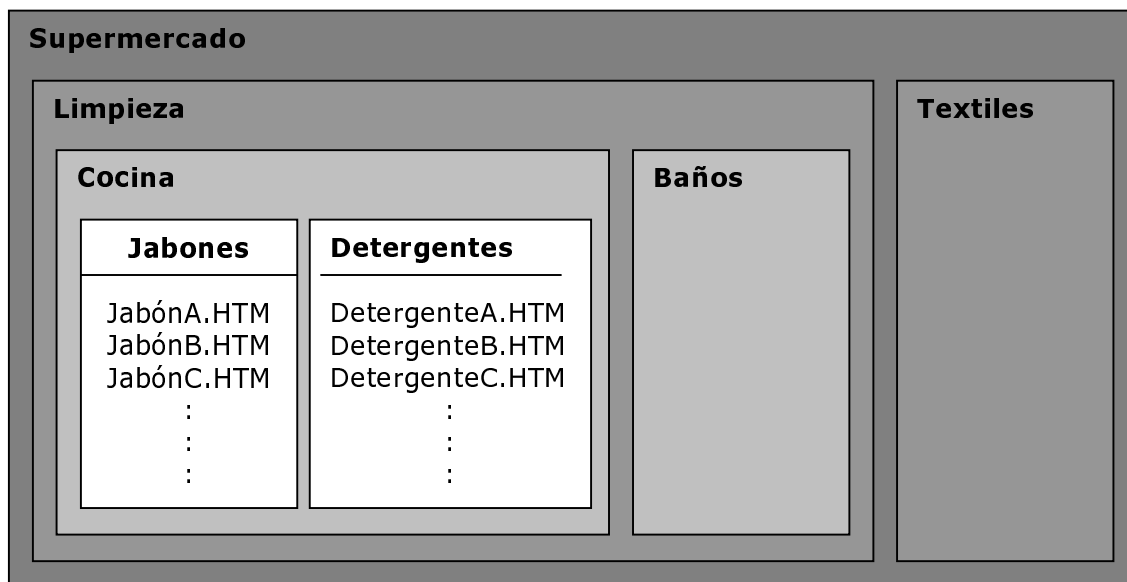
Al especificar los actores del sistema y sus casos de uso generales, comienzan a surgir conceptos de diseño que son imprescindibles para luego continuar con el diseño final de la solución. Estos conceptos están presentes en la creación de una página *Web* y principalmente en como se logra luego la generación del comportamiento dinámico.

4.3.1 Jerarquización de grupos de páginas

En los sitios se suelen agrupar las páginas en diferentes categorías. Por ejemplo en un supermercado, se pueden identificar páginas de fiambrería, limpieza, textiles, etc.; dentro del grupo de fiambrería, a su vez están las páginas de quesos, embutidos, etc.; entro de la de quesos las de quesos duros, cremosos, etc. Así se a forma un jerarquía, en la que una página pertenece a uno o más grupos que a su vez pueden pertenecer a otros grupos.

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema



Para que el diseñador pueda configurar los Links Dinámicos, es importante que pueda manejar el concepto de grupos de páginas. Este concepto puede ser muy útil en el momento de establecer las reglas que definan el comportamiento de un *Link Dinámico*. Por lo tanto, el sistema considera la existencia de estas estructuras.

4.3.2 Parámetros de Usuarios

Hasta este momento, se ha manejado el concepto de propiedades o parámetros de los usuarios, sin entrar en detalles sobre su significado.

Los usuarios que utilizan el sitio poseen características propias, como ser sexo, edad, nacionalidad, etc. a las que llamamos parámetros de los usuarios. La cantidad de parámetro es infinita, pero a los efectos prácticos se selecciona un conjunto reducido de los mismos, que sea significativo a los efectos de distinguir grupos de usuarios con necesidades específicas que puedan ser atendidos desde el sitio Web.

Hay dos modos de saber cuales son las parámetros de los usuarios, la primera y más obvia es mostrar un formulario donde el usuario pueda ingresarlos directamente. La segunda, es intentar descubrirla en base a su comportamiento en el sitio. Por ejemplo una persona que en el 90% de los accesos a un sitio de deportes accede a alguna página de fútbol, será un aficionado al fútbol y un potencial comprador de productos de fútbol.

El sistema permite a los diseñadores definir una serie de propiedades de los usuarios. Podemos identificar propiedades de dos tipos; propiedades de texto y propiedades enumeradas. Las propiedades de texto sirven para identificar valores arbitrarios, que no es posible o no es tan fácil definir en el momento del diseño, como por ejemplo: "Nombre", "e-mail", "Dirección". Las propiedades

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

enumeradas, en cambio, son más versátiles, y se adaptan mejor a valores que se conocen en el momento del diseño. Estas permiten identificar perfectamente valores de propiedades como "Sexo", "Tipo de música", y en general propiedades que se pueden agrupar en rangos.

Estas propiedades pueden ser utilizadas para luego generar automáticamente una página de registro. La forma de hacer esto es crear una página *Web* que contiene un formulario. En dicho formulario se destina una caja de texto para cada propiedad de tipo texto y un menú desplegable para cada propiedad de tipo enumerado.

Dicha página debe ser la página de presentación para cada usuario nuevo del sitio *Web* por lo tanto se debe tomar en consideración que debe tener un diseño atractivo, y específico del sitio *Web*. Esto es imposible de lograr al generar una página automática, por lo tanto debe ser editada luego por el diseñador. La única limitación en el diseño es que se deben conservar los campos insertados para no perder la funcionalidad de registro.

Por otro lado se encuentra la información que se puede obtener de los usuarios a partir de la navegación. La extracción de esta información puede hacerse en forma explícita o pidiendo al sistema que determine cuál es la mejor opción. Por ejemplo se puede querer saber cuál es la página más vista por una persona, esto es una petición explícita. Por otro lado se puede pedir el mejor hipervínculo para el usuario en base a la historia de los usuarios que tienen el mismo perfil de navegación.

Para satisfacer las primeras es necesario un módulo estadístico, en cambio para satisfacer a las segundas será necesario un módulo más inteligente de Data Mining sobre la navegación, conocido como *Web Mining*. Ese módulo crearía grupos de usuarios llamados Clusters, que determinan el perfil de un usuario sin que este haga más que navegar.

La tarea de implementación de dichos módulos es la correspondiente al grupo de taller "*Análisis de patrones de búsqueda con Web Mining*".

4.3.3 Reglas de definición del comportamiento dinámico

Para configurar un Link Dinámico se debe implementar un sistema de reglas que manipulen su comportamiento. Para definir estas reglas contamos con dos elementos, los parámetros de usuario y los patrones de comportamiento de navegación de los usuarios.

Mediante los parámetros de usuario se pueden crear condiciones para determinar si una regla es aplicable o no. Por ejemplo, puede ser de interés poder crear reglas para usuarios de sexo masculino, menores de 20 años, etc. De esta manera se utiliza los parámetros de usuario para establecer que reglas se aplican a estos.

Un problema que se plantea en este modelo, es que más de una regla puede tener una condición válida simultáneamente para determinado usuario. Por lo tanto debe definirse una forma de determinar cual de estas será la que se aplicará. Esto se soluciona asignando una prioridad a cada regla.

El sistema utilizado de prioridades, condiciones y reglas, esta inspirado en el método que utiliza el lenguaje de programación ADA para implementar la comunicación entre procesos concurrentes. Por este motivo, de aquí en más, llamaremos a este conjunto de reglas, prioridades y condiciones "Sistema de Guardas". A continuación se describe más en detalle como funciona el Sistema de Guardas.

Una guarda es regla de comportamiento compuesta por tres elementos:

1. Una condición sobre los parámetros de usuario.
2. Una prioridad.
3. Una consulta para el motor de recomendación (Regla de comportamiento).

Cada *Link Dinámico* tendrá una lista de guardas, estas estarán compuestas por una condición (referente a los parámetros del usuario), una prioridad y una consulta. Al momento de procesar un *Link Dinámico*, se evalúan todas las condiciones. Seguidamente se realiza un sorteo entre todas las guardas para las cuales la condición fue verdadera. En este sorteo cada guarda tiene como peso su prioridad. Una vez sorteada una guarda, se invoca al Motor de Recomendación con la consulta de la guarda como parámetro. Esta consulta es interpretada por el Motor de Recomendación, definiendo así la interfaz entre ambas aplicaciones. Para el caso de que ninguna guarda evalúe su condición como verdadera, se aplica una consulta especial del *Link Dinámico*.

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

La probabilidad de que una guarda sea sorteada será la siguiente:

$$p_i = \frac{\varphi_i}{\sum_{h=1}^n \varphi_h}$$

Donde p_i representa la probabilidad de que la i-esima guarda de condición válida para el usuario sea sorteada entre las n guardas de condición válida, y φ_i representa el peso (prioridad) asignada a la i-esima guarda.

Número	Prioridad	Condición	Consulta
1	20	Sexo == F && Edad < 30	SELECT BEST(1) FOR...
2	40	Edad >= 18	SELECT URL(1)
3	40	Edad < 18	SELECT URL(3)
:	:	:	:
:	:	:	:

Este sistema nos permite definir criterios para distintos usuarios, aún más, permite definir distintos criterios para los mismos usuarios y con que frecuencia usarlos.

Las condiciones son expresiones lógicas sobre los parámetros de los usuarios, constituyen una herramienta que permite aplicar ciertas consultas únicamente para los usuarios de determinado perfil. Por ejemplo se puede solicitar la página más vista por las mujeres menores de 30, únicamente cuando el usuario es una mujer menor de 30 años. La condición de una guarda no tiene por que ser una condición que defina a un grupo de usuarios predefinido. Por ejemplo es posible solicitar la página más vista de la sección deportes por los menores de 15 años, cuando el usuario es mayor de 35 años (por ejemplo en vísperas del día del niño).

Las consultas son dinámicas y por lo tanto no siempre retornarán la misma página, sin embargo para un mismo usuario es posible que la página retornada por una consulta se mantenga estable por un período prolongado, el diseñador o administrador del sitio puede querer incluir en un *Link Dinámico* varias consultas y que al azar se resuelva alguna de estas de modo que la página varíe de un acceso a otro. Esto es posible mediante este sistema ya que si más de una condición es válida para el usuario, se realiza un sorteo entre todas estas guardas y se resolverá la consulta de la guarda ganadora. Incluso el diseñador puede ponderar cada guarda dándole una prioridad.

Por ejemplo, se quiere que para las personas de más de 1,90 metros de altura se muestre un hipervínculo a una página de artículos para altos en un 90% de los casos y a una página de artículos de basketball en un 10% de los casos. Para lograr esto, se crean dos guardas con la condición "altura > 1,90"; la primera realiza una consulta para obtener la mejor página de artículos para altos, asignándosele prioridad 9. A la segunda, con una consulta para obtener la mejor página de basketball, se le asigna prioridad 1.

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

En el Apéndice F, se exponen detalladamente dos ejemplos sobre el Sistema de Guardas.

Condiciones

Como se explico anteriormente, las condiciones son expresiones lógicas compuestas por constantes, parámetros de usuario, operadores lógicos y la función Visited.

```
CondElem    := Cte | <parámetro de usuario> Op <valor> |  
              (CondElem) Op (CondElem) | Visited (Pagina) Op Valor |  
              Belongs(Id_Grupo) | !(CondElem)  
  
Id_Grupo    := Integer  
  
Pagina       := String  
  
Cte          := True | False  
  
Valor        := Integer  
  
Op           := < | > | >= | <= | ==
```

La función Visited retorna la cantidad de veces que el usuario accedió a una página. Esta función es implementada por el Motor de Recomendación. Visited, es una función que se toma como ejemplo de implementación, en un futuro se podría implementar otro tipo de funciones que obtengan información del *Log*.

La función Belongs indica si el usuario pertenece a determinado grupo de usuarios. De esta manera se evita tener que repetir la condición que define a un grupo. Además en caso de que se modifique la definición de un grupo de usuarios, el cambio se ve impactado automáticamente en todas las reglas.

4.3.4 Lenguaje de Consultas para las reglas de comportamiento.

Hasta aquí hemos manejado el concepto de consulta sin detallar cual es su real alcance y especificación. El presente punto, expone la motivación de la utilización de un sistema de consultas y describe brevemente su especificación.

Las consultas especifican la forma en que se le solicita al Motor de Recomendación la página que se desea para cada regla. Estas están definidas mediante un lenguaje de consulta llamado DQL (Dynamic Query Lenguaje) (Apéndice E).

Cuando el Motor de Recomendación recibe una consulta DQL, la procesa y retorna un identificador de URL, así como un motor de base de datos retorna una tupla al recibir una consulta de selección. En definitiva la consulta es la manera que tiene el administrador o el diseñador de solicitar una página por lógica dinámica.

Otra posibilidad hubiera sido utilizar funciones que reciban los parámetros necesarios sin la utilización de un lenguaje de consultas. La utilización de un lenguaje de consulta, independiza a los sistemas y permite que el Motor de Recomendación aumente sus prestaciones sin impactar en el sistema de administración. Ya que en caso de un aumento en sus prestaciones, lo único que habría que hacer es ampliar el lenguaje, manteniendo la vieja funcionalidad¹.

Una consulta tiene un formato similar a:

```
SELECT [RANK(Ord) | MOST(Ord) | HOT(Ord, Growing, default) | RANDOM]
FROM [ALL | URLGROUP(id_URLGroup)]
FOR [ALLUSERS | THISUSER(Id_User) | THISGROUP(Id_Group)]
```

SELECT, especifica qué operación de selección se desea solicitar.

- RANK(Ord) Retorna la Ord-ésima mejor URL
- MOST(Ord) Retorna la Ord-ésima URL más visitada
- HOT(Ord, Growing, Default) Retorna la Ord-ésima URL de mayor o menor crecimiento (si no hay ninguna retorna la URL Default)
- RANDOM Retorna un URL al azar

FROM, especifica a que conjunto de páginas se refiere la selección

- ALL Todas las páginas
- URLGROUP(id_URLGroup) El grupo de página de identificador id_URLGroup

¹ Esto es similar a lo que sucede cuando una consulta ANSI SQL 1 se aplica a un motor de base de datos ANSI SQL 2, la consulta sigue siendo aceptada y por lo tanto no hay necesidad de modificar los programas al cambiar la versión del motor de base de datos

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

FOR, especifica a que usuarios aplicar la función

- ALLUSERS Aplica a todos los usuarios
- THISUSER(Id_User) Aplica al usuario que esta navegando
- THISGROUP(Id_Group) Aplica al grupo de usuarios de identificador Id_Group

Asistente

Como el sistema está previsto para usuarios no informáticos, el sistema cuenta con un asistente (Wizard) que permite que un usuario arme las consultas sin la necesidad conocer DQL.

Ampliación para condiciones

Como se explica en la sección anterior, a las consultas se le puede agregar la función Visited. El DQL especifica esta interacción con el Motor de Recomendación.

El formato de estas consultas es el siguiente:

```
VISITED URL(Id_URL) | URLGROUP(Id_URLGroup)
FOR THISUSER(Id_User)
```

VISITED, indica si se refiere a la cantidad de accesos a la página de identificador Id_URL o al grupo de páginas Id_URLGroup.

4.3.5 Texto e Imágenes Asociadas a una URL.

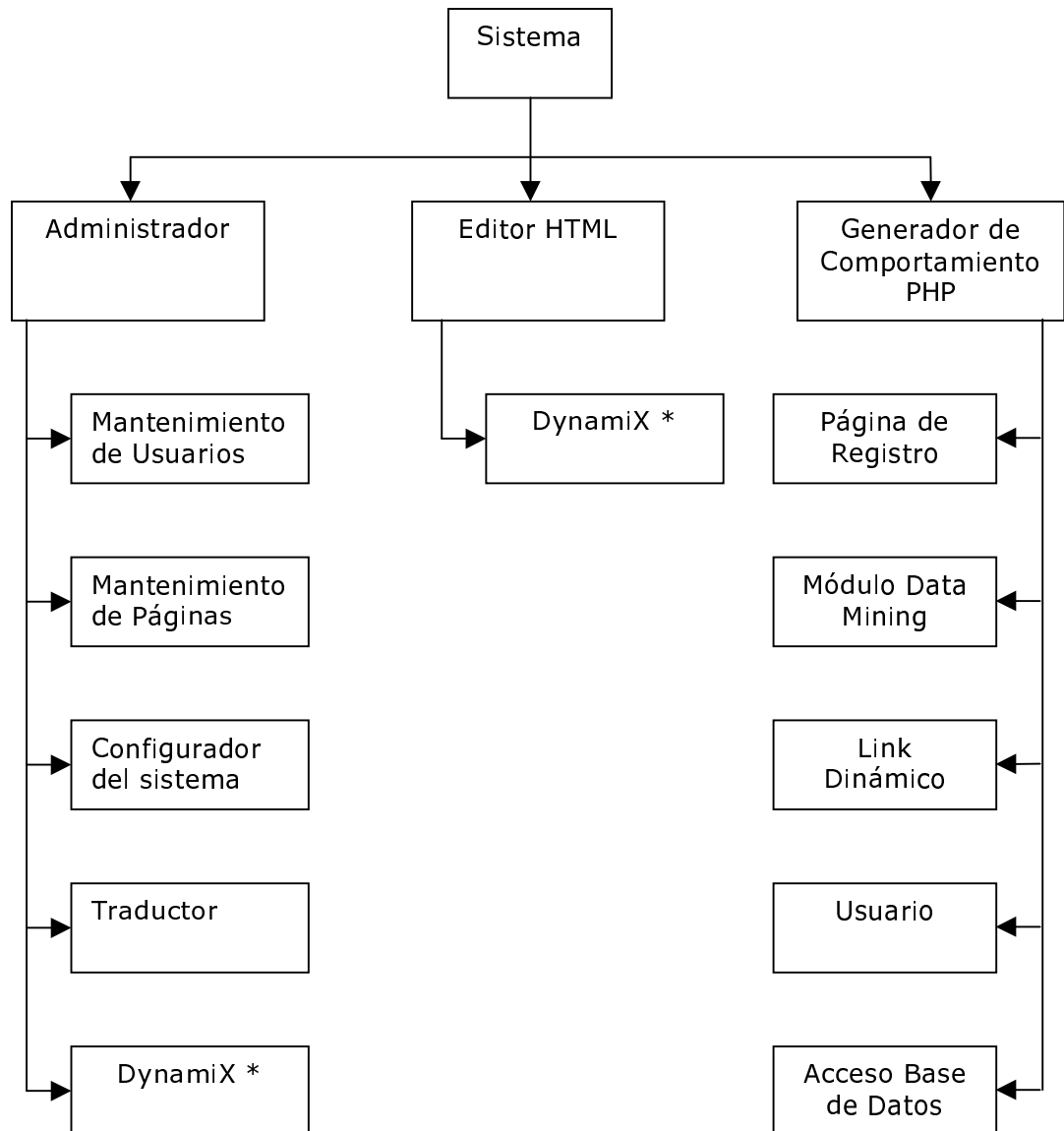
Cuando el diseñador inserta un *Link Dinámico*, no sabe cual será el destino del mismo cuando los usuarios accedan. Esto en algunos casos impide mostrar un texto o una imagen relacionadas con el destino del hipervínculo. Por ejemplo, un sitio de lugares turísticos podría querer que cada hipervínculo mostrará una postal de lugar.

En otros casos el diseñador desea mantener un texto o una imagen fija para un *Link Dinámico*, sin que importe cual es el destino final.

Para resolver este problema se crea una lista de textos e imágenes asociadas a una URL, de este modo, el diseñador puede indicar que se despliegue un texto o imagen para el destino final. En el ejemplo anterior, si el destino es Punta del Este, se accederá a la lista de imágenes de Punta del Este y se mostrará una al azar.

Por lo tanto, como parte de la configuración de un *Link Dinámico*, se debe especificar si el mismo tiene un texto o imagen fijas o si por el contrario, se debe seleccionar un texto o una imagen de la lista asociada del destino seleccionado.

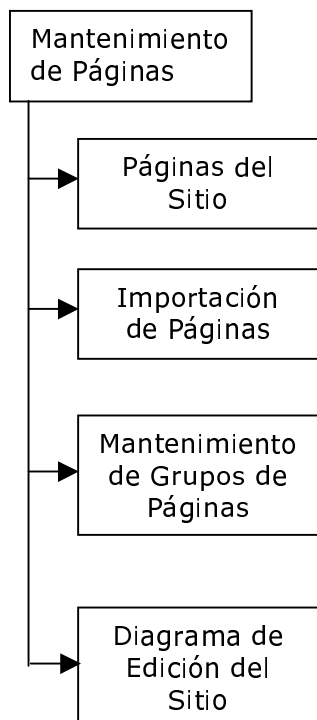
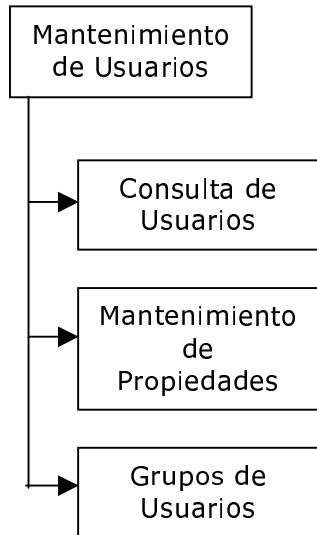
4.4. Módulos del Sistema



* Las dos entidades DynamiX refieren a implementaciones para distintos ambientes del mismo módulo.

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema



4.5. Módulo Administrador

Una vez que el diseñador traduce el sitio y coloca las páginas PHP en el servidor *WEB*, el sitio está listo para ser publicado. Sin embargo una vez que se publica el sitio, debe haber una manera de mantenerlo y administrarlo sin tener que crear el sitio nuevamente con cada modificación.

Para esto se implementó el Administrador. Se trata de un módulo a ser usado justamente por el administrador del sitio, que permite modificar el comportamiento del sitio y permite establecer una serie de parámetros necesarios para poder configurar los *Links Dinámicos*.

Mediante el administrador se puede además insertar *Links Dinámicos* y modificar su configuración utilizando el módulo DynamiX. De esta manera luego que el diseñador crea las páginas estáticas, puede insertar con DynamiX los *Links Dinámicos* en estas páginas. También es posible insertar los *Links Dinámicos* con el módulo DynamiX desde el editor HTML.

4.5.1 Casos de uso del Administrador

Los casos de uso particulares del administrador son los siguientes:

Casos de uso del Mantenimiento de Usuarios

- Consultar Usuarios
- Agregar , quitar o modificar propiedades de usuario
- Agregar , quitar o modificar grupos de usuarios
- Recalcular usuarios de un grupo

Casos de uso del Mantenimiento de Páginas

- Agregar o quitar una página
- Agregar, quitar o modificar un *Link Dinámico* de una página
- Agregar, quitar o modificar grupos de páginas
- Agregar o quitar una página a un grupo de páginas
- Asignar un grupo de páginas como integrante de otro
- Agregar y quitar imágenes o textos asociados a una página

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

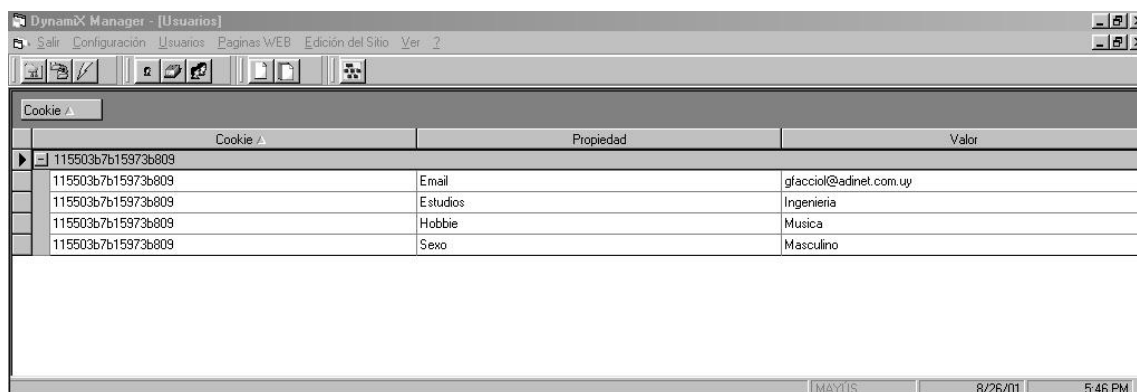
Casos de uso Generales del Administrador

- Importación de un sitio estático existente (para incluir *Links Dinámicos*)
- Traducción ¹
- Visualización gráfica del sitio ²
- Agregar, quitar o configurar proyectos
- Configuración general del proyecto

4.5.2 Mantenimiento de Usuarios

Consulta de usuarios

Despliega una grilla en la que se muestran los usuarios registrados en el sitio. El reporte muestra el identificador de usuario, y el valor de cada propiedad de este. Se permite agrupar, de modo de poder ver la lista de usuarios o los distintos valores de una propiedad.



The screenshot shows a web application window titled "DynamX Manager - [Usuarios]". It has a menu bar with "Salir", "Configuración", "Usuarios", "Paginas WEB", "Edición del Sitio", and "Ver 2". Below the menu is a toolbar with various icons. The main content area displays a table with the following data:

Cookie ^	Propiedad	Valor
115503b7b15973b809		
115503b7b15973b809	Email	glacciol@adinet.com.uy
115503b7b15973b809	Estudios	Ingenieria
115503b7b15973b809	Hobbie	Musica
115503b7b15973b809	Sexo	Masculino

The status bar at the bottom shows "MAYUS", "8/26/01", and "5:46 PM".

Lista de usuarios y sus propiedades

¹ Se pensó que el mejor lugar para incluir el traductor es en el programa de administración, de modo de centralizar la interfase con los responsables del sitio.

² Se trata de una interfaz gráfica que permite ver un grafo del sitio de manera que el administrador pueda tener una visualización del sitio. Además mediante esta interfaz, puede identificar las páginas y editarlas (se llama al editor que seleccione en la configuración general). También puede identificar los objetos dinámicos y editar su configuración modificando el comportamiento de estos sin tener que re-compilar el sitio. (Se implementa en conjunto con el Diagrama de Edición del Sitio)

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

Mantenimiento de propiedades

Permite ingresar, quitar y modificar parámetros de los usuarios. Se despliegan dos grillas relacionadas. En la primera se muestra la lista de propiedades y en la segunda muestra en caso de tratarse de una propiedad enumerada, los valores válidos del enumerado.

The screenshot shows a window titled 'Propiedades' with a table of properties and a section for the 'Sexo' property values.

	Nombre	Tipo
▶	Sexo	Tipo Enumerado
	Estudios	Tipo Enumerado
	Hobbie	Tipo Enumerado
	Email	Tipo Texto
*		

Valores posibles de Sexo

	Texto
▶	Masculino
	Femenino
*	

Mantenimiento de grupos de usuarios

Los grupos de usuario a diferencia de los grupos de páginas, se definen por comprensión. Para esto se introduce una condición sobre las propiedades de usuarios. Este módulo muestra una grilla desde donde se pueden realizar las altas, bajas y modificaciones a los grupos de usuarios.

Además la pantalla incluye un botón que permite generar un grupo de usuarios, esto es, evaluar para cada usuario la condición del grupo e incluir en el grupo aquellos usuarios para los cuales la condición se verificó.

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

Descripción del grupo	Condición de filtro	Fecha de actualización
Hombres	\$Sexo == 1	
Mujeres	\$Sexo == 2	

Generar!

MAYÚS 8/26/01 6:23 PM

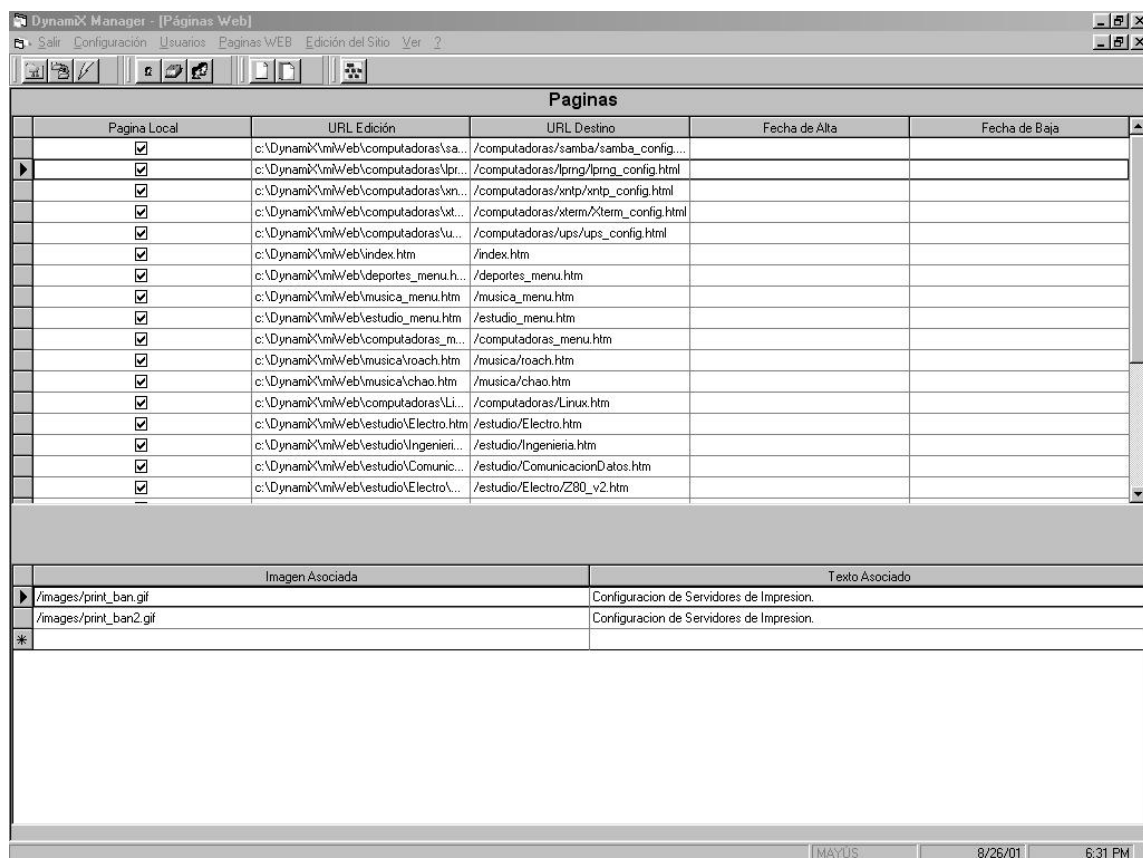
Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

4.5.3 Mantenimiento de Páginas

Páginas del Sitio

Muestra una grilla con las páginas del sitio. Mediante esta grilla se pueden agregar , quitar o modificar las páginas. En la parte inferior de la pantalla se muestra una grilla relacionada con la primera en donde se muestran los textos e imágenes asociadas a la página seleccionada en la grilla principal.



Importación de Páginas

Permite agregar automáticamente las páginas que componen el sitio a partir de los archivos .htm que se encuentren en un directorio origen. Así se evita la carga manual utilizando la funcionalidad expuesta en el punto anterior. Esta funcionalidad es de especial utilidad cuando se parte de un sitio existente sin *Links Dinámicos*, o se actualizan las páginas de un sitio.

Mantenimiento de grupos de páginas

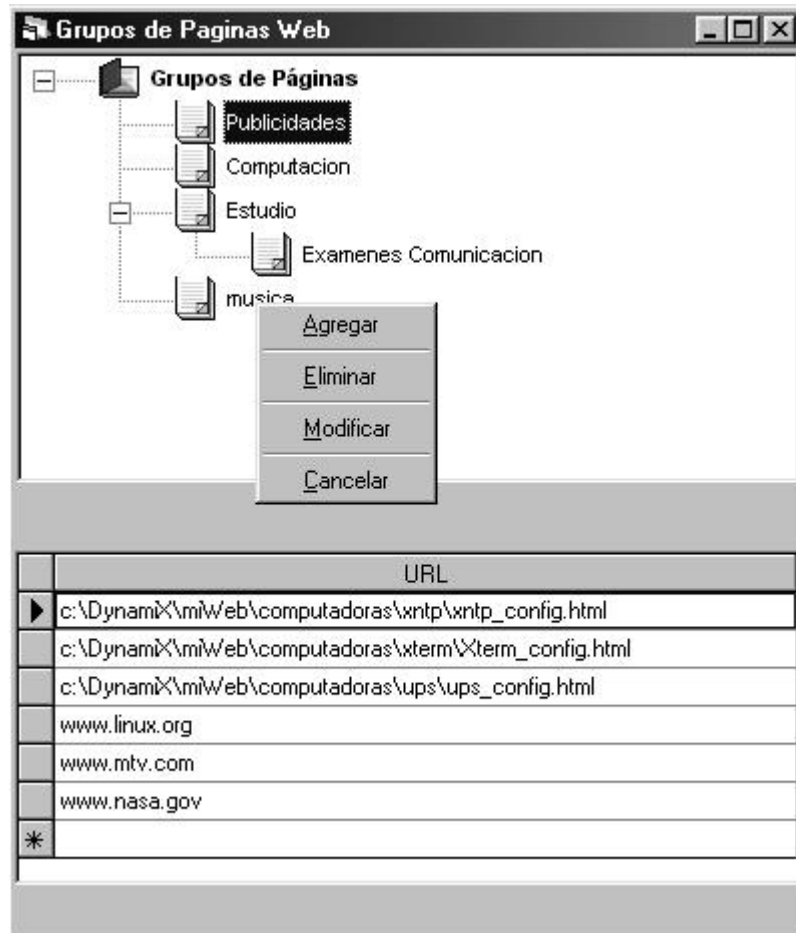
Aquí se muestra una pantalla en donde hay un árbol jerárquico de grupos de páginas. Cada elemento de este árbol es un grupo de páginas. Un grupo que aparezca en el árbol como hijo de otro indica que sus páginas también

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

pertenecen al grupo padre. Por ejemplo, una página del grupo "Exámenes Comunicación" también es una página del grupo "Estudio".

En la parte inferior se encuentra una grilla relacionada con el árbol en donde es posible editar las páginas que pertenecen a un grupo.

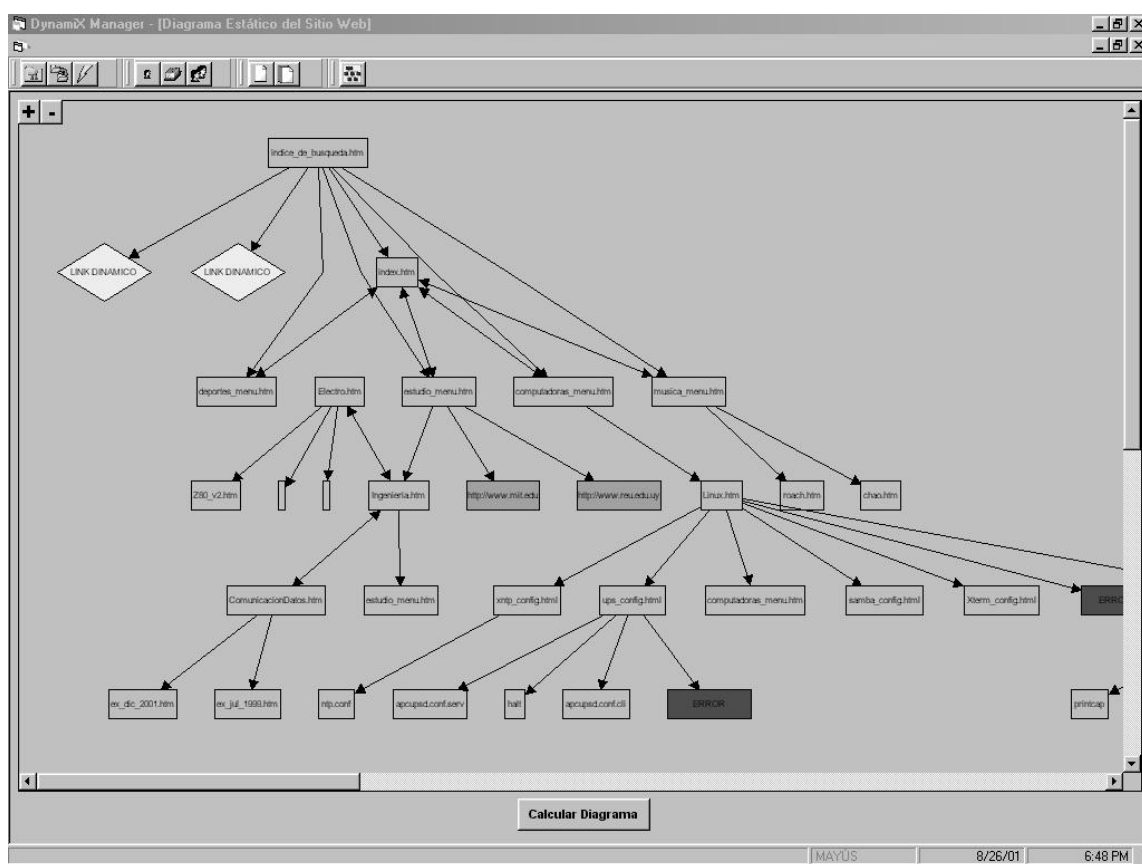


Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

Diagrama de edición del sitio.

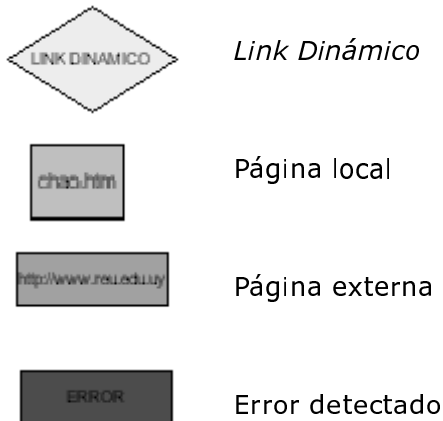
El diagrama de edición del sitio despliega un grafo dirigido donde hay distintos tipos de nodos. Se representan las páginas del sitio como los vértices del grafo. Los hipervínculos entre las páginas son representados por las aristas del grafo. Una arista dirigida de P a Q, indica que en la página P hay un hipervínculo (estático) a Q. También hay vértices que representan *Links Dinámicos*. Una arista dirigida de la página P al *Link Dinámico* D, indica que en la página P hay un *Link Dinámico* D.



Mediante un doble clic sobre un *Link Dinámico*, se despliega el DynamiX. Esta acción sobre una página estática llamará a un editor HTML previamente configurado. En el caso de un hipervínculo externo, se abre el navegador configurado para poder visualizar el destino del mismo.

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema



Leyenda de los símbolos del diagrama de edición del sitio.

4.5.4 Configuración del sistema

Permite modificar los valores de configuración general

Configuración

PROYECTO: demo

ClassId: clsid:1396E47C-84E9-11D5-BB07-0050BA86B50B

Directorio de Origen: c:\Dynamix\miWeb

Directorio de Destino: c:\inetpub\wwwroot

Nombre de Cookie: Galletas

Página de Configuración: /config.htm

Página Principal: /frames.htm

Editor HTML: C:\Archivos de programa\Microsoft Office\Office\FRONTPG.EXE

Explorador Internet: C:\Archivos de programa\Internet Explorer\IEXPLORE.EXE

CONFIGURAR BASE DE DATOS

Agregar **Edición** **Eliminar**

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

4.5.5 Traductor

Una vez que el diseñador crea las páginas del sitio, estas se traducen con el objetivo de convertir las páginas HTML con *Links Dinámicos* insertados (que no pueden ser interpretados por un navegador) en páginas PHP. Para esto lee el contenido de las páginas HTML sustituyendo los *Links Dinámicos* por invocaciones a procedimientos PHP que manejan el comportamiento dinámico. (Implementados en el Generador de Comportamiento PHP)

El código HTML creado por el editor contendrá un *tag* similar al siguiente:

```
<object classid="clsid:58DA8D8A-9D6A-101B-AFC0-4210102A8DA7" id="Dinamico">
  <param name="Numero_De_Objeto " value="2646">
</object>
```

El "classid" es un número único correspondiente a la implementación del módulo DynamiX como ActiveX. Cuando se inserta un *Link Dinámico*, se crea un registro en una tabla de *Links Dinámicos* y se almacena la configuración del mismo en una base de datos. El parámetro "Numero_De_Objeto" se refiere al registro en la base de datos que fue creado para este *Links Dinámico*. Notar que en el *tag* no se guardan los valores para las propiedades de configuración de un *Link Dinámico*, simplemente se guarda el identificador interno del mismo (parámetro Numero_De_Objeto). La configuración se almacena en la base de datos de modo que con el programa Administrador se puedan cambiar la configuración y por lo tanto modificar el comportamiento sin la necesidad de volver a traducir el sitio.

Cuando el traductor encuentre el *tag*, lo sustituirá por una llamada a un procedimiento PHP que procese el comportamiento dinámico.

En el ejemplo de código anterior aparece un *Link Dinámico* con identificador 2646, por lo tanto se sustituirá por una llamada como la siguiente:

```
ProcesarDinamico(2646)
```

El lenguaje PHP incluye al HTML por lo tanto el resto de la página se copiará exactamente igual que como esta en el HTML.

Ejemplo:

Código HTML original:

```
Línea 1
Línea 2
<object classid="clsid:58DA8D8A-9D6A-101B-AFC0-4210102A8DA7" id="Dinamico">
  <param name="Numero_De_Objeto " value="2646">
</object>
Línea 3
```

Código PHP generado:

```
Línea 1
```

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

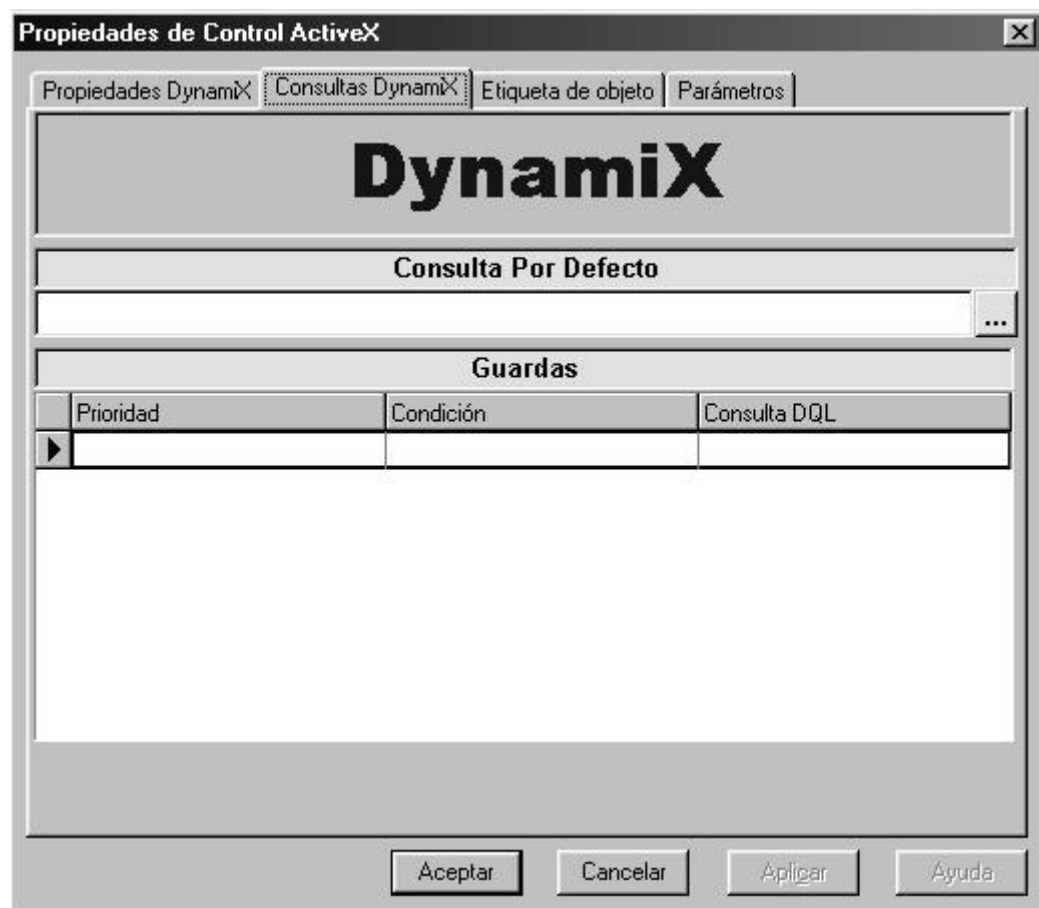
Línea 2

```
<? ProcesarDinamico(2646); ?>1
```

Línea 3

4.5.6 DynamiX

Este módulo implementa el mantenimiento de *Links Dinámicos*. Existen dos implementaciones, una como una parte del administrador y otra como un ActiveX para ser insertado desde un editor HTML. Ambas implementaciones tienen las mismas funcionalidades, pero fueron desarrolladas para diferentes entornos de aplicación.



Vista de DynamiX en su implementación como ActiveX

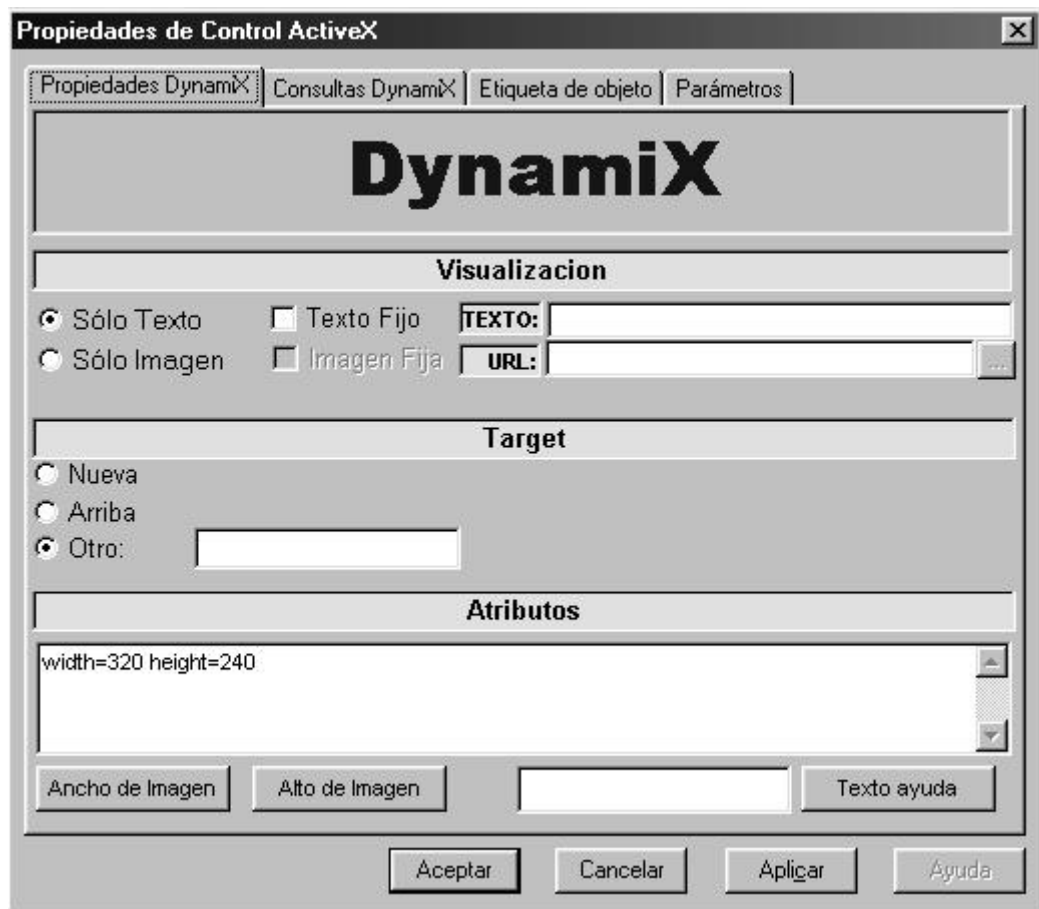
¹ Los tags `<? ... ?>` indican código PHP embebido.

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

En el caso de la implementación ActiveX, encontramos que algunos editores de HTML no permiten acceder a las pantallas personalizadas de propiedades de los ActiveX. Estos editores pueden ser utilizados, pero la configuración de los *Links Dinámicos* se deberá realizar desde el programa administrador.

Configuración de un *Link Dinámico*



Página de propiedades del *Link Dinámico*

En primer lugar hay que indicar si el *Link Dinámico* será un texto o una imagen. Una vez determinado esto hay que indicar si el texto o la imagen serán fijas o si variarán de acuerdo a la página a la que refiera el *Link Dinámico*.

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

Cada página del sitio puede tener una lista de imágenes y textos asociados, si no se elige la opción de texto o imagen fija, al evaluarse en tiempo de navegación un *Link Dinámico*, se accederá a la lista de textos e imágenes de la página seleccionada y mediante un sorteo ponderado por la importancia de cada imagen o texto se determinará la imagen o texto a mostrar.

Cuando se utilizan páginas con marcos se puede incluir el TARGET del DynamiX. Se puede indicar uno en particular, pedir que utilice uno nuevo o que sobre escriba el actual.

Además se puede incluir una ayuda TIP, que se mostrará al pasar el mouse sobre el Link.

Estos parámetros hacen a la visualización del *Link Dinámico* pero no a su comportamiento. Para esto se utiliza el sistema de guardas antes descrito que se edita en la segunda solapa.

4.6. Módulo Editor HTML

La implantación de un sitio mediante nuestro sistema comienza al igual que en el caso de un sitio estático por crear las páginas que lo conformarán mediante un editor de páginas HTML. Mediante este editor se podrán insertar *Links Dinámicos* los cuales se podrán configurar para tener distinto tipo de respuesta. Para definir esta configuración el diseñador deberá cargar una serie de propiedades y armar una lista de guardas que definan el comportamiento. Esta información que el diseñador brinda al sistema, se almacena en una base de datos, a la que se accede vía ODBC. Para poder definir la configuración se contará con una página de propiedades de los *Links Dinámicos*.

4.5.1 DynamiX

Cuando se inserta el ActiveX aparecerá en pantalla y el diseñador podrá modificar su posición y tamaño como el de cualquier elemento. Cuando accede a la ventana de propiedades, aparecerá una pantalla donde el usuario puede ingresar los datos de configuración del *Link Dinámico*.

4.7. Módulo Generador de Comportamiento PHP

Como se vio en la sección anterior, el manejo del comportamiento dinámico queda a cargo de procedimientos generales implementados en PHP. A continuación haremos una pequeña descripción. Para obtener el pseudo-código completo, referirse a [anexo pseudo código PHP].

El procedimiento recibe como parámetro un número interno de *Link Dinámico* y genera código HTML para mostrar el objeto según su configuración y las características del usuario.

Los pasos que sigue este procedimiento son:

El procedimiento en primer lugar accede a la lista de guardas del objeto. Obtiene luego los valores de las propiedades del usuario. Para cada guarda evalúa la condición según las propiedades de los usuarios. Realiza un sorteo ponderado por el peso, entre las guardas de condición válida, y obtiene su consulta.

Si ninguna guarda tuviese condición válida, toma la consulta por defecto. Llama al motor de recomendación con la consulta, obteniendo así una página. Si el *Link Dinámico* tiene texto o imagen fijas toma el texto o la imagen indicadas en la configuración del *Link Dinámico*.

Si el objeto tiene texto o imagen variables según la página recomendada, accede a la tabla de textos e imágenes asociadas y realiza un sorteo ponderado por el peso, entre las tuplas correspondientes a la página. Obteniendo así el texto o la imagen. Con el texto o la imagen obtenida y la página recomendada, inserta el hipervínculo.

4.6.1 Página de Registro

La página de registro es el modulo PHP que se encarga de procesar los datos del formulario de registro. Cuando el usuario termina de llenar el formulario y presiona "Registrarse" los datos son ingresados en la base de datos y asociados a la cookie de seguimiento¹. De esta forma cada vez que el usuario ingrese al sitio, se podrá identificar perfectamente.

4.6.2 Modulo Data Mining

El modulo de Data Mining es una interfaz de comunicación entre el Generador de Comportamiento PHP y el módulo Motor de Recomendación. La comunicación entre estas aplicaciones se lleva a cabo mediante un protocolo de comunicación especificado en el Apéndice C.

Al separar la interfaz entre aplicaciones en un modulo, se hace transparente el medio de comunicación con el Motor de Recomendación y nos independiza del mismo.

4.6.3 Usuario

El módulo Usuario, es la representación interna de los datos del usuario dentro del sistema. Se trata de una forma más de abstracción de datos para independizar la aplicación del medio de obtención de información sobre el usuario.

Además este módulo es capaz de evaluar las Condiciones que componen las Guardas, aplicadas sobre las propiedades del usuario para determinar la validez de la Guarda. Para mas detalles sobre este tema referirse al Apéndice B.

4.6.4 Acceso a Base de Datos

Al igual que en el punto anterior, nos queremos abstraer del mecanismo de acceso a datos, de esta manera este modulo implementa interfaces uniformes para acceder a los distintos datos necesarios para procesar el *Link Dinámico*.

4.6.5 Link Dinámico

Este módulo es el encargado de integrar todas las funcionalidades de los módulos anteriores, para lograr el comportamiento dinámico dentro de las paginas traducidas. La forma de interacción con los otros módulos y una explicación detallada de su funcionamiento se encuentra en el Apéndice B.

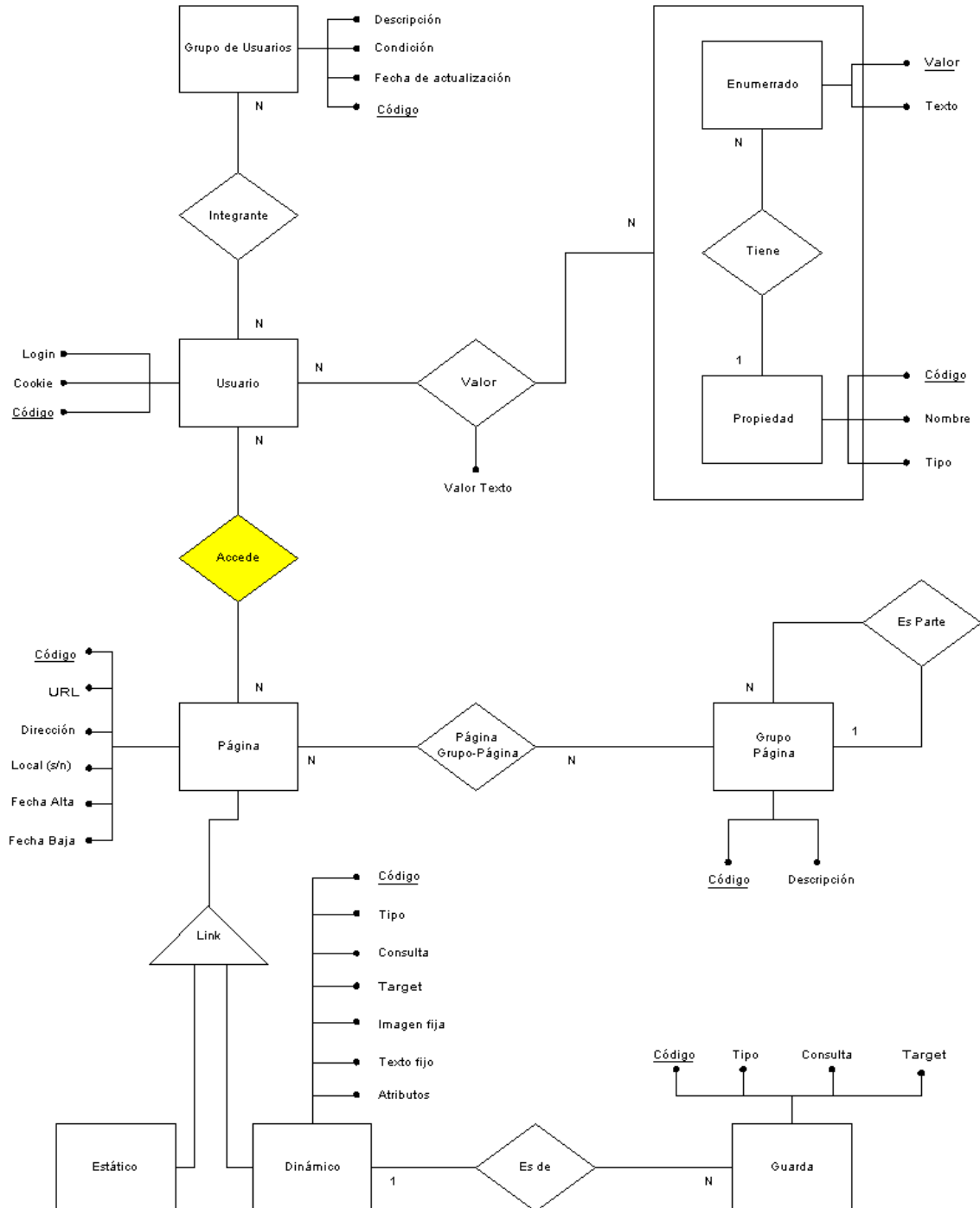
¹ Es la cookie que se entrega como identificador único a cada usuario que accede al sitio. Por mas información referirse al Apéndice B.

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

4.8. La base de datos

4.8.1 MER

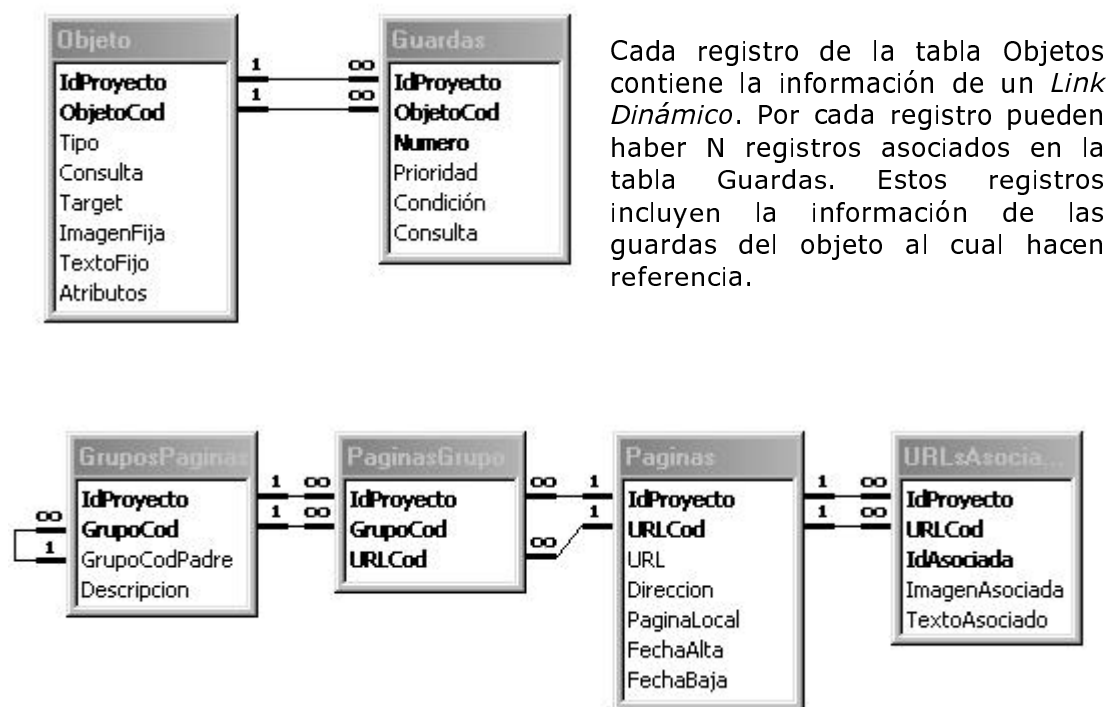


Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

4.8.2 Tablas

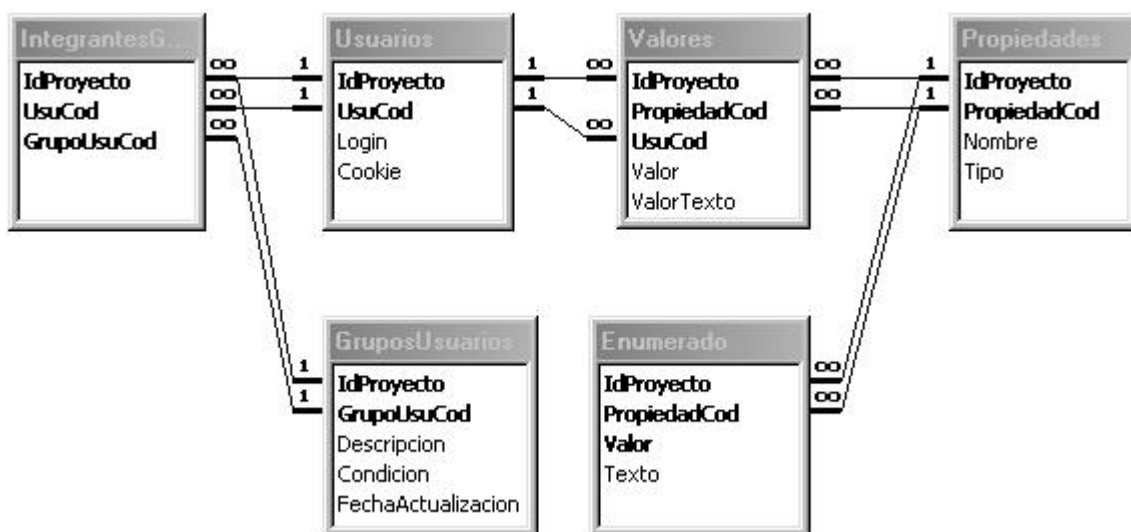
El sistema de tablas permite que se puedan manejar varios sitios desde el mismo servidor, asignando a cada tupla en todas las tablas un campo `IdProyecto` que forma parte de todas las claves primarias e identifica al sitio.



Los registros de la tabla **Páginas**, contienen la información de las páginas del sitio. Cada registro puede tener N registros asociados en la tabla **URLsAsociados**, estos contienen una imagen y un texto asociado al registro de la tabla **Páginas** al que pertenecen. También cada registro de **Páginas**, puede asociarse a N registros de **PáginasGrupo** cuyos registros indican que la página `URLCod` pertenece al grupo de páginas `GrupoCod` cuya información esta en un registro de la tabla **GruposPáginas**.

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema



Los registros de la tabla Usuario contienen la información de los usuarios, esto es el, internamente los usuarios se identifican por de UsuCod pero externamente por una Cookie. Cada usuario puede ser integrante de un grupo de usuarios, por lo tanto un registro de la tabla Usuarios puede tener N registros asociados de la tabla IntegrantesGrupo que a su vez están asociados a un registro en la tabla GruposUsuarios.

La tabla propiedades contiene los registros de las propiedades de usuario, como las propiedades pueden ser enumeradas, cada registro se puede asociar a uno de la tabla Enumerado que tiene los registros con los distintos valores que se puede asignar a una propiedad enumerada.

Los registros de la tabla Valores, asocian para cada registro de usuario un valor para cada propiedad.

Configuración
IdProyecto
Descripcion
ClassId
DirectorioOrigen
DirectorioDestino
NombreCookie
NombrePaginaConfiguracion
NombrePaginaPrincipal
Editor
DirectorioXML
ReloadTimeout

La tabla Configuración contiene todos los datos referentes al sitio.

4.9. Portabilidad

4.9.1 Utilización de múltiples sitios

Existen empresas que tienen más de un sitio. Por lo tanto el sistema permite trabajar con varios sitios desde el mismo servidor. En el administrador se selecciona el sitio y en la base de datos todas las entidades refieren a algún sitio en particular.

4.9.2 Ambientes requeridos

En la solución se encuentra que existen varios ambientes de trabajo involucrados en el sistema:

- El ambiente de trabajo del usuario final (navegante)
- El ambiente de trabajo del diseñador
- El ambiente de trabajo del administrador
- El ambiente de trabajo de la base de datos
- El ambiente de trabajo del servidor del sitio
- El ambiente de trabajo de los motores estadísticos

El ambiente de trabajo del usuario final

Como el servidor envía código HTML, que es el estándar soportado por todos los navegadores, el usuario final puede utilizar cualquier plataforma, no importa el sistema operativo que utilice, siempre y cuando tenga un navegador. Incluso puede que el usuario final utilice un dispositivo exclusivo de navegación (que no sea una computadora de propósito general)

El ambiente de trabajo del diseñador

El diseñador debe poder utilizar un editor de texto que soporte controles ActiveX, estos son objetos propios del sistema operativo Windows, por lo tanto todos los editores que soportan ActiveX son de Windows (es posible que hallan implementaciones de estos editores para otros sistemas operativos pero que no tengan la prestación de incluir ActiveX). Por lo tanto el diseñador trabajará en plataforma Windows (95, 98, ME, NT o 2000).

El ambiente de trabajo del administrador

El programa del administrado está hecho para Windows por lo tanto es claro que la plataforma de trabajo del administrador será Windows (95, 98, ME, NT o 2000).

Generación de sitios Web paramétricos adaptativos

Diseño del Sistema

El ambiente de trabajo de la base de datos

No se utiliza una base de datos en particular, el acceso a la base de datos es mediante ODBC (Open Data Base Connectivity) [14] y además todas las consultas SQL utilizadas son ANSI (1) [15] por lo tanto la elección de la base de datos corre por cuenta de la organización que utiliza el sitio. Por esto el ambiente de la base de datos será el requerido por la base de datos elegida, la única restricción es que se pueda realizar la conexión ODBC.

El ambiente de trabajo del servidor del sitio

El servidor del sitio es a elección de la organización que utiliza el sitio, la restricción es que soporte PHP, por lo tanto la restricción sobre el sistema operativo es que exista una implementación del servidor elegido y que soporte PHP con las siguientes funcionalidades especiales ODBC y UDP (User Datagram Protocol) [16] (utilizado para conectarse con el motor estadístico)

El ambiente de trabajo de los motores estadísticos

El servidor a elección está implementado para UNIX por el otro grupo de taller, única restricción de nuestra parte es que se soporte UDP.

4.9.3 Comentarios sobre las plataformas soportadas

Desde el planteo del problema se consideró como imprescindible que la plataforma de navegación sea lo más amplia posible ya que es de interés que desde cualquier navegador se pueda acceder al sitio.

La plataforma de diseño y administración si bien hubiera sido preferible que también sea multiplataforma, creemos que soportar la plataforma Windows es suficiente, en la practica no limita su utilización ya que actualmente es por mucho el sistema operativo más utilizado en el ambiente PC, además para el ambiente de diseño y administración no se requiere de un gran nivel de seguridad como si se requiere para el caso del servidor *WEB* y el servidor de base de datos.

En cuanto a la plataforma de la base de datos y el servidor *WEB*, se vio que si bien establecer una plataforma en particular no era inaceptable, era una opción poco recomendable ya que se trata de decisiones estratégicas de las organizaciones, tomadas sobre la base de muchos aspectos.

5. Informe del Proyecto

5.1. Evolución de la Solución

El primer paso

La primera etapa fue la búsqueda de productos similares. Se encontraron productos que facilitan la creación de sitios a partir de bases de datos de organizaciones, por ejemplo para el manejo de catálogos o de e-business.

Solución al comportamiento dinámico

Al enfocarnos en la construcción de la solución, tomamos la decisión de implementar un producto para utilizar editores existentes y no implementar un nuevo editor.

Con la idea de crear una extensión para editores HTML, se decidió la implementación de un método que maneje el comportamiento desde el servidor utilizando herramientas como ASP, PHP o CGI, y no mediante la utilización de Java Applets u ActiveX. Finalmente optamos por PHP luego de haber analizado la opción de utilizar ASP, CGI y otras.

En cuanto a como introducir los *Links Dinámicos* en el editor, nos inclinamos por utilizar ActiveX como *tags* y luego mediante un proceso traducir las páginas HTML que contienen estos *tags* por páginas PHP.

Por último, se decidió que el traductor no crearía código para cada *Link Dinámico*, si no que invocaría a una función genérica con un código interno de objeto. Por lo que la configuración se almacenaría en una base de datos y no en el propio *tag* del objeto.

Surgimiento de conceptos

Tomamos la decisión de manejar el comportamiento a partir de *Links Dinámicos* que representarían un tipo especial de hipervínculos. Ya que estaba claro cómo se insertarían e invocarían los links dinámicos desde las páginas *Web*, sólo faltaba averiguar como especificar su comportamiento. Una vez más aparecieron dos opciones: invocar una función distinta para cada funcionalidad o idear un lenguaje para especificar el comportamiento. De esta forma surgió el lenguaje DQL que ofrece flexibilidad y posibilidades de expansión. [Apéndice E]

Para potenciar al lenguaje, surgieron los conceptos de propiedades de usuario, (Ej.: Edad, Sexo, Profesión...) y las jerarquías de páginas.

Generación de sitios Web paramétricos adaptativos

Informe del Proyecto

Guardas y Prioridades

En reuniones con el tutor se plantearon situaciones en las que se necesitaba comportamientos diferentes de los links dinámicos dependiendo de un cierto perfil del usuario. Por otro lado, el grupo de taller "*Análisis de patrones de búsqueda con Web Mining*" no considera los parámetros del usuario en las consultas, motivo por el cual para mantener la funcionalidad, nuestro sistema paso a considerarlo.

Como resultado hubo que idear una manera de disociar el comportamiento de los links dependiendo de los valores que tomaran ciertas condiciones basadas en los parámetros de los usuarios. De esta manera aparecieron las guardas (por la similitud a las de ADA[17]).

El Administrador

En un principio DynamiX se pensó como el producto principal del desarrollo, sin embargo, al avanzar al análisis y surgir nuevas funcionalidades, fue cada vez más evidente que el rol de mantenimiento de datos no era correctamente contemplado.

Esto condujo al desarrollo del Administrador, que originalmente surgió como una aplicación de mantenimiento, pero que evolucionó hasta finalmente estar presente en cada momento del diseño del sitio *Web*. Convirtiéndose así en la pieza central de la aplicación. De esta forma, el Administrador reúne funcionalidades tanto de mantenimiento, como de configuración de comportamiento.

6. Conclusiones finales

6.1. Contribución del proyecto

Cada vez existe mayor interés en desarrollar aplicaciones para Internet. En particular, la generación de sitios que se adapten al usuario y que tomen en cuenta sus intereses, es un área aún en investigación. Asimismo, la aplicación de técnicas para el análisis estadístico de los hábitos de navegación de los usuarios en Internet puede ser de gran utilidad en el ámbito comercial. Pero para ello debe pasar primero por un estudio académico de sus reales posibilidades.

No podemos dejar de lado la contribución del proyecto en la formación de los estudiantes que integramos el equipo de desarrollo. Estos han obtenido amplios conocimientos acerca del funcionamiento y posibilidades de desarrollo en Internet.

Para la construcción de la herramienta se vincularon áreas diferentes como ser bases de datos, Data Mining, Web Mining, lenguajes de programación, comunicación entre procesos y tecnología Internet. Los estudiantes adquirimos importantes conocimientos en HTML, PHP y tecnologías de Internet en general. Además de se obtuvo una visión general de las posibilidades existentes en cuanto al desarrollo de sitios de Internet. Esto nos permitió llevar a la práctica conocimientos adquiridos en varias asignaturas de la carrera. El proyecto nos brindó la posibilidad de aplicar el ciclo completo de un proyecto de software, y en particular, en un campo más general que el de tradicionales sistemas de gestión comercial, en lo cual los integrantes del proyecto poseíamos cierta experiencia.

6.2. Trabajos futuros

Es importante considerar que el proyecto tiene una frontera definida a partir de los recursos y tiempo disponibles para realizarlo. Sin embargo se intentó que los sea lo más ampliable posible de modo de poder extender el proyecto fácilmente. A continuación presentamos ideas que es posible aplicar para extender las posibilidades del proyecto en un futuro.

6.2.1 Extender las posibilidades de elementos dinámicos

El sistema permite la inserción de *Links Dinámicos*, y que estos tengan imagen y texto asociados. Se puede configurar una serie de imágenes o textos que serán mostrados con cada página sorteada al evaluar un *Links Dinámicos*. Sin embargo no es posible definir imágenes o textos dinámicos pero que no sean hipervínculos. El sistema de guardas y traducción utilizado, sin embargo permite incluir esta funcionalidad con poco esfuerzo adicional.

Generación de sitios Web paramétricos adaptativos

Conclusiones finales

Un ejemplo de esto es crear “pieles” (skins) para las páginas *Web* mediante el uso de la tecnología CSS (Cascade Style Sheets) [10]. Mediante esta tecnología se podría lograr que se tengan predefinidos dichos estilos (por ej.: diferentes colores de fondo, diferentes tipos de letra, etc.) , y en base al comportamiento del usuario se seleccione el más adecuado y la página parezca estar totalmente personalizada para este usuario.

6.2.2 Extender las posibilidades del Administrador

Reportes de comportamiento

Luego de procesar el *log*, es posible utilizar la información de los accesos de los usuarios, y los datos que se tiene sobre los parámetros de los usuarios con el fin de combinarlos. Sería interesante que el sistema brinde reportes utilizando esa información que puede ser de especial utilidad para el departamento de Marketing de una corporación además de para los responsables de un sitio. Más aún esa información debería ser exportable para poder ser analizada con alguna herramienta de Data Mining u OLAP.

Asistente para el armado de guardas

Si bien el sistema de guardas es muy potente en cuanto a las posibilidades que permite, no es muy sencillo de utilizar sin un entrenamiento mínimo previo. Especialmente se hace difícil la asignación de las prioridades en *Links Dinámicos* de muchas guardas. Por lo tanto se podría agregar un asistente (o agente) que ayude al usuario a construir un sistema de guardas.

6.2.3 Trabajos futuros sobre los motores estadísticos

La arquitectura de la solución es lo suficientemente flexible como para ampliar las funcionalidades de los motores estadísticos gradualmente a medida que evoluciona el proyecto. Esto es debido a que el sistema de consultas dinámicas permite que al extender el lenguaje se mantengan funcionales todas las definiciones previas siempre que no se altere la sintaxis de las funcionalidades anteriores.

Puede ser interesante permitir que los motores estadísticos tomen información de los usuarios obtenida de los parámetros (como el sexo) además de las del comportamiento del usuario en el sitio *Web*.

6.2.4 El Link Dinámico

En la etapa de análisis se eligió un *tag* específico para implementar el *Link Dinámico*. Sin embargo es posible considerar una tarea mucho más ambiciosa, que consiste en desarrollar un Editor Dinámico completo. De esta forma sería totalmente transparente para el diseñador la creación de páginas *Web* dinámicas, integrando todas las aplicaciones en una única herramienta.

6.2.5 El lenguaje de traducción

La utilización de PHP también fue una elección de análisis [Apéndice B]. Al igual que para PHP, también se podrían desarrollar traductores para otros lenguajes como por ejemplo ASP. Para ésta tarea sólo sería necesario reimplementar el módulo Generador de comportamiento PHP para otros lenguajes, y como resultado se dispondría de varios módulos "Generador de comportamiento YYY". De esta forma es posible ampliar la cantidad de lenguajes soportados.

6.2.6 El lenguaje de consultas DQL

Otro aspecto importante es que la interfaz entre el Motor de Recomendación y ejecución del Generador de comportamiento PHP mediante el sistema de consultas. Se propone como trabajo a futuro ampliar el lenguaje de consultas (DQL) a medida que se mejora y se avanza en el motor estadístico y de Data Mining. Mientras se conserve la compatibilidad con las versiones anteriores de DQL, no será necesario corregir las implantaciones anteriores. Simplemente cambiando el Motor de Recomendación se pueden agregar nuevas capacidades. Por lo tanto es posible mejorar los motores estadísticos y de *Web* Mining sin que ello implique que se deban realizar grandes modificaciones en el Administrador, ni en la implementación del *Link Dinámico*.

Apéndice A

Análisis de tecnologías de páginas dinámicas

Análisis de opciones

A continuación analizamos las tecnologías de páginas dinámicas que hoy día dominan el mercado y explicamos porqué se eligió un lenguaje de servidor como herramienta para la implementación de sitios dinámicos.

Al elegir la herramienta para la implementación de los hipervínculos dinámicos se tomaron en cuenta el objetivo que se quiere lograr con la implementación de sitios dinámicos, las ventajas que ofrecían las distintas opciones y la curva de aprendizaje de cada tecnología.

El objetivo es lograr un comportamiento dinámico de las páginas Web creadas, de forma que se facilite la navegación de los usuarios, y que el tiempo de acceso a la información sea razonable.

Al analizar el problema surgen dos soluciones importantes, las cuales detallaremos a continuación.

1. Páginas Web dinámicas generadas en el momento utilizando tecnología CGI (Common Gateway Interface). En esta tecnología la información que el usuario recibe no está presente en el servidor sino que es generada por una aplicación, que funciona en conjunto con el servidor Web. Las aplicaciones CGI son programas que son ejecutados por el servidor Web al recibir una petición de página y generan como salida el contenido HTML que se envía a los navegadores.

Una aplicación es muy flexible y admite varias maneras de resolver el problema de generar páginas dinámicas. Aquí detallamos algunas opciones.

- Páginas HTML con modificaciones instantáneas.

Se trata de páginas HTML almacenadas en el servidor. La aplicación CGI se encarga de leer éstas páginas y genera otro documento HTML como salida. Este conserva la misma estructura, contenido y aspecto pero se le agregan hipervínculos para facilitar al usuario el acceso a la información de acuerdo a sus patrones de navegación. Este enfoque es poco flexible debido a que los *Links Dinámicos* se incluyen en el momento de la navegación. Esto impide que se pueda conocer de antemano el lugar donde se incluirá un *Link Dinámico*, lo que potencialmente puede alterar el diseño de la página. Una alternativa a este enfoque, es diseñar la página de forma que tenga un marco o una tabla especial donde se colocan los *Links Dinámicos*.

- Páginas HTML con marcas especiales.

Generación de sitios Web paramétricos adaptativos

Apéndice A – Análisis de tecnologías de páginas dinámicas

Esta opción es básicamente igual a la anterior con la diferencia que se utilizan marcas especiales que indican donde se deben incluir los *Links Dinámicos*. Esta alternativa resuelve el problema de diseño de la página, pero de igual forma carece de la flexibilidad suficiente para alcanzar una edición y un diseño satisfactorio. La principal desventaja es que los editores HTML convencionales no serán capaces de interpretar un nuevo símbolo que no cumpla con los estándares HTML. Desde el punto de vista de la eficiencia, el rendimiento se verá afectado debido a que es necesario analizar toda la página para buscar y sustituir cada marca, lo que implica una demora en la respuesta al usuario.

- Aplicación CGI genérica.

Esta última opción, requiere de una aplicación desarrollada especialmente que se encargue de generar todo el contenido del sitio. Esta es la opción de mayor flexibilidad, puesto que al ser generadas en el momento, las páginas pueden cambiar totalmente su contenido en cada visita.

2. Páginas ASP, PHP u otro lenguaje embebido de ejecución en servidor *Web*.

Esta alternativa permitiría que la página conservara su aspecto y en los lugares donde se desea que aparezca un *Link Dinámico* se embebe código fuente del lenguaje seleccionado que será interpretado por una aplicación en el servidor, la cual genera el documento HTML que luego se envía al usuario.

Este tipo de lenguajes cuenta con herramientas específicas como acceso a base de datos y manejo de protocolos para Internet.

La tecnología seleccionada es la descrita en el punto (2). A continuación exponemos algunas razones de porque ésta tecnología se aplica mejor al problema planteado.

En cuanto a la eficiencia, los sitios *Web* generados con la tecnología de lenguajes embebidos, que se ejecutan en el servidor, son por lo general menos eficientes que los que son generados por aplicaciones CGI. Esto se debe a que mientras las aplicaciones CGI ejecutan programas generalmente optimizados, los lenguajes que se embeben en las páginas HTML son interpretados. Como contrapartida, estos últimos requieren un conocimiento menos profundo de programación.

CGI fue de las primeras tecnologías de páginas dinámicas, siendo utilizada ampliamente en un principio, pero poco a poco ha perdido su presencia en la *Web*. Su principal desventaja es que la construcción de un sitio implica un gran esfuerzo de diseño y programación, mientras que con las tecnologías de lenguajes embebidos, se integran pequeños Scripts dentro de páginas HTML estándar. Esto permite a un diseñador gráfico editar la página con un editor comercial y luego dejar el trabajo de programación a un técnico especializado.

Generación de sitios Web paramétricos adaptativos

Apéndice A – Análisis de tecnologías de páginas dinámicas

De esta manera se puede cambiar el diseño del sitio sin necesidad de reprogramar, lo que generalmente no es posible utilizando CGI.

Debido a la flexibilidad de edición y la simplicidad de los lenguajes de Script¹ se tomo la decisión de adoptar este tipo de tecnología.

En el próximo punto se comparan los diferentes lenguajes disponibles y cual fue el usado en la implementación de la aplicación.

Selección del lenguaje

Relevamos dos de los principales lenguajes que dominan el mercado de lenguajes de Script en páginas HTML: PHP y ASP. [28] [29] [30] [31] [32]

ASP

ASP (Active Server Pages) junto con PHP pertenece a la categoría de Lenguajes de Script, lo cual indica que permite insertar pequeñas porciones de código embebidas en una página HTML. Fue desarrollado por Microsoft, y es soportado por el servidor *Web* IIS (Internet Information Server) también desarrollado por Microsoft. Existe también una versión para UNIX sobre el servidor *Web* Apache, pero es relativamente poco usada.²

ASP es un marco de trabajo que soporta varios lenguajes de scripts, en ASP el más utilizado es VBScript, pero también existe la posibilidad de utilizar Perl y JavaScript.

Al ser desarrollado por Microsoft, ASP no es gratuito, y su implementación no está disponible. Esto ayuda a que sea consistente y fácil de usar, pero impide su evolución constante y el desarrollo de componentes adicionales específicos y gratuitos, característica de las herramientas de implementación abierta.

PHP

PHP surge como un sucesor de Perl, aunque luego evolucionó independientemente. Está disponible sólo como Lenguaje de Script de Servidor, a diferencia de otros Lenguajes de Script como VBScript y JavaScript, que se encuentran disponibles en servidor y en los navegadores *Web* que provean soporte para éstos.

A diferencia de ASP, PHP es una tecnología de implementación abierta, lo cual implica que su implementación está disponible. Esto permite acceso a tecnología sin costo, permitiendo llegar a millones de usuarios fácilmente. También facilita la colaboración de programadores de todo el mundo, que ha permitido desarrollar un gran número de funcionalidades y ha hecho que esté disponible en varias plataformas. Una desventaja de este tipo de implementaciones es que no proveen soporte al producto, puesto que no hay una empresa que lo respalde.

¹ Server Side Scripting Language

² ChiliSoft ASP – <http://www.chilisoft.com>

Generación de sitios Web paramétricos adaptativos

Apéndice A – Análisis de tecnologías de páginas dinámicas

A continuación exponemos las razones por las cuales se seleccionó PHP.

Basamos la selección en las siguientes características.

- Facilidad de programación.
- Eficiencia.
- Funcionalidades disponibles.
- Acceso a datos.
- Costo económico.
- Migración de plataformas

Facilidad de programación

PHP es un lenguaje híbrido entre Perl, C y C++. Fue concebido para la programación *Web*, considerando aspectos de rendimiento en acceso a datos y disponibilidad de herramientas. Para aquellos que ya conocen C o Perl, resultan familiares las definiciones y los estilos de programación.

ASP soporta múltiples lenguajes de script, principalmente VbScript y JavaScript. Ambos resultan muy fáciles para los que conocen Visual Basic y Java. En este aspecto ASP ofrece mayor flexibilidad, además cuenta con un ambiente de programación desarrollado por Microsoft, Visual Interdev.

Eficiencia

PHP ha demostrado tener mejor rendimiento en la práctica. ASP soporta varios lenguajes de Script, por lo que internamente tiene un “wrapper” de lenguaje, esto unido al hecho de que el manejo de memoria y el acceso a los componentes externos no es óptimo lo hacen menos eficiente comparado con PHP.

Funcionalidades disponibles

PHP cuenta con muchas funciones extra que simplifican la programación. En ASP las funcionalidades extra deben agregadas como componentes adicionales, que en la mayoría de los casos son desarrollados por otras empresas y deben ser adquiridos por separado. PHP es más conveniente si se tiene la necesidad de utilizar funcionalidades específicas.

Acceso a datos

Uno de los objetivos de PHP es la simplicidad de acceso a datos. La integración entre PHP y MySQL¹ es muy buena, pero además se provee soporte para varios servidores de bases de datos (por ej.: Oracle, SQL Server, etc.) y estándares de acceso a datos (por ej: ODBC).

¹ El servidor de bases de datos de MySQL esta muy optimizado para responder consultas eficientemente, sin embargo no es comparable a servidores de gran porte como Oracle o MSSQL en termino de inserciones y transacciones. (Esta aclaración fue encontrada en la referencia 2)

Generación de sitios Web paramétricos adaptativos

Apéndice A – Análisis de tecnologías de páginas dinámicas

ASP provee acceso a datos a través de componentes como por ejemplo ADO (ActiveX Data Objects) desarrollados por Microsoft.

En este caso PHP y ASP ofrecen funcionalidades similares.

Costo

Al ser una implementación abierta PHP, es gratuito. ASP en cambio, esta incluido en IIS, el entorno de desarrollo Visual Studio, ambos de Microsoft. Otra desventaja de ASP es que si se utilizan componentes adicionales, estos deben ser adquiridos por separado.

Migración de plataformas

ASP está pensado para la plataforma Windows de Microsoft, pero existe una aplicación para UNIX sobre el servidor *Web Apache*, que permite ejecutar Scripts ASP bajo Linux. Sin embargo no es posible importar los componentes adicionales, debido a que éstos están desarrollados para Windows. PHP soporta múltiples plataformas. Esta es una ventaja notoria respecto a ASP.

Apéndice B

Descripción del módulo PHP de DynamiX.

Introducción

Este anexo contiene información detallada acerca de como se desarrollo el módulo PHP de DynamiX. Esta organizado en tres secciones:

- En la primera de ellas se describe la arquitectura del módulo.
- En la segunda se presentan notas importantes sobre el desarrollo de este módulo.
- Finalmente se discuten posibles mejoras futuras aplicables sobre este módulo.

Clases del sistema

En PHP el modelo de objetos es menos estricto que otros lenguajes orientados a objetos por lo que todas las funciones y variables de los objetos son públicas. Queda a criterio del programador decidir cuales funciones se exportan y cuales no.

Se mostraran:

- Las variables globales de control de la aplicación.
- Las clases y sus métodos.
- Funciones importantes que no están en clases.
- La aplicación de Registro.
- Para terminar la interrelación entre las clases.

Variables Globales de Control:

Muchos comportamientos por defecto del Objeto Dinámico, así como modos de debug y otras funcionalidades se establecen mediante variables globales en el sistema. Estas están en el archivo ObjetoDinamico.inc, y se pueden modificar de estas maneras:

\$CONTROL_DINAMICO_DEBUG

Dependiendo del valor de esta variable se pueden obtener distintos niveles de debug, que se muestran en la página Web resultante.

0 : No hay información de debug.

1 : Muestra las consultas DQL que se ejecutan.

2 : Muestra consultas DQL y SQL hacia el ODBC.

\$CONTROL_DINAMICO_COOKIE_LIFETIME

Es el tiempo de vida que permanecen las cookies en los navegadores de los usuarios. Por defecto se establece un período en grande, para que el sistema retenga el usuario lo más posible.

Generación de sitios Web paramétricos adaptativos

Apéndice B – Descripción del módulo PHP de DynamiX

Si se usa 0, la cookie solo dura durante la sesión activa del browser.

0 : una sesión.

157680000 : cantidad de segundos (5 años).

CONFIGURACION DEL MÓDULO DE DATAMINING

La comunicación con el motor de recomendación requiere de una IP y un puerto. Para esto se utilizan dos variables. En la especificación se definió 6000 como puerto aunque este es configurable.

`$DATAMINING_SERVER_HOST = "127.0.0.1"`

`$DATAMINING_SERVER_PORT = "6000"`

\$USANDO_MS_IIS_5

El IIS² 5 tiene un *bug* en el manejo de las cookies, que se manifiesta durante el registro del usuario. Para evitarlo hay que evitar hacer un REDIRECT en esa página, en cambio se ingresa un botón con un mensaje que indica presionarlo.

True : se esta usando IIS 5

False : se esta usando otro servidor HTTP

Clase: Cookie

Este objeto es la representación de una cookies.

Constructor: Cookie(\$CookieName, \$LifeTime)

Se invoca una sola vez por página con su constructor Cookie(name,lifetime), si no hay cookie en la sesión del lado del cliente se entrega una, sino toma el valor de la existente.

La generación de la cookie es basada en el hash MD5 de un numero aleatorio.

Variables: name, value

El valor de la Cookie se almacena en la variable value, y el nombre de la cookie en name.

Clase: DataMiningModule

Este objeto hace de interfase con el motor de recomendación, abstrayendo así el modelo del *Link Dinámico*.

Constructor: DataMiningModule(\$remoteIP, \$remotePORT)

Cuando se invoca el constructor se pasa la dirección IP y el puerto del servidor de consultas dinámicas, de aquí en más el objeto queda inicializado y sabe a donde enviar las consultas

² Microsoft Internet Information Server (IIS): es un servidor web para la línea de servidores Microsoft NT y 2000. Esta aplicación se probó con las versiones 4 y 5 de IIS y la 5 presenta este problema.

Generación de sitios Web paramétricos adaptativos

Apéndice B – Descripción del módulo PHP de DynamiX

Método: ExecuteTrace(\$User, \$ActualPageCod)

La función ExecuteTrace lanza un paquete de trace tal como es especificado en el [Apéndice C] , para poder rastrear el movimiento del usuario entre las páginas.

Método: ExecuteDQLquery(\$Consulta)

La función ExecuteDQLquery envía una consulta al Motor de Recomendación y espera una respuesta, la comunicación se hace vía UDP como se especifica en el [Apéndice C].

Clase: Date

Para la funcionalidad de fecha de alta y baja de páginas, fue necesario implementar una clase Date para encapsular estos 3 valores.

Constructor: Date(\$Anio, \$Mes, \$Dia)

La clase no hace más que almacenar los valores.

Variables: Year, Month, Day

Estas son las variables donde se almacena la fecha.

Función: DateMayor(\$A,\$B)

Función: DateIgual(\$A,\$B)

Para comparar fechas alcanza con 2 funciones que reciban 2 objetos Date y los comparen. Indican si una es mayor que la otra, o si son iguales.

Clases Auxiliares:

Estas clases se crearon para encapsular los datos de distintas entidades almacenadas en la base de datos, para abstraer las consultas ODBC y facilitar el acceso a los datos, sin necesidad de consultar la base en todo momento. Los constructores de estas clases reciben como parámetros un identificador para extraer los datos de las bases.

Class Usuario: Contiene todos los datos del usuario actual.

Class Guardas: Contiene todas las guardas del objeto actual y además sabe evaluarlas para un Usuario dado.

Class DatosDinamico: Contiene las guardas y muchos otros datos importantes para ObjDinamico.

Funciones Auxiliares:

Generación de sitios Web paramétricos adaptativos

Apéndice B – Descripción del módulo PHP de DynamiX

No todas las funciones del sistema están contenidas en objetos, algunas de ellas deben estar en el ambiente global para funcionar, otras son demasiado genéricas para ser parte de un objeto.

selectRand(\$a) : Muchas de las decisiones del Objeto Dinámico se toman basándose en selecciones aleatorias ponderadas. Justamente esta función implementa esta clase de algoritmo.

phpExtensionConverter(\$direccion) : En desuso, sirve para traducir una dirección de un archivo de extensión *.html* a uno *.php*.

BELONGS(\$grupo) ; VISITED_URL() ; VISITED_GRP(\$id_Grupo_URL): Este grupo de funciones se usan para interpretar las condiciones que se usan en las guardas, están en el ambiente global porque es el que se hereda al usar la función *eval()*.

Clase: ObjDinamico

Es el objeto principal del sistema, hace uso de todos los demás objetos e implementa todo el comportamiento del *Link Dinámico*.

Constructor: ObjDinamico(\$IdProyecto, \$dataBaseName, \$UserName, \$Password, \$URLCod)

El constructor de esta clase recibe toda la configuración necesaria para que la aplicación se conecte con una base de datos y extraiga la información necesaria para que el objeto funcione correctamente. Se ejecuta una sola vez en cada página del sitio y realiza las conexiones con la base de datos.

El parámetro IdProyecto indica cual de los proyectos que hay en la base de datos es el que se esta usando actualmente.

El parámetro URLCod es el que indica que página se esta interpretando en este momento y es el que se usa para el seguimiento del usuario.

Constructor: ObjDinamico(\$IdProyecto, \$dataBaseName, \$UserName, \$Password, \$URLCod).

```
//Conecta con la DB.  
//Extrae Los datos de configuración general para el sitio.  
//Conecta con el recomendador de Data Mining.
```

Método: TestCookie()

Este método se invoca una vez en cada página del sitio menos en la página de configuración. Se encarga de obtener la cookie del usuario o asignarle una si no tiene y en este caso mandarlo a la página de registro. Si el usuario tiene la cookie pero no aparece en la base de datos se trata como si no tuviera cookie.

Además se ejecuta el seguimiento del usuario.

Método: TestCookie().

```
//Crea un objeto cookie, así obtiene o establece la cookie.
```

Generación de sitios Web paramétricos adaptativos

Apéndice B – Descripción del módulo PHP de DynamiX

```
//Si el usuario no esta registrado: REDIRECT a página de configuración  
//Sino ejecuta el Trace del Data Mining.
```

Método: TestCookieRegister()

Es como TestCookie() solo que esta hecha para ejecutarse solo desde la aplicación de registro, y no ejecuta seguimiento.

Método: Dinamico(\$ObjId)

Recibe el identificador del *Link Dinámico* con el cual se obtiene la información para evaluarlo e insertar el hipervínculo en la página *Web*.

Método: Dinamico(\$ObjId).

```
//si no hay conexión con la base algo no funciona.  
If( no hay Conexión DB ) return 0;  
  
// extrae todos los datos correspondientes a este objeto y los almacena en el objeto  
DD.  
DD = DatosDinamico( ObjId );  
  
//evalua las guardas para este usuario, las validas se almacenana en GuardasValidas  
GuardasValidas = DD.Guardas.evaluar( UsuarioActual );  
  
If (count(GuardasValidas) == 0 ) { //si no hay gurdas validas  
    //se ejecuta la definida por defecto DD.consultaDefault.  
    PaginaSeleccionada = ExecuteDQLQuery ( DD.consultaDefault );  
}  
else {  
    //sino se selecciona al randomicamente (considerando las ponderaciones) una de  
    las validas  
    GuardaSeleccionada = selectRand( GuardasValidas );  
    // y se ejecuta  
    PaginaSeleccionada = ExecuteDQLQuery( GuardaSeleccionada );  
}  
  
//determinar si la página seleccionada esta o no habilitada para verse:  
//si fue dada de alta y si no fue dada de baja.  
//Sino se sale  
  
//insertar el link en la página dependiendo del tipo del mismo.  
Switch ( DD.tipoLink ) {  
    Case "Solo Texto" : echo "<a href = PaginaSeleccionada> DD.TextoFijo </a>"  
  
    Case "Solo Imagen" : echo "<a href = PaginaSeleccionada><img  
src=DD.ImagenFija></img></a>"  
  
    Case "Solo Texto Dinamico" : {  
        Text = selectRand(los textos disponibles para la PaginaSeleccionada)  
        echo "<a href = PaginaSeleccionada> Text </a>"  
    }  
  
    Case "Solo Imagen Dinamica" : {  
        Imagen = selectRand(las imagenes disponibles para la PaginaSeleccionada)  
        echo "<a href = PaginaSeleccionada><img src= Imagen></img></a>"  
    }  
}
```

Aplicación de registro: Register.inc

Generación de sitios Web paramétricos adaptativos

Apéndice B – Descripción del módulo PHP de DynamiX

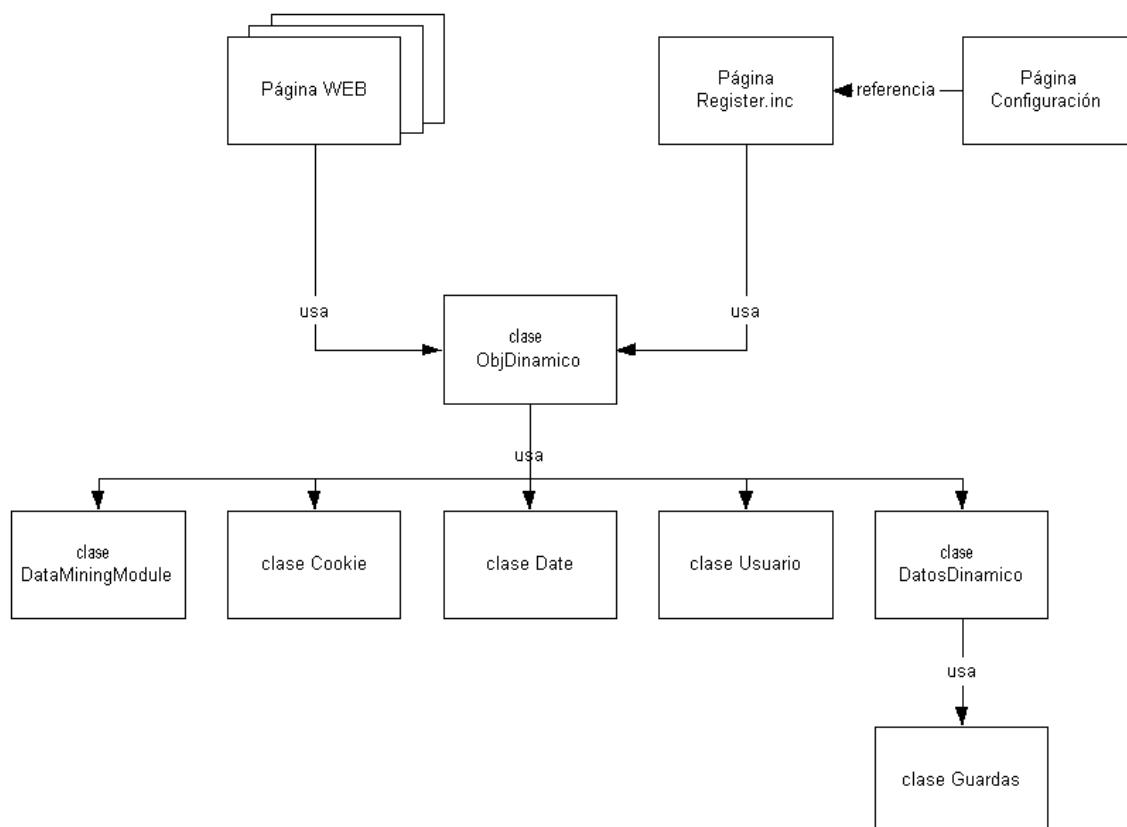
Esta aplicación se encarga de almacenar en la base de datos las propiedades de los usuarios que acaban de llenar el formulario de registro (configuración).

Cuando el usuario confirma la página, se envían los datos de propiedades ingresados mediante el método POST y se registran en la base.

Dependencias entre clases:

Además de las dependencias entre clases, se incluye otras tres entidades, que hacen parte del árbol de dependencias debido a que en ellas es donde se utilizan las clases.

- Páginas Web:
En todas las páginas del sitio se ejecutan métodos de la clase *ObjDinamico*, a veces simplemente para hacer un trace del usuario, otras para insertar un *Link Dinámico*.
- Página de configuración:
Es una página especial que no tiene ninguna componente ejecutable, solo es un formulario. Los datos que se rellenan en este formulario son luego procesados por *Register.inc*
- Página Register.inc:
Almacena los datos del usuario en la base de datos y lo habilita para acceder a las demás páginas del sitio.



Generación de sitios Web paramétricos adaptativos

Apéndice B – Descripción del módulo PHP de DynamiX

Acotaciones

Se implementó un único módulo para manejar el comportamiento de todos los *Links Dinámicos* generados por el traductor.

A continuación se describen las decisiones de implementación más relevantes.

Extensiones de archivos

Las páginas generadas deberían tener extensión php de otra forma no se invocaría al interprete de PHP para las páginas traducidas. Pero en la mayoría de los servidores (lo confirmamos para OmniHTTPD, Apache e IIS) existen formas de derivar hacia el interprete PHP las páginas html, no solo para todo el sitio *Web* sino que para directorios específicos o páginas individuales.

En vista de esto, aprovechando esta posibilidad de los servidores, evitamos cambiar la extensión de las páginas y por consiguiente de nombre (*index.html* -> *index.php*). Esto evita dificultades en el proceso de traducción ya que una página que hace referencia a otra en el propio sitio también debería ser modificada para adaptarse a la extensión.

Cookies

- Asumimos que el sitio tiene una página inicial de registro de nuevos usuarios.
- Para cada página del sitio accedida por los usuarios, el servidor verifica la existencia de una cookie de identificación y en caso de no hallarse la misma se entrega la cookie al *usuario (usuario-navegador)*. Seguidamente se verifica que el usuario identificado por la cookie este registrado en la base de datos, caso contrario se lo reenvía a la página inicial de registro. Este proceso se lleva a cabo para cada página del sitio, por lo que todas las páginas pueden redireccionar al usuario a la página inicial de registro.
- Si el diseñador no desea solicitar propiedades y el usuario no posee la cookie, se entrega la cookie al usuario y no se realiza ninguna acción posterior excepto la activación del usuario (*usuario* -> *Usuarios.login* ="OK").

Existen tres tipos de cookies:

- Permanentes
- A término (que caducan)
- De sesión.

PHP no tiene funciones para enviar cookies permanentes, pero se puede simular el efecto entregando cookies con fecha de caducidad lejana en el tiempo.

//cookie a término caduca en 100 segundos

```
setcookie("Nombre_Cookie", "Valor_Cookie", time()+100, "");
```

Generación de sitios Web paramétricos adaptativos

Apéndice B – Descripción del módulo PHP de DynamiX

```
//cookie de sesión se borra al cerrar el navegador  
setcookie("Nombre_Cookie", "Valor_Cookie , 0 , "");
```

Los servidores *Web* más sofisticados (por ejemplo Apache e IIS) pueden entregar una cookie automáticamente a cualquier usuario nuevo que ingrese a cualquier página de un sitio, como parte de una funcionalidad de seguimiento de sesiones incorporado. Si bien en se pide exactamente ese comportamiento : *"en cada página se entrega una cookie de no existir"*, es bueno tener rutinas que cumplan con estos cometidos en caso de no estar usando uno de estos servidores (por ejemplo OmniHTTPD), y en caso contrario poder reemplazar la rutina sin más complicaciones.

Para asignar el valor de la cookie hay también varias opciones, en general se puede entregar un valor que corresponda con alguna clave incremental de la base de datos o se puede generar un valor basado en la hora, y usarla como clave. El primer caso, requiere que algún programa busque en la DB valores disponibles de cookie y entregue el valor luego de registrar el usuario y finalmente volver a acceder la DB.

Por ejemplo:

```
$conexion = odbc_pconnect("DATABASE","", "");  
$resultado = odbc_exec( $conexion, "SELECT MAX(UsuarioClave) FROM  
Usuarios");  
$cookie = odbc_result($resultado, 1);  
$cookie++;
```

En el segundo caso, el propio servidor HTTP puede entregar una cookie con esta característica y que además cumple con las condiciones de unicidad de clave para la base de datos.

La función `getNewCookieValue()` retorna el valor para una nueva cookie en caso de no tener generación automática de cookies.

Por ejemplo:

Valores basados en la hora

```
$cookie = md5( uniqid(rand()) );  
// genera un identificador unico (uniqid) basado en la hora  
// y en el numero pseudoaleatorio generado por rand.  
// Con esto aseguramos unicidad de los valores.  
// Aplicando un hash MD5 aseguramos no predicibilidad.  
// MD5 : 428d4562019f83461140a821b78782b0  
// uniqid : 413a09d10bad6ae
```

Autenticación de usuarios

El estándar HTTP establece que las cookies que posee un navegador deben ser enviadas en la solicitud de la página si la cookie pertenece a esa página o sitio.

De esta manera las cookies que se encuentran en el cliente pueden y deben ser verificadas en su autenticidad por el servidor que las recibe (y con ella la

Generación de sitios Web paramétricos adaptativos

Apéndice B – Descripción del módulo PHP de DynamiX

autenticidad del usuario) o entregar una cookie nueva en caso de que el cliente no la tenga.

En este esquema el cliente en realidad no es una persona, sino que es un navegador por lo que se pueden suscitar polémicas acerca de que usuario "humano" representa una cookie en su navegador. Para resolver esta duda existe la opción de pedir al usuario un login para el sitio.

No obstante consideramos que en general es frecuente que haya un usuario "representativo" (aquel que más utiliza) del PC-Navegador. Además hay que tomar en cuenta que se pretende implementar una herramienta que simplifique la tarea de navegar por un sitio. Si el usuario debe identificarse cada vez que entra al sitio, tal vez no se pierda la simplicidad buscada.

El siguiente segmento de código invoca una función de verificación y de ser necesario redirige el navegador a una página inicial de registro. En la página inicial de registro se ingresan datos del usuario que son útiles para la selección dinámica de los hipervínculos.

Esta función debe ser invocada antes de generarse el cuerpo de las páginas enviadas como respuesta HTTP, debido a que el cabecal donde se encuentra la sentencia de redireccionamiento (redirect) se envía antes que cualquier otro dato de la página.

```
if(verificaUser($UserCookie)) { //$ UserCookie = 413a09d10bad6ae;
    //redirecciona
    header ("Location: ./config.html");
    exit;
}
```

VerificaUser es una función que revisa en la base de datos si existe la Cookie del usuario. En PHP las Cookies aparecen como variables en el programa o se encuentran en el array asociativo HTTP_COOKIE_VARS[]. [3]

Seguimiento de la Navegación (Tracking)

Cuando un usuario accede al sitio, se generan logs de acceso para su sesión que luego son procesados en forma batch por el motor de recomendación. Para poder ofrecer un servicio de recomendación en tiempo real, es preciso informar al motor de recomendación la ubicación del usuario a medida que accede a las paginas del sitio.

Entonces cada página al ser ejecutada envía al motor de recomendación³ una pareja de datos (IdUsuario, IdPagina) que se usan para determinar por que paginas esta transitando el usuario antes de hacer una recomendación.

Evaluación de condiciones de las Guardas

La condición de la guarda es una expresión lógica que se evalúa en base a los valores que toman las propiedades de usuario. Para evaluar estas

³ Esta información se envía mediante un paquete UDP, especificado en el protocolo de interconexión de tipo TRACE (ver Apéndice C)

Generación de sitios Web paramétricos adaptativos

Apéndice B – Descripción del módulo PHP de DynamiX

condiciones el camino el camino más natural es programar un intérprete para un lenguaje de condiciones, pero PHP ofrece una funcionalidad que se pudo utilizar para resolver la evaluación.

La función *eval(string)* toma como parámetro el código de un programa PHP, que es ejecutado en un ambiente separado y devuelve el valor de retorno del programa.

La solución consistió en escribir un programa con esta forma:

```
Propiedad1 = Valor1;
Propiedad2 = Valor2;
.
.
PropiedadN = ValorN;
If( Condicion ) then return 1;
Else return 0;
```

Pasando este programa al *eval()* se puede saber si el usuario cumple o no la condición.

Named Anchor

El resultado final de la ejecución de un *Link Dinámico* es un hipervínculo estándar en la página *Web*. En algunas ocasiones es útil tener una forma de referirse a estos hipervínculos, sobre todo para los Java Scripts. Para ello se incluye una etiqueta en los *Links Dinámicos*.

Mejoras futuras

Mejoras en el manejo de sesiones

Una vez que el usuario se ha identificado, se cargan valores de la base de datos con el objeto de parametrizar la página actual. Esta carga se hace siempre para cada página accedida, por lo que puede considerarse un *overhead* innecesario.

Sería bueno eliminar este overhead en versiones futuro. Para ello se puede utilizar funciones de seguimientos de sesión integradas en PHP v4, que se encargan de lo siguiente:

- Iniciar el seguimiento del usuario (mediante cookies o URL parameter) de forma transparente para el programador.
- Almacenar la información de cada sesión (esto se hace serializando los objetos en archivos temporales).
- Garbage collection que identifica sesiones muertas.

Generación de sitios Web paramétricos adaptativos

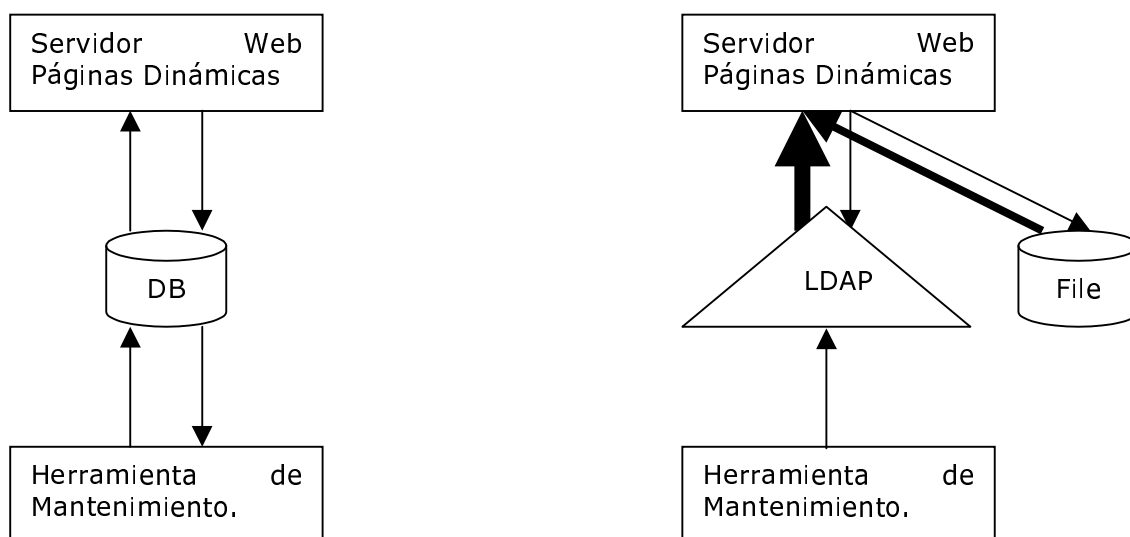
Apéndice B – Descripción del módulo PHP de DynamiX

- AutoStart de las sesiones mediante entrega de identificadores únicos.

Estas funcionalidades facilitan el desarrollo evitando el manejo de cookies y aceleran la aplicación resultante ya que reducen los accesos a la base de datos (para obtener los datos de usuario cada vez).

Aceleración de acceso a datos

Delegando a PHP (versión 4 únicamente) el manejo de sesiones se consigue minimizar el acceso a las bases de datos durante la carga. Asimismo existe otro conjunto de datos que se recuperan en cada acceso a página o al instanciar un ObjetoDinamico, para esto es inevitable acceder a alguna clase de base de datos.



Comparativo de diseños: a la izquierda la versión original con consultas hacia la base de datos para todas las operaciones. A la derecha la versión optimizada propuesta usando session_management de PHP v4 y LDAP para almacenar los datos.

Las flechas gruesas indican que la operación es más frecuente pero más rápida que la contraparte en base de datos relacional.

Para acelerar el acceso a estos datos la solución propuesta es utilizar bases LDAP (Lightweight Directory Access Protocol) que se asimila a una base de datos relacional pero almacena la información en una estructura de árbol. Para más detalles de LDAP. [ver Apéndice E]

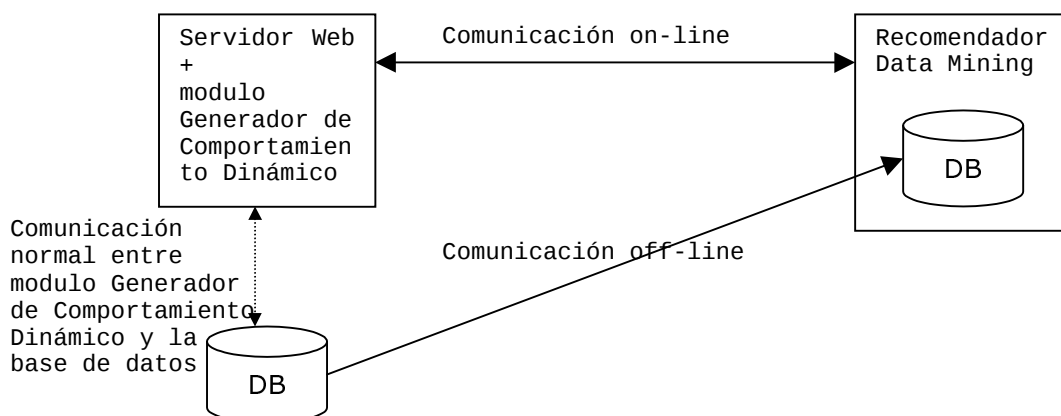
En definitiva esta solución aprovecha nuevos recursos tecnológicos para mejorar el rendimiento de la aplicación. Los cambios necesarios impactarían únicamente el módulo que genera las páginas dinámicamente y en menor medida en la herramienta de administración.

Apéndice C

Interconexión con el Motor de Recomendación.

Introducción

Tal como se plantea la solución, el motor de recomendación es una aplicación compleja que puede correr en una computadora distinta a la del servidor Web. Es importante esta característica ya que los procesos involucrados en el motor de recomendación pueden perjudicar el rendimiento del Servidor Web. Es por ello que es de fundamental importancia definir un mecanismo de interconexión remota entre estos ambos "procesos".



En nuestro modelo hay dos canales de comunicación que llamaremos on-line y off-line.

Descripción de la Componente off-line

Se encarga de sincronizar la configuración del sitio y la información de usuarios entre las aplicaciones, ya que la mayoría de las consultas que se ejecutaran vía este protocolo harán referencia extensa a estos datos, y el pasaje on-line de toda la información necesaria para la resolución de las consultas se considera un desgaste innecesario.

La idea de es replicar algunos conjuntos de datos desde la base de datos central de DynamiX hacia la base de datos del motor de recomendación.

Los datos a enviar son:

- Conjunto de Usuarios: cookie de identificación e identificador interno del sistema.

Generación de sitios Web paramétricos adaptativos

Apéndice C – Interconexión con el Motor de Recomendación

- Conjunto de Páginas: dirección URL de la página e identificador interno de la misma.
- Definiciones de Grupos de usuarios por extensión: tuplas del tipo: (identificador de grupo de usuarios, identificador de usuario).
- Definiciones de Grupos de páginas por extensión: tuplas del tipo: (identificador de grupo de páginas, identificador de página).

El mecanismo propuesto para el envío de esta información fue originalmente planteado mediante el formato XML. Luego se intercambiaría la información de formato XML mediante la utilización de un canal de comunicación estándar. (Por ejemplo SMTP, FTP, http, etc.)

En esta ocasión como las aplicaciones se analizaron conjuntamente con otro grupo de taller se tomo la decisión final de común acuerdo de compartir las estructuras de las bases de datos, y por eso mismo no se avanzo mucho en la implementación de las técnicas antes descritas. Por lo que finalmente se optó por sincronizar las bases de datos por copia o por algún método de replicación que ofrecen los DBMS.

Descripción de la Componente on-line

El servicio de resolución de consultas DQL requiere algunos datos adicionales además de la consulta en sí misma. En primer lugar requiere que durante la sesión de un usuario se reporten las páginas que visita. Para eso cada página deberá reportarse a un proceso Trace que envía esta información una sola vez por página. Por otro lado se requiere de un mecanismo de confirmación para asegurarse de que las consultas se envíen y que lleguen las respuestas.

Para resolver el primer punto se dispone de un paquete de Trace (UDP) sin acuse de recibo, y para el segundo otro tipo de paquete de Consulta (UDP) sin acuse de recibo pero con respuesta.

- Información general: Los paquetes se envían desde el módulo de generador de comportamiento PHP a la aplicación cliente mediante un paquete UDP, al puerto 6000 del motor de recomendación.
- El paquete que se envía hacia el servidor, tiene un formato específico:
<bTIPO>;<strDATA>;

Dependiendo del tipo de paquete <bTIPO> en strDATA hay:

1. **TIPO = T**, el paquete es de Trace que es utilizado para rastrear el camino recorrido por el usuario dentro del grafo de navegación. Estos paquetes tienen la forma:

T;<id_user>;<url_cod>;

Este paquete NO tiene un protocolo asociado, en si es enviarlo sin esperar

Generación de sitios Web paramétricos adaptativos

Apéndice C – Interconexión con el Motor de Recomendación

respuesta alguna y el servidor lo procesa. Sin embargo algunas implementaciones retornan un paquete de ACK conteniendo el mensaje "OK".

2. **TIPO = C**, el paquete es de Consulta y tiene la forma:

C;<id_consulta>;<consulta_DQL>;

Este paquete si tiene un protocolo para obtener una respuesta a la consulta:

- El cliente queda en espera de una respuesta en el mismo puerto por el cual envió la consulta.
- Si no se recibe respuesta en un tiempo preestablecido se asume que el paquete se perdió, en este caso hay 2 opciones:
 - i. Abandonar la Búsqueda.
 - ii. Reintentar enviando nuevamente la consulta (con el mismo id_consulta ¹).

Nota: En el cliente perfectamente puede haber más de una aplicación haciendo consultas simultáneamente, debido a que cada una se hace a través de un nuevo puerto UDP.

- El motor de recomendación por defecto queda a la espera de paquetes en el puerto 6000.
- En cuanto recibe la consulta, separa la **<consulta_DQL>**, la resuelve y devuelve la URL resultante al puerto del cliente (el cliente como envió un paquete, también envió el puerto desde el cual lo mando) con el mismo identificador de consulta:

R;<id_consulta>;<url>;

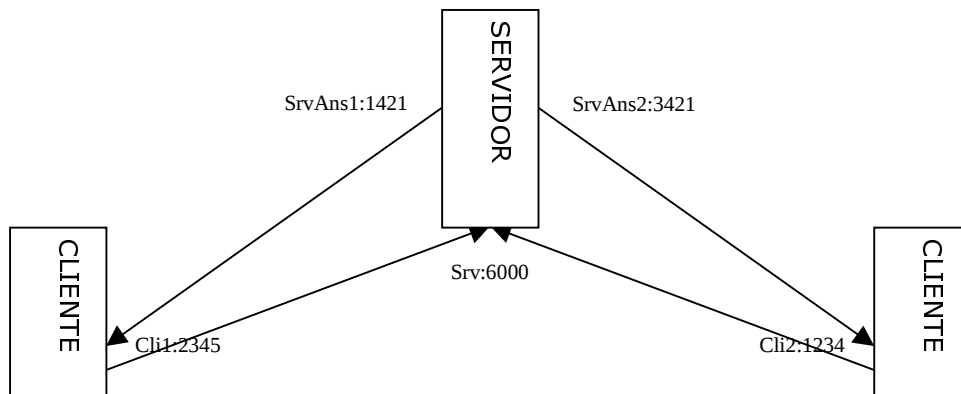
- Mientras resuelve la consulta que acaba de recibir, el servidor puede atender nuevas consultas que vayan llegando al puerto 6000 (UDP).

¹ id_consulta es un identificador generado en el cliente para identificar la consulta y la respuesta del servidor a la misma.

Generación de sitios Web paramétricos adaptativos

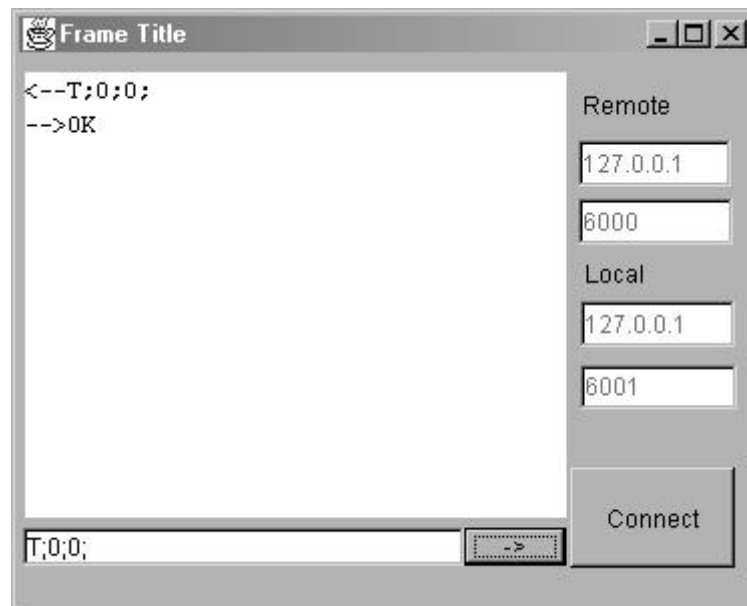
Apéndice C – Interconexión con el Motor de Recomendación

Intercambio de paquetes para el protocolo on-line



Implementaciones

- Para la implementación del generador de comportamiento dinámico se creó un aplicativo de prueba que cumple con la especificación, y se dejó a un lado el manejo de los timeouts.
- Se desarrolló una aplicación en Java y otra en Visual Basic para probar el protocolo de comunicación UDP, ya que no disponíamos de una versión del motor de recomendación operativa. Esta aplicación tenía capacidades de escuchar y contestar las comunicaciones en puertos arbitrarios.



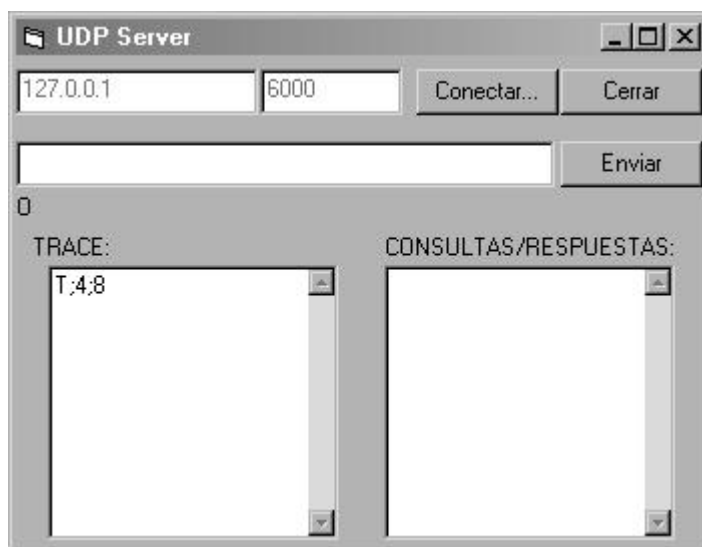
Versión Java de la aplicación de prueba.

- También se escribió un motor de recomendación primitivo que resuelve las consultas DQL más sencillas. Este servidor no es capaz de atender varias conexiones simultáneas, no tiene capacidad de atender otras

Generación de sitios Web paramétricos adaptativos

Apéndice C – Interconexión con el Motor de Recomendación

conexiones hasta que la última consulta se haya contestado. Para nuestras necesidades esta implementación resultó ser suficiente y eficiente, además de ser sencilla y cumplir con su cometido en nuestro entorno de pruebas (con 1 o 2 procesos clientes).



Servidor de prueba desarrollado en Visual Basic.

Consideraciones

- En vista de que las consultas se transfieren mediante un protocolo sin acuse de recibo, es preferible que el servidor este en la misma red física que los clientes, para minimizar la pérdida de paquetes.
- Como las solicitudes (Request) entran en un solo paquete UDP, luego de recibir una consulta, el servidor esta listo para recibir otra nueva consulta. Luego las respuestas el servidor las reenvía a los correspondientes puertos en los clientes. Así se logra soporte para el manejo de varios usuarios en forma simultanea.
- La posibilidad de emplear protocolo TCP en lugar de UDP se descartó debido a que se trata de un protocolo muy pesado para el tipo de comunicación que llevamos a cabo, y enlentecería la respuesta del motor de recomendación.

Apéndice D

Acceso a datos

DynamiX requiere una base de datos para guardar información sobre cada proyecto, sus usuarios, sus grupos de páginas, guardas, etc. Por ello es que se realizó un breve estudio sobre las grandes tendencias en acceso a datos y en particular bases de datos.

Como en otras tecnologías, a la hora de seleccionar una base de datos se deben considerar el proyecto en particular que plantea el requerimiento. Existen varios parámetros generales a considerar que dependiendo del proyecto determinarán la elección. Como breve introducción planteamos algunos parámetros comunes a considerar.

- Frecuencia de Accesos

Se refiere a la cantidad de accesos que se deben atender por minuto o segundo. En una aplicación *Web* el número potencial puede ser enorme dependiendo del sitio, por lo tanto es un parámetro fundamental.

- Tipo de Uso

El uso que se le puede dar a la base de datos puede ser solo para acceder datos (solo lectura), solo para guardar datos (solo escritura) o una mezcla de ambas. Algunos productos están específicamente optimizados para alguna de éstas modalidades.

- Recursos consumidos

A la hora de evaluar la configuración de un servidor de base de datos hay parámetros especiales a considerar. La CPU no suele ser el más importante, en general la cantidad de memoria RAM y la velocidad del disco son fundamentales. Ciertas bases de datos requieren un umbral mínimo de requerimientos muy altos, mientras que otras no tanto.

- Capacidad Transaccional

La capacidad transaccional puede ser necesaria o no en una base de datos, dependiendo de los requerimientos de la aplicación. Si se necesitan grabar transacciones complejas en múltiples tablas, esta característica realmente es muy necesaria.

- Plataformas Soportadas

Muchas bases de datos están ligadas a arquitecturas determinadas. Si los servidores de base de datos deben ser de cierta arquitectura la elección de la base de datos se verá limitada por su disponibilidad en la plataforma elegida

- Escalabilidad

Generación de sitios Web paramétricos adaptativos

Apéndice D – Acceso a datos

Hoy en día un sitio *Web* puede crecer en muy poco tiempo decenas o cientos de veces en cantidad de usuarios. Por ello una consideración importante es la posibilidad en la cual aumentando el hardware poder aumentar el rendimiento considerablemente.

- Precio

Aunque en efecto esta característica no es técnica, afecta seguramente la elección de una base de datos. El rango de precios oscila entre bases de datos gratuitas hasta versiones comerciales muy costosas. Por supuesto su calidad, confiabilidad y los demás puntos expuestos se verán muy afectados por lo general, dependiendo del precio.

MS Access

Microsoft Access permite la creación de pequeñas bases de datos para el usuario inexperto fácilmente. No es necesario tener conocimientos sobre el funcionamiento de una base de datos, MS Access guía al usuario en el proceso ofreciendo soluciones preconfiguradas para usos comunes.

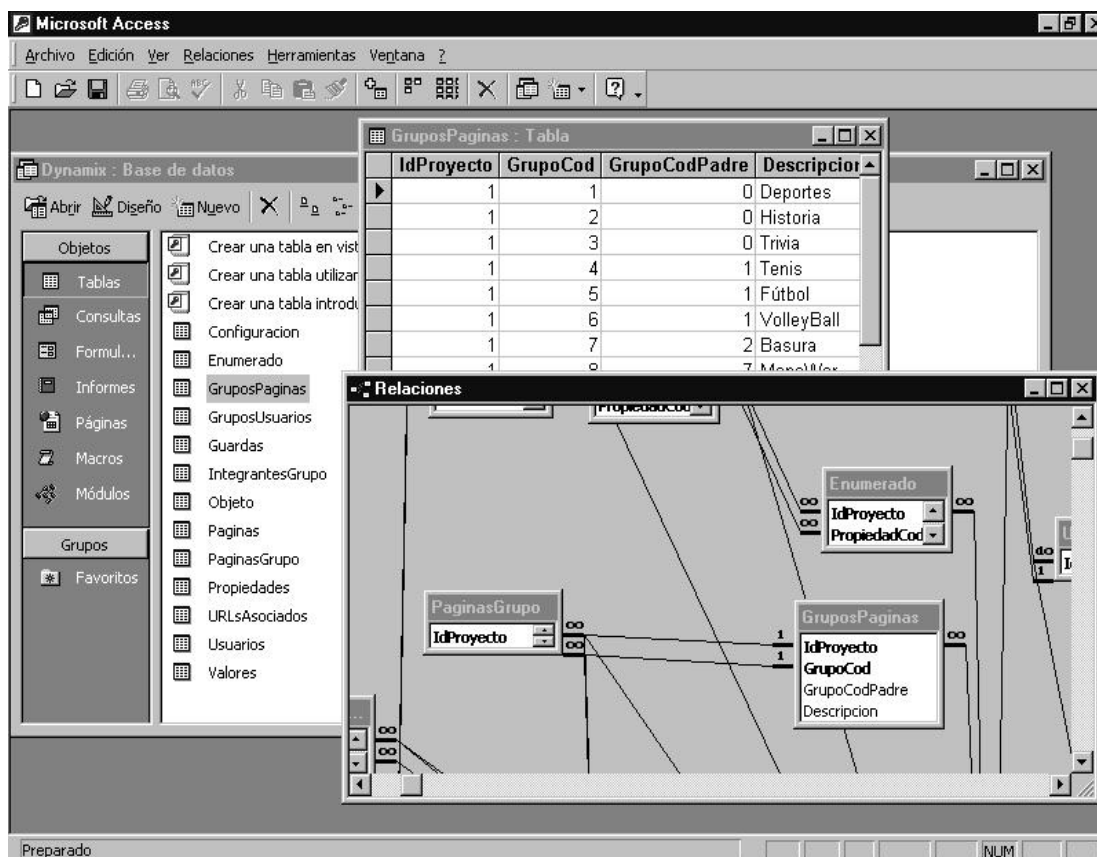
Aunque se ofrecen varias características implementadas por las grandes bases de datos, no es una opción generalmente utilizada para implementar sitios *Web* por varias razones. La principal razón es que MS Access no es una base de datos cliente servidor. Esto implica que todas las consultas deben resolverse totalmente por el cliente, lo cual reduce el rendimiento significativamente sobre todo cuando el cliente y la base de datos se encuentran en distintos equipos. Otro problema es la escalabilidad de MS Access, el cual no permite distribuir y paralelizar bases de datos en múltiples servidores, técnica utilizada para aumentar la escalabilidad.

De todos modos si el sitio utiliza una pequeña base de datos MS Access continua siendo una opción viable. Provee integridad transaccional por lo que es apta para transacciones que implican más de una tabla. No genera logs de cambios, lo que implica que tras una caída no se asegura que la base de datos continúe en un estado consistente 100%. Posee control de integridad referencial, aunque no posee control de integridad declarativa ni triggers y store procedures.

Su entorno de trabajo es sumamente amigable. Tiene una gama de editores visuales para la creación de tablas, índices, relaciones, consultas, formularios, reportes, etc., lo cual hace muy fácil e intuitivo su uso.

Generación de sitios Web paramétricos adaptativos

Apéndice D – Acceso a datos



Oracle

Oracle es un clásico en lo que se refiere a tecnología de bases de datos. Desde hace un largo tiempo ha sido pionera en esta área implementando su famosa base de datos en la mayor cantidad de plataformas desarrolladas por cualquier otra compañía.

Aunque tiene un costo importante asociado, muchas veces superior a sus competidores, Oracle no deja de ser una opción más que atractiva. Implementa de manera muy confiable y eficiente todas las características necesarias para una base de datos comercial de alto rendimiento. Ha empujado al mercado innovándolo y desarrollando de una manera más que meritoria.

Oracle provee un conjunto de herramientas gráficas para la administración de su base de datos. Además provee de herramientas de desarrollo muy simples y eficientes como ser drivers de acceso nativos, drivers ODBC, proveedores OLE DB, etc.

Oracle es una propuesta sólida y confiable para cualquier proyecto mediano o grande, y es probablemente uno de los productos más usados por los sitios Web comerciales en la actualidad.

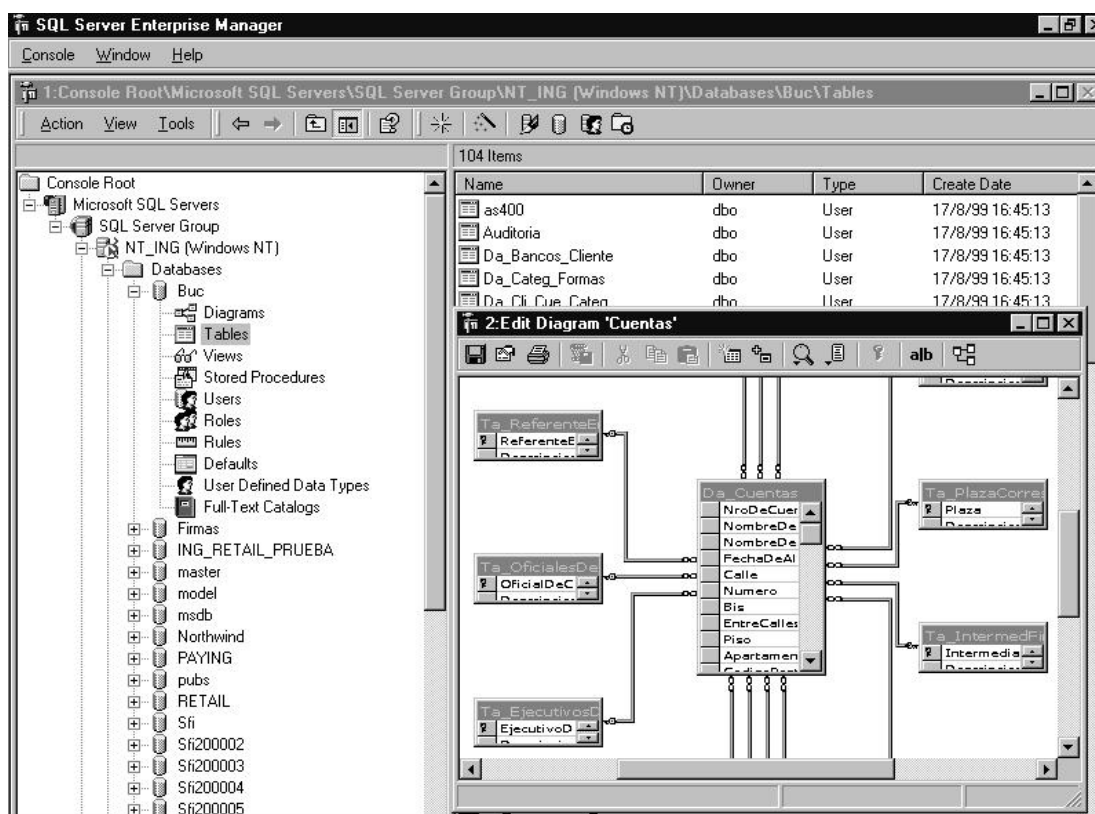
Generación de sitios Web paramétricos adaptativos

Apéndice D – Acceso a datos

MS SQL Server

MS SQL Server con su última versión 2000 se perfila como una de las principales bases de datos comerciales. Inicialmente surgió como un proyecto conjunto con Sybase, famoso fabricante de base de datos para el entorno Unix. Junto con la salida del sistema operativo OS/2 se necesitaba una base de datos para hacer realmente atractivo al nuevo producto.

Desde entonces Microsoft compró los códigos fuentes a Sybase y continuó con el desarrollo de MS SQL Server independientemente. El producto pasó de ser una base de datos de escritorio a ser realmente una seria candidata a base de datos para empresas. Ofrece todas las características expuestas en la introducción de la sección compitiendo fuertemente con el resto de las bases de datos tradicionales como Oracle, Sybase y DB2.



Junto con el motor de base de datos se proveen un grupo interesante de herramientas visuales de administración y herramientas de desarrollo comparables a las provistas por Oracle.

Recientemente Microsoft se ha unido con Compaq intentando romper marcas de alto rendimiento en procesamiento de sitios Web comerciales, logrando excelentes resultados.

Por todo esto MS SQL Server es una opción interesante a la hora de elegir por una tecnología de bases de datos.

Generación de sitios Web paramétricos adaptativos

Apéndice D – Acceso a datos

LDAP's

Los árboles LDAP tienen mejor rendimiento en aplicaciones que tienen una alta relación lecturas/escrituras, precisamente la letra "L" de LDAP corresponde a liviano (Lightweight en inglés) porque el protocolo es más liviano y rápido que uno transaccional utilizado de base de datos.

En los árboles de LDAP cada nodo corresponde a un conjunto de información (atributos). Cuando se hace una búsqueda se recorren índices especializados. Las búsquedas no pueden ser tan complejas como una consulta SQL, pero pueden organizarse por niveles del árbol. [18]

Las mejoras de rendimiento provienen del hecho que en realidad las consultas a la base de datos en nuestra aplicación son muchas y con LDAP se resolverían muy rápido, además el acceso a los datos en archivos temporarios es mucho menor que el de acceso a datos en la base sobre todo considerando la existencia de cache de disco.

Existen numerosos productos en múltiples plataformas que ofrecen LDAP como solución para bases de consulta masiva. Uno de ellos es Active Directory de Microsoft cuyo núcleo es un LDAP. Se pueden mencionar además productos de Novel NDS, Netscape Directory Service y OpenLdap.

Plataforma seleccionada: Bases Relacionales SQL compatibles + ODBC

Volviendo al caso de DynamiX, el proyecto que estamos estudiando, podemos sacar conclusiones interesantes con respecto a nuestros requerimientos.

La frecuencia de accesos a la base de datos será en una pequeña medida ocasionada por el diseñador gráfico al crear y mantener el sitio, pero la mayoría de accesos se dará por los usuarios. Por lo tanto no es posible determinar de antemano cual será la frecuencia total, es de suponer que puede ser muy alta.

El tipo de uso que se le dará a la base de datos es lectura-escritura aunque con ciertas particularidades. El diseñador del sitio utilizará la base como lectura-escritura para diseñar el sitio, pero el usuario del sitio solo utilizará la opción de lectura para decidir su navegación. Por lo tanto se espera que la base sirva como solo lectura en la mayor cantidad de tiempo.

Los recursos consumidos restringirán la elección de la base de datos a la potencia del hardware del servidor que se disponga y al tamaño de la base de datos esperada. Todo esto depende directamente del sitio en el cual se esté utilizando DynamiX.

La capacidad transaccional es un elemento necesario para DynamiX, pues se utiliza al diseñar el sitio. Por otro lado cabe destacar que la capacidad transaccional se utilizará en un porcentaje bajo, lo que implica que no se requiere de alto rendimiento en este caso.

Con respecto a las plataformas soportadas tenemos ciertas restricciones en este caso. Las herramientas de diseño de DynamiX se implementarán en plataforma Microsoft Windows, por lo tanto se debe escoger una base de datos que sea posible acceder desde un cliente Windows. Esto no implica que el servidor deba ser en plataforma Windows, alcanza con que soporte clientes Windows.

La escalabilidad de la base de datos es un requerimiento que nuevamente depende del sitio para el cual se utiliza DynamiX. Si se piensa tener un crecimiento grande en la cantidad de usuarios, esto será importante a la hora de elegir la base de datos. Por último, el precio dependerá del presupuesto del cliente que utilice DynamiX.

En conclusión, varias características necesarias de la base de datos dependen directamente del sitio en particular para el que se esté utilizando DynamiX. Además, la elección de una base de datos con costo implica un gasto adicional al cliente, sin contar que tal vez éste ya disponga de un servidor de base de datos existente en el cual se pudo haber invertido una cantidad importante de dinero y tiempo.

Tomando en cuenta todas estas consideraciones y conociendo las opciones mencionadas anteriormente, resulta en principio conveniente elegir LDAP como la solución óptima en rendimiento y en costos. Sin embargo al

Generación de sitios Web paramétricos adaptativos

Apéndice D – Acceso a datos

momento seleccionar entre las distintas opciones no teníamos acceso a esta clase de aplicaciones. Por lo expuesto en esta sección seleccionamos una tecnología de acceso a datos que nos permita acceso a una variedad importante de proveedores de bases de datos.

Nuestra única restricción fundamental es que ésta pueda ser accedida desde un cliente Windows y sea transaccional. Nuestra elección fue utilizar la tecnología de Microsoft ADO DB junto con ODBC, la cual nos da una gama de bases de datos posibles más que interesante y en combinación con ODBC Bridge para UNIX nos amplía el horizonte de plataformas soportadas.

ODBC

ODBC es un estándar para la conexión de bases de datos presentado por Microsoft que representa la sigla **O**pen **D**ata **B**ase **C**onnectivity. El lenguaje utilizado para comunicarse con las bases de datos es SQL, estándar internacional aceptado en el mercado. Su principal objetivo es permitir a una aplicación acceder a múltiples bases de datos utilizando el mismo código fuente.

La base de ODBC es la implementación de drivers, pequeñas bibliotecas desarrolladas por los fabricantes de bases de datos o terceros que implementan las primitivas de ODBC. Los drivers pueden ser instalados y cargados directamente por el usuario. Una aplicación no tiene que conocer a que base de datos accede si se ajusta al estándar, por lo tanto no requiere recodificar ni recompilar la aplicación.

La mayoría de las bases de datos ofrecen drivers ODBC para plataformas Windows. Por lo tanto esta tecnología nos permite cumplir nuestro requerimiento de soportar diversas bases de datos bajo arquitectura Microsoft Windows.

OLE DB

OLE DB se basó en la aceptación de ODBC para bases de datos relacionales. Trata de dar una respuesta universal al acceso de diversos tipos de Repositorios de Datos, en los que están incluidos las bases de datos relacionales. OLE DB va más a lejos que ODBC intentando proveer acceso a repositorios de datos como la *Web*, servidores de mail, servidores OLAP, etc.

Para el acceso de bases de datos relacionales continúa soportando ODBC, por lo que permite el acceso a cualquier base de datos que tenga drivers ODBC. También existen drivers o proveedores OLE DB diseñados específicamente para bases de datos populares como SQL Server, Oracle, etc., y se espera que en un futuro no se tenga que todos los drivers ODBC sean sustituidos por proveedores OLE DB.

Apéndice E

Especificación del lenguaje DQL.

El lenguaje de consulta (DQL) se utiliza para especificar las solicitudes de recomendación de páginas. Cada una de las sentencias del lenguaje se utiliza para realizar una solicitud diferente. A continuación se describe la sintaxis y la semántica del lenguaje.

Sintaxis del lenguaje

Tipos de datos de Base: numero_entero, numero_float y url (url es un texto con formato de URL).

usuario = grupo = posición = página := numero_entero

direccionDefault = direccionpagina:= url

tr-shot := numero_float

consulta := consulta_Select | consulta_Visited

consulta_Visited :=
VISITED URL (página) FOR THISUSER (usuario) |
VISITED URLGROUP (grupo) FOR THISUSER (usuario) |

consulta_Select :=
SELECT tipo_Consulta [FROM conjunto_Paginas] [FOR conjunto_Usuarios]

tipo_Consulta :=
RANK(posición) | RANDOM | MOST(posición) | HOT(posición, tr-shot,
direccionDefault) | HOT(posición, tr-shot, página) | URL(direccionPagina) |URL
(página)

conjunto_Paginas :=
ALL | URLGROUP (grupo)

conjunto_Usuarios :=
ALLUSERS | THISUSER(usuario) | THISGROUP(grupo)

Generación de sitios Web paramétricos adaptativos

Apéndice E – Especificación del lenguaje DQL

Matriz de Parámetros válidos

	ALL	URLGROUP	ALLUSERS	THISUSER	THISGROUP
RANK	✓	✓	✗	✓	✓
MOST	✓	✓	✓	✓	✓
RANDOM	✓	✓	✗	✓	✗
HOT	✓	✓	✓	✓	✓
URL	✗	✗	✗	✗	✗

Semántica del lenguaje

SELECT HOT (n,tr,dfl) FROM URLGROUP (gp) FOR THISGROUP (gu)

Para esta consulta necesitamos definir el concepto de velocidad de accesos. La velocidad de accesos es la cantidad de accesos a una misma página en un mismo intervalo de tiempo.

Esta consulta requiere que se encuentre una página perteneciente al grupo de páginas *gp*. La página seleccionada será la que registre el *n*-ésimo incremento en la velocidad de accesos. Donde se cuentan los accesos realizados por los usuarios del grupo de usuarios *gu*. Además el ritmo de crecimiento debe superar el valor *tr*, donde *tr* es la cota inferior que representa el diferencial de accesos a la página en un tiempo constante. Si no existe una página que cumpla tales condiciones, se retorna la página *dfl*.

SELECT HOT (n,tr,dfl) FROM ALL FOR THISGROUP (gu)

Esta consulta es análoga a la anterior, aplicada sobre el conjunto total de páginas del sitio.

SELECT HOT (n,tr,dfl) FROM URLGROUP (gp) FOR THISUSER (u)

Esta consulta es análoga a la primera, donde sólo se cuentan los accesos de usuario *u*.

SELECT HOT (n,tr,dfl) FROM ALL FOR THISUSER (u)

Esta consulta es análoga a la anterior, aplicada sobre el conjunto total de páginas del sitio.

SELECT URL (n)

La consulta requiere que se el identificador de la URL *n*.

SELECT URL (url)

La consulta requiere que se retorne la URL que se pasa por *url*.

SELECT RANK (n) FROM ALL FOR THISUSER (u)

La consulta requiere que se encuentre la página con el *n*-ésimo mejor valor de recomendación para la sesión actual del usuario *u*, tomando en cuenta los patrones de navegación descubiertos dentro del sitio.

SELECT RANK (n) FROM URLGROUP (gp) FOR THISUSER (u)

La consulta requiere que se encuentre la página, perteneciente al grupo de páginas *gp*, con el *n*-ésimo mejor valor de recomendación para la sesión actual

Generación de sitios Web paramétricos adaptativos

Apéndice E – Especificación del lenguaje DQL

del usuario u , tomando en cuenta los patrones de navegación descubiertos dentro del sitio.

SELECT RANK (n) FROM ALL FOR THISGROUP (gu)

La consulta requiere que se encuentre la página con el n -ésimo mejor valor de recomendación para la sesión actual del usuario u , tomando en cuenta los patrones de navegación presentados por usuarios pertenecientes al grupo de usuarios gu .

Esta consulta aún no ha sido implementada

SELECT RANK (n) FROM URLGROUP (gp) FOR THISGROUP (gu)

La consulta requiere que se encuentre la página, perteneciente al grupo de páginas gp , con el n -ésimo mejor valor de recomendación para la sesión actual del usuario u , tomando en cuenta los patrones de navegación presentados por usuarios pertenecientes al grupo de usuarios gu .

Esta consulta aún no ha sido implementada

SELECT RANDOM FROM ALL

La consulta requiere que se encuentre cualquier página al azar de entre todas las páginas que actualmente integran el sitio *Web*.

SELECT RANDOM FROM URLGROUP (gp)

La consulta requiere que se encuentre cualquier página al azar de entre todas las páginas que pertenecen al grupo de páginas gp .

SELECT RANDOM FROM ALL FOR THISUSER (u)

La consulta requiere que se seleccione una página al azar, entre todas las páginas que podrían recomendarse al usuario u según su sesión actual y los patrones de navegación descubiertos dentro del sitio.

SELECT RANDOM FROM URLGROUP (gp) FOR THISUSER (u)

La consulta requiere que se seleccione una página al azar, perteneciente al grupo de páginas gp , entre todas las páginas que podrían recomendarse al usuario u , según su sesión actual y los patrones de navegación descubiertos dentro del sitio.

SELECT MOST (n) FROM ALL FOR THISUSER (u)

La consulta requiere que se encuentre la n -ésima página más visitada, por el usuario u .

SELECT MOST (n) FROM ALL FOR ALLUSERS

La consulta requiere que se encuentre la n -ésima página más visitada, por todos los usuarios del sitio.

Generación de sitios Web paramétricos adaptativos

Apéndice E – Especificación del lenguaje DQL

SELECT MOST (n) FROM ALL FOR THISGROUP (gu)

La consulta requiere que se encuentre la n -ésima página más visitada, por los usuarios pertenecientes al grupo gu .

SELECT MOST (n) FROM URLGROUP (gp) FOR THISUSER (u)

La consulta requiere que se encuentre la n -ésima página, perteneciente al grupo de páginas gp , más visitada, por el usuario u .

SELECT MOST (n) FROM URLGROUP (gp) FOR ALLUSERS

La consulta requiere que se encuentre la n -ésima página, perteneciente al grupo de páginas gp , más visitada por todos los usuarios del sitio.

SELECT MOST (n) FROM URLGROUP (gp) FOR THISGROUP (u)

La consulta requiere que se encuentre la n -ésima página, perteneciente el grupo de páginas gp , más visitada, por los usuarios pertenecientes al grupo gu .

VISITED URL (p) FOR THISUSER (u)

La consulta requiere determinar si el usuario u visitó alguna vez la página p durante alguna sesión que haya mantenido en el sitio *Web*.

VISITED URLGROUP (gp) FOR THISUSER (u)

La consulta requiere determinar si el usuario u visitó alguna vez cualquier página del grupo de páginas gp durante alguna sesión que haya mantenido en el sitio *Web*.

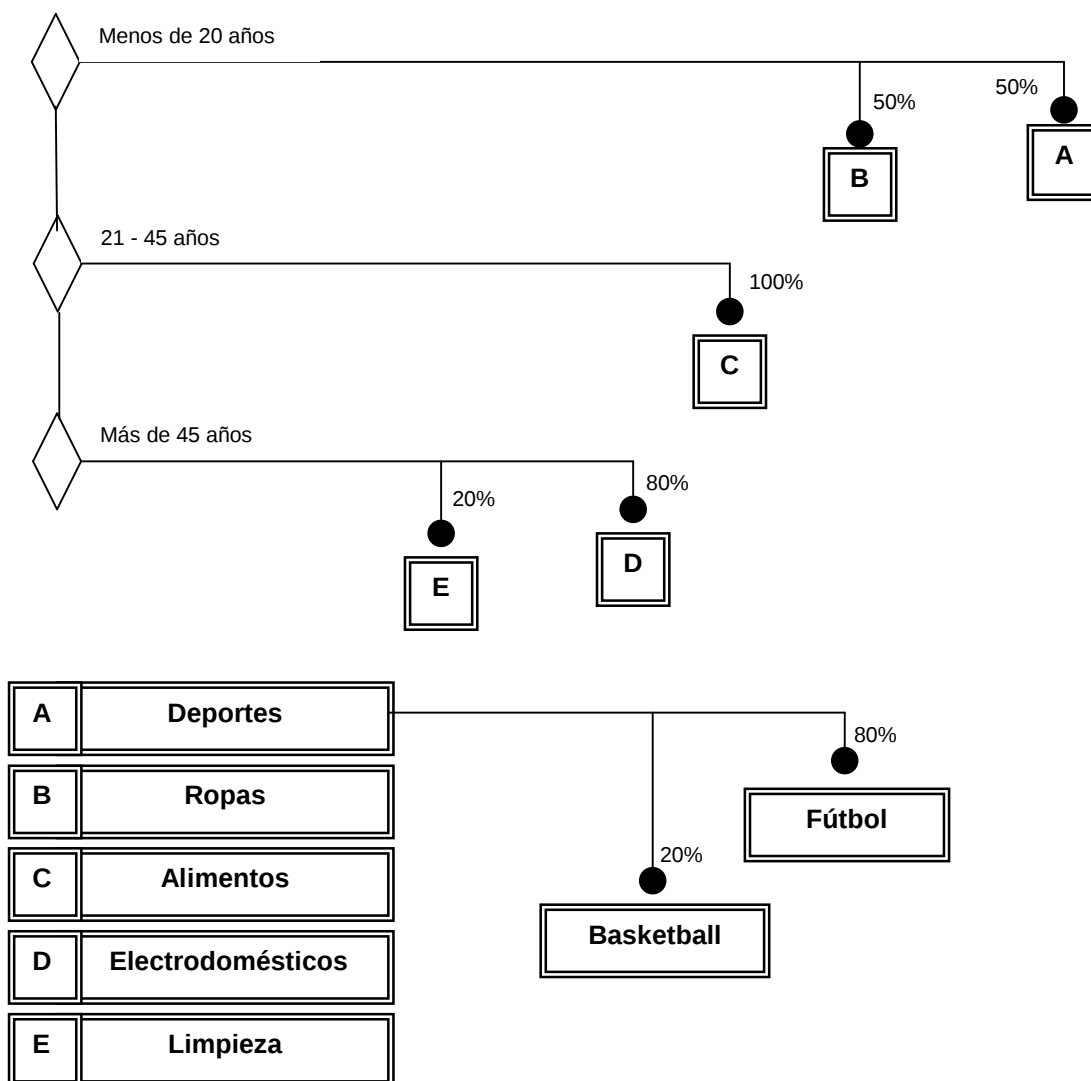
Apéndice F

Dos ejemplos de especificación de Guardas.

Ejemplo 1:

Supongamos que se desea colocar un hipervínculo de acceso rápido en la página principal de un supermercado. El supermercado desea que apenas un cliente ingrese vea una oferta de un producto que le interese al usuario.

Por lo tanto se definen grupos de usuarios y consultas de páginas para estos como se describe a continuación:



Por lo tanto se crearán las siguientes guardas:

Primera:

Prioridad: 40

Generación de sitios Web paramétricos adaptativos

Apéndice F – Dos ejemplos de especificación de Guardas

Condición: Edad < 20
Consulta: Mejor página de Fútbol

Segunda:

Prioridad: 10
Condición: Edad < 20
Consulta: Mejor página de Basketball

Tercera:

Prioridad: 50
Condición: Edad < 20
Consulta: Mejor página de Ropas

Cuarta:

Prioridad: 100
Condición: Edad >= 20 && Edad <= 45
Consulta: Mejor página de Alimentos

Quinta:

Prioridad: 20
Condición: Edad > 45
Consulta: Mejor página de Limpieza

Sexta:

Prioridad: 80
Condición: Edad > 45
Consulta: Mejor página de Electrodomésticos.

La primera, segunda y tercera guarda, refieren a las personas menores de 20 años, como se quiere que la mitad de las veces se acceda a una página de deportes y la otra mitad a una de ropa, las Prioridades de las guardas de ropa (tercera) y de deportes (primera y segunda) deben ser iguales. Las páginas de deportes que interesan son fútbol y basketball, por lo tanto se crean las guardas primera y segunda, con probabilidad de 8 a 2 a favor de la de fútbol como se quería.

Para las personas de entre 21 y 45 años, hay una sola opción que se considera con la guarda Cuarta. Las personas mayores de 45 años tienen dos posibles destinos, electrodomésticos en el 80% de los casos y Limpieza en el 20% de los casos. Esto se representó mediante las posibilidades de 80 y 20 respectivamente.

Hay que notar que por un tema de mejor comprensión se tomaron las Prioridades basadas en 100, esto no es obligatorio.

Generación de sitios Web paramétricos adaptativos

Apéndice F – Dos ejemplos de especificación de Guardas

Ejemplo 2:

Se desea ahora, mantener el criterio anterior para los hombres, para las mujeres se desea mantener el criterio anterior en el 50% de los casos y en el 50% Restante, ir a la página más vista de ropa.

Se agrega la guarda siguiente:

Séptima:

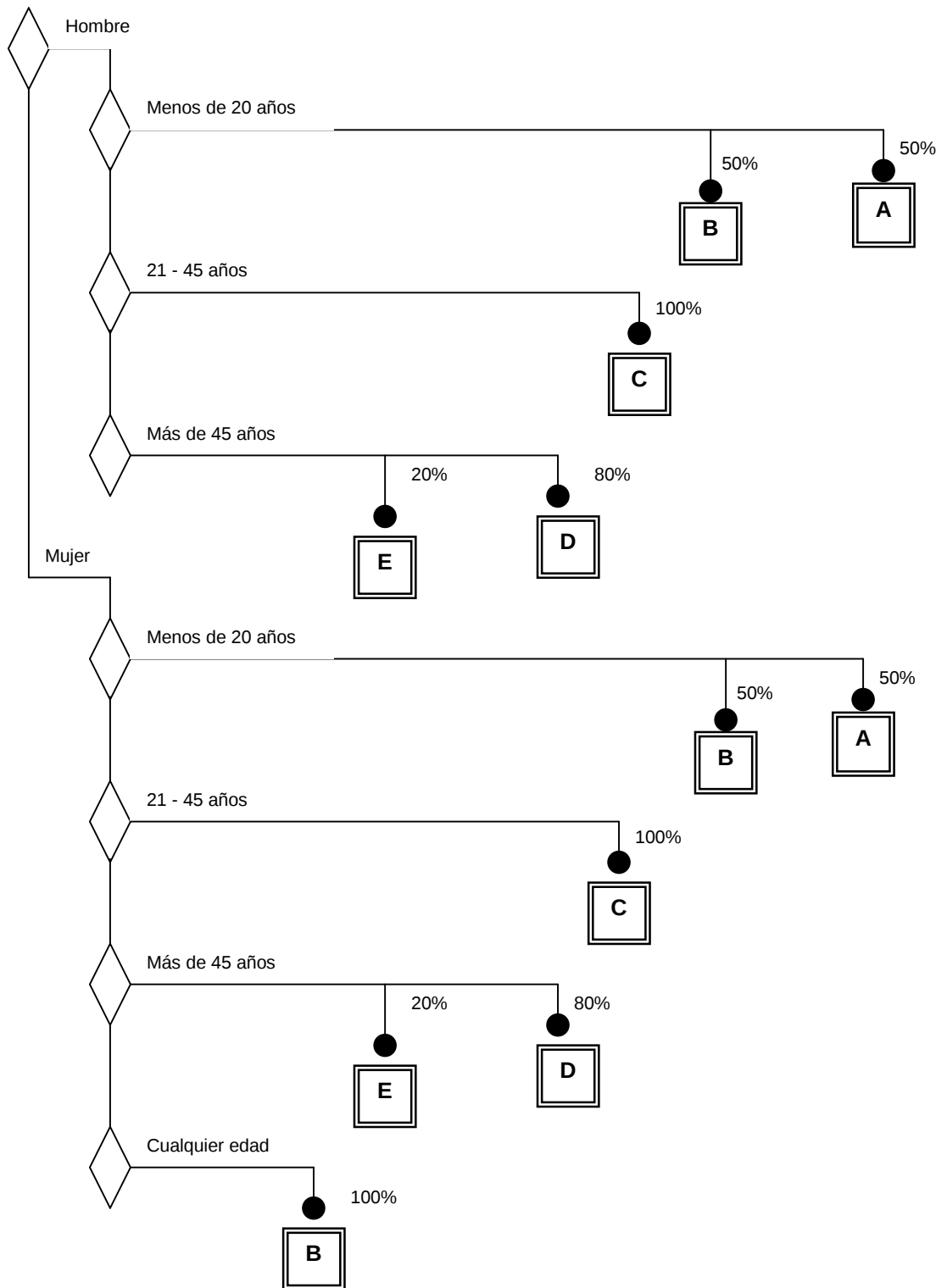
Prioridad: 100
Condición: Sexo = F
Consulta: Más visitada página de Ropa.

De esta manera el total de probabilidad de guardas válidas para hombres será de 100, correspondientes a las guardas primera a sexta habilitadas, para mujeres en cambio será de 200, 100 correspondientes a las guardas primera a sexta habilitadas y 100 correspondientes a la guarda séptima. Por ejemplo en el caso de una persona menor de 20, si es hombre, la probabilidad de ir a una página de basketball es de 10% ya que la probabilidad total de las guardas habilitadas suma 100 y la probabilidad de la guarda que lleva a la mejor página de basketball es 10. En cambio, para una mujer la suma de las Prioridades de las guardas habilitadas es 200 y la guarda de basketball es 10, por lo tanto irá a la mejor página de basketball en el 5% de los casos.

Guardas válidas			
Hombres		Mujeres	
Primera	40	Primera	40
Segunda	10	Segunda	10
Tercera	50	Tercera	50
		Séptima	100
	100		200

Generación de sitios Web paramétricos adaptativos

Apéndice F – Dos ejemplos de especificación de Guardas



Apéndice G

Configuración del servidor Web con PHP

El servidor web requiere ser configurado antes de poder funcionar con el sitio dinámico. Se utilizaron los siguientes servidores Web para probar los procedimientos de configuración de los mismos: IIS 4, IIS 5, Apache y OmniHTTPD sobre Windows. También se experimento con Apache en Linux, pero las pruebas están inconclusas.

Hay dos prerequisites del servidor web:

Por un lado comunicación directa o indirecta con la plataforma de edición, el traductor genera los archivos es un trabajo aparte copiarlos a un lugar donde estos sean visibles por el Servidor WEB. Unas opciones son: sistemas de archivos de red (Ej: NFS->WFW vía Samba o NTFS->WFW etc...), envío por FTP o en un Disquete.

Otro requisito es que el servidor esta funcionando y ya esta configurado para brindar paginas html y php.

En este proceso de configuración buscamos conectarnos con la base de datos del sitio y configurar el servidor para aceptar las extensiones de DynamiX.

Podemos separar la tarea de configuración en 3 partes:

- o Configuración del DSN.
- o Configuración del PHP.
- o Configuración de la aplicación del servidor Web.

Configuración del DSN

El DSN es el nombre del origen de datos para ODBC, configurándolo aseguramos la conexión con la base de datos del sitio. La configuración del DSN en Windows se hace de la manera descrita en el Manual de DynamiX [Manual de DynamiX; Sección de DSN].

En Linux se logra la conectividad ODBC mediante aplicaciones que hacen de "puente" con los servidores ODBC. Configurando correctamente estas herramientas se puede acceder a bases de datos ODBC, estas pruebas no se pudo realizar.

Los productos que se consiguieron fueron:

- iODBC [<http://www.iodbc.org/odbc-phpHOWTO.html>]
- Easysoft ODBC-ODBC Bridge [www.easysoft.com]
- UnixODBC [www.unixODBC.org]

Generación de sitios Web paramétricos adaptativos

Apéndice G – Configuración del servidor Web con PHP

Configuración del PHP

La configuración de PHP se limita a establecer la ruta para los archivos de inclusión de DynamiX (los archivos .inc).

En Windows se edita el archivo php.ini ubicado en C:\WINDOWS o C:\WINNT

Mientras que en Linux se ubica en /etc/.

Dentro del archivo hay que modificar la línea de include_path:

```
include_path = "c:\DynamiX\php\;"
```

Cuando el servidor web es IIS 5 hay que modificar una variable dentro de los archivos de inclusión de DynamiX, esto es para evitar un Bug del Servidor descrito en el Apéndice B.

El archivo que se debe modificar se llama ObjetoDinamico.inc y hay que agregar la línea:

```
USANDO_MS_IIS_5 = 1;
```

Antes de:

```
CONTROL_DINAMICO_DEBUG = 0;
```

Configuración del Servidor Web

El servidor web requiere una configuración especial, procesa las paginas html como si fueran php, para lograr esto hay que Especificarlo en el servidor. De esta manera todas las paginas traducidas del sitio dinámico, pueden ser procesadas por DynamiX.

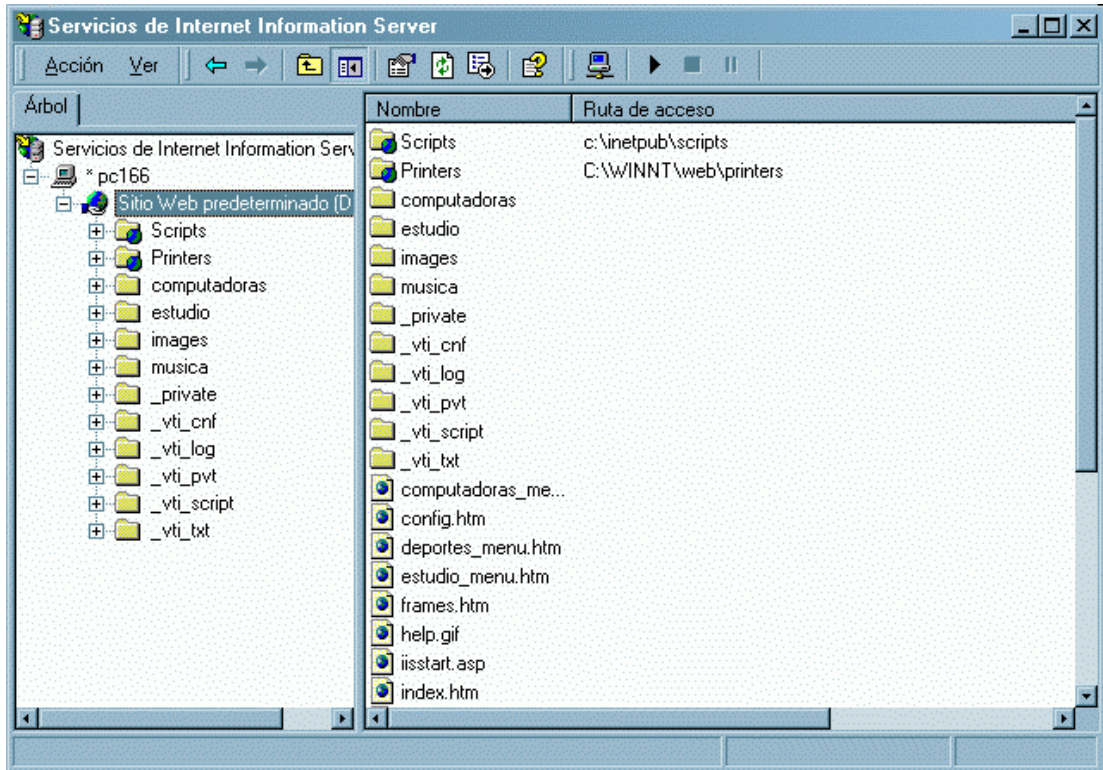
La configuración es distinta para cada servidor web, aquí reportaremos los detalles de configuración para los servidores usados en las pruebas.

En IIS:

- Se selecciona en el "Panel de Control" el "Administración de servicios de Internet".
- Con el botón derecho de ratón se hace Clic en la raíz del sitio Web, y se selecciona Propiedades.

Generación de sitios Web paramétricos adaptativos

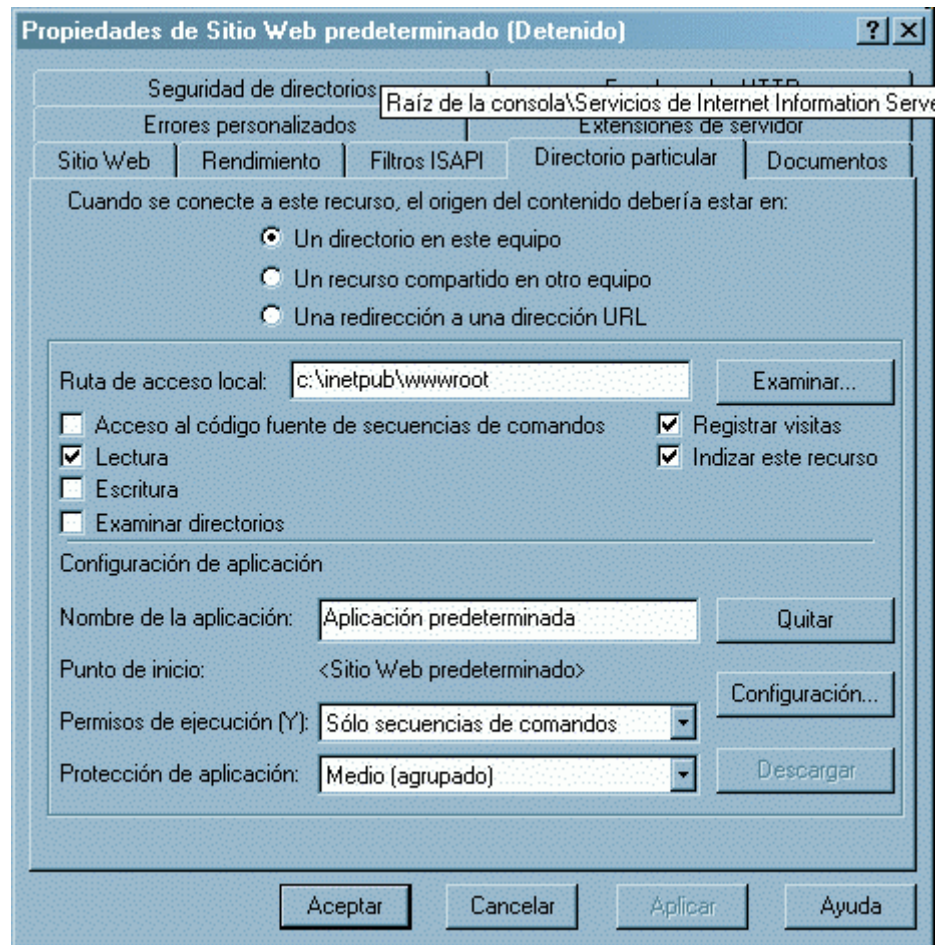
Apéndice G – Configuración del servidor Web con PHP



- Seleccionamos el tab de : "Directorio Particular" y en la sección "Configuración de Aplicación" pulsar en Configuración.

Generación de sitios Web paramétricos adaptativos

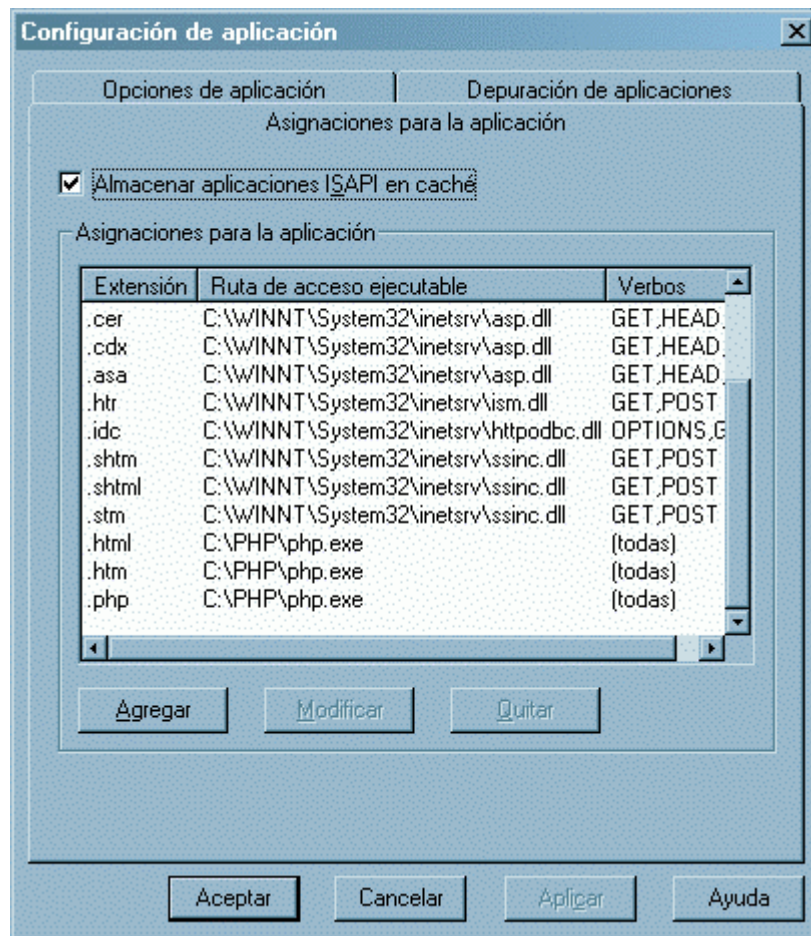
Apéndice G – Configuración del servidor Web con PHP



- Aparecerá una ventana con las asociaciones entre Extensiones de archivos y aplicaciones que los procesan.
- Agregar o Modificar las extensiones htm y html, para que estén asociadas al procesador de PHP (tal como se ve en el ejemplo).

Generación de sitios Web paramétricos adaptativos

Apéndice G – Configuración del servidor Web con PHP



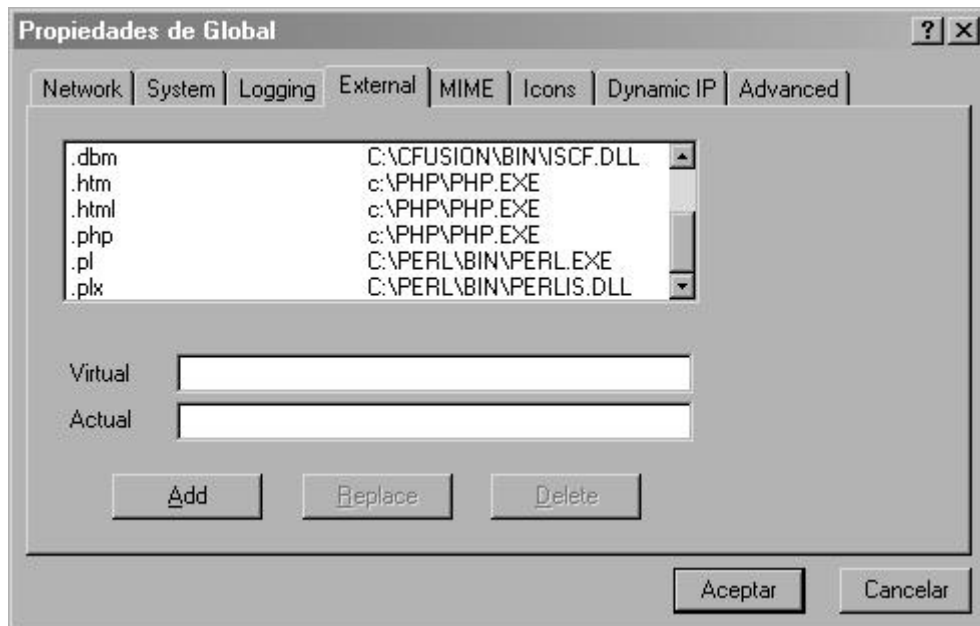
- Salir de las ventanas y aceptar en cada una las modificaciones.
- Reiniciar el servidor Web.

En OmniHTTPD:

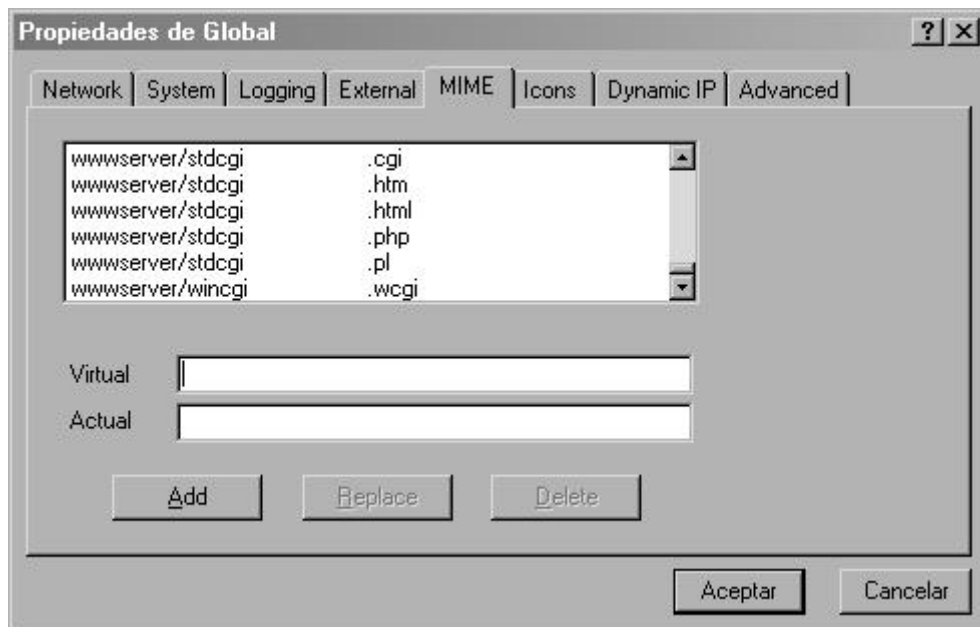
- Ir a la configuración del OmniHTTPD, y seleccionar "Web Server Global Settings"

Generación de sitios Web paramétricos adaptativos

Apéndice G – Configuración del servidor Web con PHP



- En el tab "External"
- Agregar las extensiones htm y html como están en la imagen.



- Luego ir al tab MIME y modificar las entradas de la lista correspondientes a htm y html, para que quede como en la imagen.
- Grabar todo y Reiniciar el servidor.

Generación de sitios Web paramétricos adaptativos

Apéndice G – Configuración del servidor Web con PHP

Apache:

La configuración de Apache para Windows es igual a la de Linux. Para configurar Apache hay que incluir el modulo de PHP o compilar Apache con soporte PHP [7], en ambos casos hay que modificar el archivo de configuración httpd.conf y registrar los tipos PHP.

```
<IfModule mod_php4.c>
    AddType application/x-httpd-php3 .php4
    AddType application/x-httpd-php3 .php
    AddType application/x-httpd-php3 .htm
    AddType application/x-httpd-php3 .html
    AddType application/x-httpd-php3-source .phps
</IfModule>
```

Apéndice H

Descripción de la Base de Datos

Tabla : Configuración Almacena los datos básicos de configuración del proyecto		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
Descripcion	Texto(255)	Descripción del proyecto
ClassId	Texto(255)	ClassId del ActiveX "Link Dinamico"
DirectorioOrigen	Texto(255)	Directorio del cual se toma el sitio
DirectorioDestino	Texto(255)	Directorio donde se guardan las pág. Compiladas
NombreCookie	Texto(50)	Nombre de la cookie a guardar en los usuarios
NombrePaginaConfiguracion	Texto(255)	Nombre de la página de configuración generada por el sistema
NombrePaginaPrincipal	Texto(255)	Nombre de la página principal del sitio
Editor	Texto(255)	Ubicación del Editor HTML por defecto
Explorador	Texto(255)	Ubicación del Navegador HTML por defecto

Tabla : Enumerado Guarda la configuración de las propiedades enumeradas del proyecto		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
PropiedadCod*	Entero Largo	Código de Propiedad
Valor*	Entero Largo	Valor numérico a almacenar
Texto	Texto(50)	Texto a mostrar al usuario

Generación de sitios Web paramétricos adaptativos

Apéndice H – Descripción de la Base de Datos

Tabla : GruposPaginas Guarda los grupos de páginas configurados para cada proyecto		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
GrupoCod*	Entero Largo	Código del grupo de páginas
GrupoCodPadre	Entero Largo	Código del padre del grupo si lo tiene
Descripcion	Texto(50)	Nombre descriptivo del grupo

Tabla : GruposUsuarios Guarda la configuración de grupos		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
GrupoUsuCod*	Entero Largo	Código del grupo de usuarios
Descripcion	Texto(50)	Descripción del grupo
Condicion	Texto(50)	Condición que define por comprensión el conjunto de usuarios que integran el grupo
FechaActualizacion	Fecha/Hora	Última Fecha/Hora de armado del grupo

Tabla : Guardas Configuración de guardas para los “Links Dinámicos”		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
ObjetoCod*	Entero Largo	Identificación del Link Dinámico al que pertenece la guarda
Numero*	Entero Largo	Orden de ingreso de la guarda
Descripcion	Texto(50)	Descripción de la guarda
Prioridad	Entero Largo	Prioridad asignada a la guarda
Condición	Texto(255)	Condición que se debe satisfacer para ejecutar la consulta
Consulta	Texto(255)	Consulta DQL para la obtención de una URL

Generación de sitios Web paramétricos adaptativos

Apéndice H – Descripción de la Base de Datos

Tabla : IntegrantesGrupo Almacena la integración de cada grupo de usuarios		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
UsuCod*	Entero Largo	Identificación de Usuario perteneciente al grupo
GrupoUsuCod*	Entero Largo	Identificación de Grupo al cual pertenece

Tabla : Objeto Almacena la configuración de todos los objetos “Link Dinámico”		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
ObjetoCod*	Entero Largo	Identificación del Objeto
Tipo	Texto(2)	Tipo de Objeto
Consulta	Texto(255)	Consulta DQL por defecto
Target	Texto(20)	Frame target del hipervínculo
ImagenFija	Texto(255)	Ruta de imagen en caso de ser fija
TextoFijo	Texto(255)	Texto en caso de ser fijo
Atributos	Texto(255)	Atributos HTML del link

Generación de sitios Web paramétricos adaptativos

Apéndice H – Descripción de la Base de Datos

Tabla : Paginas Almacena la configuración de páginas de cada proyecto		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
URLCod*	Entero Largo	Identificación de la página
URL	Texto(255)	Dirección local de la página
Direccion	Texto(255)	Dirección HTTP normalizada
PaginaLocal	Booleano	Indica si la página es local al sitio o una referencia externa
FechaAlta	Fecha/Hora	Fecha de Alta de la página
FechaBaja	Fecha/Hora	Fecha de Baja de la página si se dio de baja

Tabla : PaginasGrupo Almacena la integración de cada grupo de páginas		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
GrupoCod*	Entero Largo	Identificación del grupo de páginas
URLCod*	Entero Largo	Identificación de la página

Tabla : Propiedades Almacena la configuración de propiedades de usuario para cada proyecto		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
PropiedadCod*	Entero Largo	Identificación de la propiedad
Nombre	Texto(30)	Nombre de la propiedad
Tipo	Texto(1)	Tipo de la propiedad. Enumerado o Texto

Generación de sitios Web paramétricos adaptativos

Apéndice H – Descripción de la Base de Datos

Tabla : URLsAsociados		
Almacena la configuración de texto e imágenes asociadas a URLs		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
URLCod*	Entero Largo	Identificación de la URL
IdAsociada*	Entero Largo	Número de URL asociado
ImagenAsociada	Texto(255)	URL de la imagen asociada
TextoAsociado	Texto(255)	Texto asociado

Tabla : Usuarios		
Almacena la información de los usuarios que acceden cada proyecto		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
UsuCod*	Entero Largo	Identificación del Usuario
Login	Texto(50)	Estado del Login
Cookie	Texto(255)	Identificación de la cookie enviada al usuario

Tabla : Valores		
Almacena los valor ingresados para las propiedades por cada usuarios		
Nombre de Campo	Tipo de Datos	Descripción
IdProyecto*	Entero Largo	Número de identificación del proyecto
PropiedadCod*	Entero Largo	Identificación de la propiedad
UsuCod*	Entero Largo	Usuario que ingreso el valor
Valor	Entero Largo	Valor para propiedades enumeradas
ValorTexto	Texto(50)	Valor para propiedades de texto

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Apéndice I

Casos de prueba de la aplicación

ESPECIFICACIÓN DE CASO DE TESTEO

Módulo: Administrador

Nº de caso: 1

Descripción General del caso: Crear una base de datos y entrar al administrador por 1era vez					
Sub-caso 1: Crear base de datos Access					
Sub-caso 2: Crear base de datos SQL Server					
Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Crear DSN apuntando a una nueva base de datos Access					
Sub-caso 2: Crear DSN apuntando a una nueva base de datos SQL Server					
Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Ejecutar DyanamiX DbConfig indicando el nombre del DSN apuntando a Access y oprimir aceptar.					
Sub-caso 2: Ejecutar DyanamiX DbConfig indicando el nombre del DSN apuntando a SQL Server y oprimir aceptar.					
Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Revisar que se hayan agregado tablas a la base de datos Access recién creada y probar ingresar al Administrador sin problemas.					
Sub-caso 2: Revisar que se hayan agregado tablas a la base de datos SQL Server recién creada y probar ingresar al Administrador sin problemas.					
Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1					
Sub-caso 2					
Sub-caso 3					

RESULTADO Ok: SI ____ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 2

Descripción General del caso: Agregar un proyecto nuevo a una base de datos ya existente. Sub-caso 1: Sub-caso 2: Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso) Sub-caso 1: Configurar el Administrador para apuntar a un DSN que apunte a una base de datos que ya contenga proyectos Sub-caso 2: Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.) Sub-caso 1: Ir al menú Configuración , en la opción Configuración dar clic en el botón Agregar . Sub-caso 2: Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR). Sub-caso 1: Revisar que se hayan grabado los datos ingresados al proyecto. Sub-caso 2: Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2					
Sub-caso 3					

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 3

Descripción General del caso: Editar los datos de un proyecto ya existente.					
Sub-caso 1: Sub-caso 2: Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Configurar el Administrador para apuntar a un DSN que apunte a una base de datos que ya contenga proyectos					
Sub-caso 2: Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Ir al menú Configuración y entrar en la opción Configuración. Seleccionar un proyecto y dar clic en el botón Editar.					
Sub-caso 2: Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Cambiar cada uno de los datos y revisar que se hayan grabado los cambios correctamente.					
Sub-caso 2: Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2					
Sub-caso 3					

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 4

Descripción General del caso: Borrar un proyecto existente.					
Sub-caso 1: Borrar un proyecto de una base con varios proyectos					
Sub-caso 2: Borrar un proyecto de una base con un único proyecto.					
Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Configurar el Administrador para apuntar a un DSN que apunte a una base de datos que ya contenga varios proyectos					
Sub-caso 2: Configurar el Administrador para apuntar a un DSN que apunte a una base de datos que contenga un único proyecto					
Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Ir al menú Configuración y entrar en la opción Configuración. Seleccionar un proyecto y dar clic en el botón Eliminar.					
Sub-caso 2: Idem					
Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Mensaje "El proyecto se ha borrado correctamente". Revisar que efectivamente se hayan borrado los datos de la base.					
Sub-caso 2:					
Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2					
Sub-caso 3					

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 5

Descripción General del caso: Cambiar de DSN utilizado por el módulo Administrador.					
Sub-caso 1: Sub-caso 2: Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Crear dos DSN que apunten a dos bases de datos diferentes. Sub-caso 2: Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Ir al menú Configuración y entrar en la opción Configuración. Seleccionar un proyecto y dar clic en el botón "Configurar Base de Datos", seleccionar un DSN y volver a entrar al Administrador. Sub-caso 2: Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Revisar que la base de datos y los proyectos a los que apunta ahora el Administrador pertenecen a la base de datos recién seleccionada. Sub-caso 2: Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2					
Sub-caso 3					

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 6

Descripción General del caso: Importar un sitio web ya existente.					
Sub-caso 1: Importar un directorio que no tenga ninguna página					
Sub-caso 2: Importar un directorio con una o más páginas html					
Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Crear un nuevo proyecto en el Administrador.					
Sub-caso 2: Idem anterior.					
Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Ir al menú Configuración y entrar en la opción Importar Sitio. Elegir un directorio sin paginas html y oprimir el botón Importar Páginas					
Sub-caso 2: Ir al menú Configuración y entrar en la opción Importar Sitio. Elegir un directorio con varias paginas html y oprimir el botón Importar Páginas.					
Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Mensaje de confirmación. Revisar que no se haya importado ninguna página.					
Sub-caso 2: Mensaje de confirmación. Revisar que se hayan importado todas las páginas del directorio.					
Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3					

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 7

Descripción General del caso: Chequeo interfase de consulta de Usuarios del sitio.					
Sub-caso 1: Sub-caso 2: Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Configurar administrador para trabajar con un proyecto de un sitio utilizado por varios usuarios web diferentes Sub-caso 2: . Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Trabajar con la interfase, agrupando por columnas Propiedad y Valor en diferentes órdenes y formas Sub-caso 2: Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Verificar que la consulta se comporte correctamente en todo momento. Sub-caso 2: Sub-caso 3:					
CONDICIÓN	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	<i>Marcar una X</i>	<i>Marcar una X</i>	<i>Sólo si el caso dio ERROR</i>		
Sub-caso 1	X				
Sub-caso 2					
Sub-caso 3					

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 8

Descripción General del caso: Módulo de mantenimiento de propiedades de usuario.					
Sub-caso 1: Agregar, Borrar y Modificar propiedades de usuario Sub-caso 2: Agregar, Borrar y Modificar valores posibles de propiedades de usuario Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Configurar administrador para trabajar con un cualquiera Sub-caso 2: Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Sub-caso 2: Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Verificar que las propiedades se guarden correctamente. Sub-caso 2: Verificar que los valores posibles de los enumerados se guarden correctamente. Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	<i>Marcar una X</i>	<i>Marcar una X</i>	<i>Sólo si el caso dio ERROR</i>		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3					

RESULTADO Ok: SI X NO

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 9

Descripción General del caso: Módulo de mantenimiento de propiedades de usuario.					
Sub-caso 1: Agregar, Borrar y Modificar propiedades de usuario Sub-caso 2: Agregar, Borrar y Modificar valores posibles de propiedades de usuario Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Configurar administrador para trabajar con un proyecto cualquiera Sub-caso 2: Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Sub-caso 2: Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Verificar que las propiedades se guarden correctamente. Sub-caso 2: Verificar que los valores posibles de los enumerados se guarden correctamente. Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	<i>Marcar una X</i>	<i>Marcar una X</i>	<i>Sólo si el caso dio ERROR</i>		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3					

RESULTADO Ok: SI X NO

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 10

Descripción General del caso: Módulo de mantenimiento de grupos de usuario.					
Sub-caso 1: Agregar, Borrar y Modificar grupos de usuario Sub-caso 2: Trabajar con el DQL Wizard para crear condiciones. Sub-caso 3: Generar grupos de usuarios					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Sub-caso 2: Configurar administrador para trabajar con un proyecto que tenga una cantidad de usuarios importante y propiedades de usuario configuradas. Sub-caso 3: Idem 2					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Sub-caso 2: Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Verificar que los grupos de usuarios se guarden correctamente. Sub-caso 2: Verificar que se generen condiciones sintácticamente correctas en todos los casos. Sub-caso 3: Verificar en la tabla integrantes grupos que se los grupos se generaron correctamente					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3	X				

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 11

Descripción General del caso: Módulo de mantenimiento de Páginas Web.					
Sub-caso 1: Agregar, Borrar y Modificar Páginas Web Sub-caso 2: Agregar, Borrar y Modificar textos e imágenes asociadas Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Configurar administrador para trabajar con un proyecto que tenga páginas Web en lo posible, de lo contrario importar un sitio y agregar páginas manualmente Sub-caso 2: Idem 1 Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Sub-caso 2: Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Verificar que las Páginas Web se guarden correctamente. Sub-caso 2: Verificar que los textos e imágenes asociadas se guarden correctamente. Sub-caso 3:					
CONDICIÓN	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	<i>Marcar una X</i>	<i>Marcar una X</i>	<i>Sólo si el caso dio ERROR</i>		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3					

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 12

Descripción General del caso: Módulo de mantenimiento de Grupos de Páginas Web.					
Sub-caso 1: Agregar, Borrar y Modificar Grupos de Páginas Web Sub-caso 2: Agregar, Borrar y Modificar URLs asociadas Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Configurar administrador para trabajar con un proyecto que tenga páginas Web en lo posible, de lo contrario importar un sitio y agregar páginas manualmente Sub-caso 2: Idem 1 Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Sub-caso 2: Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Verificar que las Grupos de Páginas Web se guarden correctamente. Sub-caso 2: Verificar que las URLs asociadas se guarden correctamente. Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	<i>Marcar una X</i>	<i>Marcar una X</i>	<i>Sólo si el caso dio ERROR</i>		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3					

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 13

Descripción General del caso: Visualización mediante diagrama de Edición del sitio.					
Sub-caso 1: Visualización de sitios complejos con URLs locales, externas y links dinámicos					
Sub-caso 2: Llamado a editor predeterminados en URLs locales					
Sub-caso 3: Llamado a navegador predeterminado en URLs externas					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Configurar administrador para trabajar con un proyecto que tenga páginas Web en lo posible, de lo contrario importar un sitio o agregar páginas manualmente					
Sub-caso 2: Idem 1					
Sub-caso 3: Idem 1					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Configurar varios sitios diferentes y ver su diagrama					
Sub-caso 2:					
Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Verificar que los diagramas de sitio sean dibujados correctamente.					
Sub-caso 2: Verificar que se invoque al editor predeterminado correctamente.					
Sub-caso 3: Verificar que se invoque al navegador predeterminado correctamente.					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3	X				

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 14

Descripción General del caso: Edición del sitio mediante Diagrama de Edición del sitio.					
Sub-caso 1: Agregar Link Dinámico					
Sub-caso 2: Pegar Link Dinámico en una página local					
Sub-caso 3: Eliminar referencia a Link Dinámico					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Configurar administrador para trabajar con un proyecto que tenga páginas Web en lo posible, de lo contrario importar un sitio o agregar páginas manualmente					
Sub-caso 2: Idem 1					
Sub-caso 3: Idem 1					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1:					
Sub-caso 2:					
Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Verificar que el Link Dinámico se agregue correctamente.					
Sub-caso 2: Verificar que se crea la referencia al Link Dinámico correctamente.					
Sub-caso 3: Verificar que se elimine la referencia al Link Dinámico correctamente.					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3	X				

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Administrador

Nº de caso: 15

Descripción General del caso: Edición del sitio mediante Diagrama de Edición del sitio.					
Sub-caso 1: Pegar Link Dinámico en más de una página local					
Sub-caso 2: Borrar un Link Dinámico con referencia a más de una página local					
Sub-caso 3: Eliminar referencia a Link Dinámico con referencia a más de una página local					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Configurar administrador para trabajar con un proyecto que tenga páginas Web en lo posible, de lo contrario importar un sitio o agregar páginas manualmente					
Sub-caso 2: Idem 1					
Sub-caso 3: Idem 1					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1:					
Sub-caso 2:					
Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Verificar que se crea la referencia al Link Dinámico correctamente.					
Sub-caso 2: Verificar que se elimine la referencia al Link Dinámico correctamente junto con sus referencias en todas las páginas donde aparecía referenciado.					
Sub-caso 3: Verificar que se elimine la referencia al Link Dinámico correctamente.					
CONDICIÓN	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3	X				
Sub-caso 4	X				

RESULTADO Ok: SI X NO

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Link Dinámico

Nº de caso: 16

Descripción General del caso: Edición de propiedades del Link Dinámico desde el Diagrama de Edición de Sitio. Sub-caso 1: Cambiar propiedades de un Link Dinámico Sub-caso 2: Cambiar guardas de un Link Dinámico Sub-caso 3: Configurar prioridades mediante asistente					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso) Sub-caso 1: Configurar administrador para trabajar con un proyecto cualquiera Sub-caso 2: Idem 1 Sub-caso 3: Idem 1					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.) Sub-caso 1: Sub-caso 2: Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR). Sub-caso 1: Verificar que las propiedades se hayan cambiado correctamente. Sub-caso 2: Verificar que las guardas se hayan cambiado correctamente. Sub-caso 3: Verificar que las prioridades se hayan cambiado correctamente.					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3	X				

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Link Dinámico

Nº de caso: 17

Descripción General del caso: Edición de propiedades del Link Dinámico desde el editor HTML Microsoft FrontPage. Sub-caso 1: Cambiar propiedades de un Link Dinámico Sub-caso 2: Cambiar guardas de un Link Dinámico Sub-caso 3: Configurar prioridades mediante asistente					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso) Sub-caso 1: Configurar administrador para trabajar con un proyecto cualquiera Sub-caso 2: Idem 1 Sub-caso 3: Idem 1					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.) Sub-caso 1: Sub-caso 2: Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR). Sub-caso 1: Verificar que las propiedades se hayan cambiado correctamente. Sub-caso 2: Verificar que las guardas se hayan cambiado correctamente. Sub-caso 3: Verificar que las prioridades se hayan cambiado correctamente.					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3	X				

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Compilador PHP y Módulo PHP DynamiX

Nº de caso: 18

Descripción General del caso: Presentación del Link Dinámico en browser cliente Sub-caso 1: Incluir un Link Dinámico de tipo "solo texto fijo" Sub-caso 2: Incluir un Link Dinámico de tipo "solo texto" con una única guarda que siempre devuelva una URL cuyos textos asociados sean conocidos. Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso) Sub-caso 1: Configurar administrador para trabajar con un proyecto que contenga un sitio Sub-caso 2: Idem 1 Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.) Sub-caso 1: Agregar el Link Dinámico del caso y compilar el sitio. Sub-caso 2: Idem 1 Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR). Sub-caso 1: Verificar en el navegador del cliente que link se despliegue correctamente con el texto indicado. Sub-caso 2: Verificar en el navegador del cliente que link se despliegue correctamente con los textos asociados a la URL de manera aleatoria cada vez que se invoca la página. Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	<i>Marcar una X</i>	<i>Marcar una X</i>	<i>Sólo si el caso dio ERROR</i>		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3					

RESULTADO Ok: SI X NO

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Compilador PHP y Módulo PHP DynamiX

Nº de caso: 19

Descripción General del caso: Presentación del Link Dinámico en browser cliente Sub-caso 1: Incluir un Link Dinámico de tipo "solo imagen fija" Sub-caso 2: Incluir un Link Dinámico de tipo "solo imagen" con una única guarda que siempre devuelva una URL cuyas imágenes asociados sean conocidas. Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso) Sub-caso 1: Configurar administrador para trabajar con un proyecto que contenga un sitio Sub-caso 2: Idem 1 Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.) Sub-caso 1: Agregar el Link Dinámico del caso y compilar el sitio. Sub-caso 2: Idem 1 Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR). Sub-caso 1: Verificar en el navegador del cliente que link se despliegue correctamente con la imagen indicada. Sub-caso 2: Verificar en el navegador del cliente que link se despliegue correctamente con las imágenes asociadas a la URL de manera aleatoria cada vez que se invoca la página. Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	<i>Marcar una X</i>	<i>Marcar una X</i>	<i>Sólo si el caso dio ERROR</i>		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3					

RESULTADO Ok: SI X NO

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Compilador PHP y Módulo PHP DynamiX

Nº de caso: 20

Descripción General del caso: Presentación del Link Dinámico en browser cliente Sub-caso 1: Incluir un Link Dinámico con propiedad target "Nueva" Sub-caso 2: Incluir un Link Dinámico con propiedad target "Arriba" Sub-caso 3: Incluir un Link Dinámico con propiedad target "Otro" apuntando a un frame particular. El Link Dinámico debe estar en otro frame distinto al frame particular.					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso) Sub-caso 1: Configurar administrador para trabajar con un proyecto que contenga un sitio Sub-caso 2: Idem 1 Sub-caso 3: Idem 1					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.) Sub-caso 1: Agregar el Link Dinámico del caso y compilar el sitio. Sub-caso 2: Idem 1 Sub-caso 3: Idem 1					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR). Sub-caso 1: Verificar en el navegador del cliente que link al ser clickeado abra una nueva ventana con el contenido de la URL destino. Sub-caso 2: Verificar en el navegador del cliente que link al ser clickeado despliegue el contenido de la URL destino en el frame donde se encuentre. Sub-caso 3: Verificar en el navegador del cliente que link al ser clickeado despliegue el contenido de la URL destino en el frame particular indicado.					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3	X				

RESULTADO Ok: SI X NO

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Compilador PHP y Módulo PHP DynamiX

Nº de caso: 21

Descripción General del caso: Presentación del Link Dinámico en browser cliente Sub-caso 1: Incluir un Link Dinámico de tipo imagen con atributos width y height determinados Sub-caso 2: Incluir un Link Dinámico de tipo imagen apuntando a una imagen que no exista y que contenga el atributo alt igual a algún texto. Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso) Sub-caso 1: Configurar administrador para trabajar con un proyecto que contenga un sitio Sub-caso 2: Idem 1 Sub-caso 3:					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.) Sub-caso 1: Agregar el Link Dinámico del caso y compilar el sitio. Sub-caso 2: Idem 1 Sub-caso 3:					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR). Sub-caso 1: Verificar en el navegador del cliente que link se despliegue con una imagen del tamaño indicado. Sub-caso 2: Verificar en el navegador del cliente que link se despliegue con el texto indicado en la propiedad alt. Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	<i>Marcar una X</i>	<i>Marcar una X</i>	<i>Sólo si el caso dio ERROR</i>		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3					

RESULTADO Ok: SI X NO

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Compilador PHP y Módulo PHP DynamiX

Nº de caso: 22

Descripción General del caso: Comportamiento del Link Dinámico en navegador cliente					
Sub-caso 1: Incluir un Link Dinámico sin guardas y con consulta por defecto en blanco					
Sub-caso 2: Incluir un Link Dinámico sin guardas y con una consulta por defecto válida					
Sub-caso 3: Incluir un Link Dinámico sin guardas y con una consulta por defecto inválida					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso)					
Sub-caso 1: Configurar administrador para trabajar con un proyecto que contenga un sitio					
Sub-caso 2: Idem 1					
Sub-caso 3: Idem 1					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.)					
Sub-caso 1: Agregar el Link Dinámico del caso y compilar el sitio.					
Sub-caso 2: Idem 1					
Sub-caso 3: Idem 1					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR).					
Sub-caso 1: Verificar en el navegador del cliente que link no se despliegue.					
Sub-caso 2: Verificar en el navegador del cliente que link se despliegue apuntando hacia la URL resultado de la consulta por defecto.					
Sub-caso 3: Verificar en el navegador del cliente que link no se despliegue.					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3	X				

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Compilador PHP y Módulo PHP DynamiX

Nº de caso: 23

Descripción General del caso: Comportamiento del Link Dinámico en navegador cliente Sub-caso 1: Incluir un Link Dinámico con guardas y con consulta por defecto en blanco, ingresando con un usuario que no cumpla ninguna guarda. Sub-caso 2: Incluir un Link Dinámico con guardas ingresando con usuario que cumpla una única guarda. Sub-caso 3: Incluir un Link Dinámico con guardas de distintas prioridades ingresando con usuario que cumpla más de una guarda.					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso) Sub-caso 1: Configurar administrador para trabajar con un proyecto que contenga un sitio Sub-caso 2: Idem 1 Sub-caso 3: Idem 1					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.) Sub-caso 1: Agregar el Link Dinámico del caso y compilar el sitio. Sub-caso 2: Idem 1 Sub-caso 3: Idem 1					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR). Sub-caso 1: Verificar en el navegador del cliente que link no se despliegue. Sub-caso 2: Verificar en el navegador del cliente que link se despliegue apuntando hacia la URL resultado de la consulta por defecto. Sub-caso 3: Verificar en el navegador del cliente que link se despliegue apuntando hacia alguna de las URL resultado de las guardas válidas. Ingresar a la página varias veces y verificar que las URLs de las guardas válidas sean mostradas con las prioridades asignadas.					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3	X				

RESULTADO Ok: SI _X_ NO ____

Generación de sitios Web paramétricos adaptativos

Apéndice I – Casos de prueba de la aplicación

Módulo: Compilador PHP y Módulo PHP DynamiX

Nº de caso: 24

Descripción General del caso: Comportamiento del Link Dinámico en navegador cliente Sub-caso 1: Incluir un Link Dinámico con algunas guardas cuyas condiciones estén mal formadas. Sub-caso 2: Incluir un Link Dinámico con alguna guarda cuya consulta esté mal formada e ingresar con un usuario que cumpla únicamente con las condiciones de esta. Sub-caso 3:					
Preparación para el caso: (condiciones necesarias para que sea posible la ejecución del caso) Sub-caso 1: Configurar administrador para trabajar con un proyecto que contenga un sitio Sub-caso 2: Idem 1 Sub-caso 3: Idem 1					
Procedimiento de ejecución: (pasos a seguir para la ejecución del caso.) Sub-caso 1: Agregar el Link Dinámico del caso y compilar el sitio. Sub-caso 2: Idem 1 Sub-caso 3: Idem 1					
Resultados esperados: (detalle claro de que resultados se deben obtener luego de la ejecución del caso, para que al momento de su ejecución sea sencillo verificar si el caso fue OK o dio ERROR). Sub-caso 1: Verificar en el navegador del cliente que el link no se despliegue. Sub-caso 2: Verificar en el navegador del cliente que el link no se despliegue. Sub-caso 3:					
CONDICION	OK	ERROR	DESCRIPCIÓN DEL ERROR	Tipo	Fecha y Hora
	Marcar una X	Marcar una X	Sólo si el caso dio ERROR		
Sub-caso 1	X				
Sub-caso 2	X				
Sub-caso 3					

RESULTADO Ok: SI _X_ NO ____

Apéndice J

Glosario de Términos

- **Hipervínculo** (Vínculo, Link local, Link externo, Link remoto, Destino local, Destino remoto):
Se define en HTML como
`<a href.= "dirección_url_pagina_html_destino"></a>`
y es un puntero a otro documento en la red.
- **Página Web** (pagina html, archivo .html):
Documento escrito en lenguaje HTML
 - o **Página local**: pagina Web almacenada en el mismo servidor Web.
 - o **Página externa**: pagina Web almacenada en otro servidor
 - o **Página dinámica**: pagina cuyo contenido varía entre accesos.
- **ActiveX**, OCX. Tecnología Microsoft para el desarrollo de componentes, parte de COM+.
- **Link Dinámico**: objeto que introduce comportamiento dinámico en las paginas.
 - o **DynamiX** (OCX, ActiveX , objeto DynamiX): instancia link para este proyecto de un objeto dinámico
- **Administrador**: aplicación de control
DynamiX Manager (Administrador DynamiX): instancia del administrador para este proyecto
- **Sistema de guardas** (GUARDA): cada guarda se compone de
 - o **Condición DynamiX**: condición que se evalúa a verdadero o falso para ver que consulta DQL se ejecuta.
 - o **Prioridades** (Probabilidades): Prioridad que se asigna a la guarda para usarse en caso de necesitar elegir entre varias guardas.
 - o **DQL**: Lenguaje de especificación de consultas para DynamiX
 - o **Consulta DQL**: Instancia de DQL.
- **Sitio dinámico**: sitio compuesto por una o más paginas dinámicas
 - o **Sitio DynamiX**: instancia de sitio dinámico para este proyecto que usa DynamiX
- **Servidor Web** (Servidor http): servidor de contenidos multimedia para Internet.
- **Texto e imagen asociados** (Texto e imagen dinámicos): texto e imagen que esta relacionado con una Pagina Web en el DynamiX Manager.
- **Jerarquia de grupos de paginas** (Jerarquia de paginas), Grupo de paginas: agrupación de paginas web.

Generación de sitios Web paramétricos adaptativos

Apéndice J – Glosario de Términos

- **Traducir sitio** (Compile site, Generate site): el acto de convertir un sitio DynamiX a un lenguaje server side.
- **Lenguaje Server Side** (language embedded, language dynamic): lenguaje de programación para páginas web que se ejecuta en el servidor. PHP y ASP son ejemplos de esta tecnología.
- **Java Servlet** (Servlet): Aplicación Java que se ejecuta en el servidor web.
- **Parámetros de usuario** (Properties of user): Conjunto de datos del usuario que se obtiene mediante un formulario.
- **URL** (URI): Universal Resource Locator (un subconjunto de Universal Resource Identifier) Identificador único de todos los documentos en la red.
- **Tag**: Marcas utilizadas en los lenguajes de marcas como por ejemplo HTML, XML, etc.
- **Usuario** (Browser): Entidad que accede al sitio web.

REFERENCIAS

- [1] "HTML 4.0 Specification", W3C Recommendation REC-html40-971218
Autores: Raggett, D., Le Hors, A. and I. Jacobs
<http://www.w3.org/TR/REC-html40>
Fecha: Diciembre 1997.
- [2] URL - RFC 1738. Uniform Resource Locators (URL)
Autores: T. Berners-Lee, L. Masinter & M. McCahill
<http://www.w3.org/Addressing/rfc1808.txt>
Fecha: Diciembre 1994.
- [3] HTTP - RFC 2616. "Hypertext Transfer Protocol -- HTTP/1.1"
Autores: Gettys, J., Fielding, R., Mogul, J., Frystyk, H., Masinter, L.,
Leach, P. and T. Berners-Lee
<http://www.w3.org/Protocols/rfc2616/rfc2616.txt>
Fecha: Junio 1999.
- [4] JAVA - www.java.sun.com
Fecha: 20/11/2001
- [5] Microsoft Visual Basic 6 Developers Workshop
Edición: 5^{ta}
Año: 1998
Autores: Jhon Clark, Craig Jeff Webb
Editorial: Microsoft Press
- [6] Professional Active Server Pages 3.0
Autores: [Alex Homer](#), [David Sussman](#), [Brian Francis](#), [George Reilly](#),
[Esposito](#), [Dino Esposito](#), [Andrea Chiarelli](#), [Bill Kropog](#), [Craig McQueen](#),
[Godfrey Nolan](#), [Simon Robinson](#), [John Schenken](#), [Kent Tegel](#)
Editorial: Wrox
- [7] PHP - www.php.net
Fecha: 20/11/2001
- [8] Programación para el Web
Año: 1998
Autores: Kris Jansa, Suleiman Calan, Steve Weakley
Editorial: Mc Graw Hill
- [9] Java en pocas palabras.
Edición: 2^{da}
Año: 1998
Autor: David Flanagan
Editorial: O'Reilly

Generación de sitios Web paramétricos adaptativos

REFERENCIAS

- [10] HTML The Definitive Guide
Edición: 3^{ra}
Año: 1999
Autores: Chuck Musciano, Bill Kennedy
Editorial: O'Reilly
- [11] Web Mining: definiciones y aplicaciones
www.webmining.cl/home.asp
Fecha: 24/02/2001
Autor: [Mariano Silva V.](#)
- [12] Web Mining: Pattern Discovery from World Wide Web Transactions
Autores: Bashad Mobasher, Namit Jain, Eui-Hong (Sam) Han, Jaideep Srivastava {mobasher,njain,han,srivasta}@cs.umn.edu
Department of Computer Science University of Minesota Minneapolis, MN 55455, USA
Technical Report 96-050 (September 3, 1996)
- [13] Proyecto de fin de carrera "*Análisis de patrones de búsqueda con Web Mining*".
Instituto de computación
Facultad de Ingeniería
Universidad de la República
Año: 2001
Autores: Leticia Peres, Luis do Rego
- [14] ODBC - www.microsoft.com/data/odbc/
Fecha: 20/11/2001
- [15] Sistemas de Bases de Datos. Conceptos Fundamentales.
Edición: 2^{da}
Año: 1997
Autores: Elmasri, Navathe
Editorial: Addison-Wesley Iberoamericana
- [16] Computer Networks
Edición: 3era
Año: 1996
Autor: Andrew S. Tanenbaum
Editorial: Prentice Hall
- [17] Concurrency in Ada
Edición: 2da
Autores: Alan Burns, Andy Wellings
Editorial: Cambridge University Press
- [18] Personalización de sitios Web
http://hotwired.lycos.com/collections/internet_law/6.05_one_one_pr.html

Generación de sitios Web paramétricos adaptativos

REFERENCIAS

- [19] Cookies
<http://www.netscapeworld.com/netscapeworld/nw-02-1997/nw-02-cookiehowto.html>
Fecha: 20/11/2001
- [20] Cookies - HTTP Cookie: Client Side State
http://www.netscape.com/newsref/std/cookie_spec.html
- [21] Cookies - RFC 2965. "HTTP State Management Mechanism"
Autores: D. Kristol, L. Montulli
<http://www.ietf.org/rfc/rfc2965.txt>
Fecha: Octubre 2000.
- [22] ODBC - iODBC: <http://www.iodbc.org/odbc-phpHOWTO.html>
- [23] Web Page Categorization and Feature Selection Using Association Rule and Principal Component Clustering
Autores: Jerome Moore, Eui-Hong (Sam) Han, Daniel Boley, MARIa Gini, Robert Gross, Kyle Hastings, George Karypis, Vipin Kumar, and Bamshad Mobasher
Department of Computer Science and Engineering / Army HPC Research Center University of Minnesota, Minneapolis, MN 55455
- [24] Modeling of Web Robot Navigation Patterns
Autores: Pang-Ning Tan and Vipin Kumar
Department of Computer Science, University of Minnesota, 200 Union Street SE, Minneapolis, MN 55455. {ptan,kumar}@cs.umn.edu
- [25] Data Preparation for Mining World Wide Web Browsing Patterns
Autores: Robert Cooley, Bashad Mobasher, and Jaideep Srivastava
Department of Computer Science and Engineering University of Minnesota, 4-192 EECS Bldg., Union St. SE Minneapolis, MN 55455, USA
- [26] Adaptive Siter: Automatically Learning from User Access Patterns
Autores: Mike Perkowitz and Oren Etzioni
Department of Computer Science and Engineering , Box 352350, University of Washington, Seattle, WA 98195-2350
{map, etzioni}@cs.washington.edu
- [27] WUM: A Web Utilization Miner
Autores: Myra Spiliopoulou, Institut für Wirtschaftsinformatik, HU Berlin, myra@wiwi.hu-berlin.de, Lukas C. Faulstich, Institut für Informatik, FU Berlin, <http://www.inf.fu-berlin.de/~faulstic>, faulstic@inf.fu-berlin.de
- [28] PHP versus ASP.
Fecha: 20/06/2001.
http://php.weblogs.com/php_vs_asp
- [29] 7 Reasons Why PHP is Better than ASP
Fecha: 20/06/2001.
http://php.weblogs.com/php_asp_7_reasons

Generación de sitios Web paramétricos adaptativos

REFERENCIAS

- [30] Switching to PHP: What's the Language Difference?
Fecha: 20/06/2001.
<http://www.zend.com/zend/art/langdiff.php>
- [31] Ask Tim! PHP vs. ASP
Fecha: 20/06/2001.
http://www.oreilly.com/ask_tim/asp_php.html
- [32] PHP vs. ASP
Fecha: 20/06/2001.
http://cyberiot.com/php_vs_asp.htm