UNVERSIDAD DE CHILE
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS
DEPARTAMENTO DE INGENIERIA INDUSTRIAL

DISEÑO Y CONSTRUCCIÓN DE UN WEB WAREHOUSE PARA ALMACENAR INFORMACIÓN EXTRAÍDA A PARTIR DE DATOS ORIGINADOS EN LA WEB

MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL INDUSTRIAL

ROBERT CERCÓS BROWNELL

PROFESOR GUÍA:

JUAN D. VELASQUEZ SILVA

MIEMBROS DE LA COMISIÓN:

MATÍAS COCIÑA VARAS

PABLO ROMÁN ASENJO

SANTIAGO DE CHILE

ABRIL 2008

RESUMEN DE LA MEMORIA
PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL INDUSTRIAL
POR : ROBERT CERCÓS BROWNELL
FECHA: 21/04/2008
PROF. GUIA: JUAN D. VELASQUEZ S.

Diseño y construcción de un web warehouse para almacenar información extraída a partir de datos de la web

El objetivo general del presente trabajo es diseñar y construir un respositorio de información respecto del uso, contenido y estructura de un sitio web. Lo anterior tiene por finalidad el aprovechamiento de los datos que genera la navegación a través de un sitio, mediante la extracción de información que sirva de apoyo para la toma de decisiones en función de mejorar su estructura y contenido.

La Web se ha convertido en un canal con altas potencialidades de uso para las organizaciones, mostrando importantes ventajas, sobretodo en sus aplicaciones en ventas y marketing. Es así como se ha generado una creciente necesidad de comprender las preferencias de los visitantes de un sitio, de manera de atraer su atención e, idealmente, transformarlo en cliente. Sin embargo, debido a la gran cantidad de datos generados a partir de la navegación y del contenido de un sitio, se hace muy complejo sacar conclusiones a partir de ellos.

Para dar inicio a esta investigación se estudiaron los algoritmos existentes para el procesamiento de los datos de la Web, además de los distintos modelos de almacenamiento para la información construida a partir de ellos.

En base a lo anterior, se desarrolló un modelo genérico para almacenar, procesar y presentar la información. Ésta se extrae a partir de datos que se obtienen mediante una estrategia no invasiva para los visitantes, utilizando para su almacenamiento la arquitectura *data warehouse*, que permite mantener información limpia, consolidada y confiable a partir de una gran cantidad de datos provenientes de múltiples fuentes.

Posteriormente, el modelo desarrollado se aplicó a un sitio web real relacionado a la industria bancaria, de manera de probar su correcto funcionamiento y utilidad.

Como resultado, se concluyó que la arquitectura implementada es efectiva para el análisis estadístico de los datos, siendo el porcentaje de conversión por objetivos el indicador más relevante para la medición del desempeño de un sitio web, pudiendo transformarse, incluso, en una dimensión del modelo de información. Se recomienda que, en trabajos futuros, se constraste los resultados de la operación de este repositorio con el de otras estrategias de obtención de la información.

INDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

INDICE DE TABLAS

CAPÍTULO 1 - INTRODUCCIÓN

En el contexto de una economía centrada crecientemente en el uso de tecnologías de la información, la World Wide Web se ha posicionado como un elemento estratégico en muchos negocios, llegando a ser incluso el *core business* de grandes empresas, como es el caso de Amazon, Google, eBay, entre otras. Las organizaciones, intuyendo esta tendencia, han intentado utilizarla como medio de atracción de clientes, hecho que ha generado una creciente necesidad de comprender el comportamiento y preferencias de los usuarios de un sitio web.

Lo anterior, sumado a la necesidad de los sitios de ser actualizados y mejorados frecuentemente, hace indispensable el contar con herramientas tecnológicas que apoyen la toma de decisiones respecto a cómo realizar los cambios en la estructura y contenido de un sitio web. Esto permite conocer mejor las preferencias de los usuarios y mejorar así su navegación de acuerdo a sus necesidades de información, lo que puede constituir una fuente importante de ventajas competitivas para una organización.

Para ello se requiere contar con información consolidada y confiable, de manera que un usuario de gestión, interesado o responsable del funcionamiento de un sitio, pueda sacar conclusiones y decidir, en función de esa información, las acciones a seguir en su sitio web en términos de su estructura y contenido. Lo anterior se puede satisfacer mediante la construcción de un repositorio de información originada en la Web (WIR) utilizando metodología de data warehousing para su desarrollo [43], que permita almacenar información enfocada a temas de relevancia para el usuario final. Dicha solución recibe el nombre de data web warehouse o, simplemente, webhouse [27].

1.1 DESCRIPCIÓN DEL TRABAJO REALIZADO Y ESTRUCTURA DEL INFORME

Esta investigación consta de dos componentes altamente acopladas: un trabajo teórico de investigación respecto de repositorios de información para la web, cuyo producto final es el desarrollo de un modelo conceptual, y una aplicación práctica del modelo genérico desarrollado. Esto último se hizo en base a algoritmos ampliamente utilizados e implementados, por lo que sólo se requirió de la construcción y poblamiento de la base de datos, junto con el posterior diseño y creación de una interfaz de consulta de la información almacenada.

En el capítulo 1 se introducen los aspectos generales de la investigación, incluyendo contribuciones y objetivos. En el capítulo 2 se hace una revisión del

sustento teórico de la investigación y el desarrollo que ésta requirió. En el capítulo 3 se muestra la metodología utilizada en su desarrollo. En el capítulo 4 se construye un modelo genérico de webhousing, estableciendo los requerimientos y componentes que debieran conformarlo, tanto en términos de la lógica de almacenamiento, procesamiento de los datos y perfiles de usuario, entre otros. En el capítulo 5 se documenta la aplicación práctica del modelo genérico, incluyendo la definición de la arquitectura del sistema, como también los resultados de las pruebas. Por último, en el capítulo 6 se mencionan las principales conclusiones obtenidas a partir de la investigación, como también algunas recomendaciones respecto del trabajo futuro en esta línea de estudio.

1.2 CONTRIBUCIONES

La generación de un modelo genérico para la implementación de un data webhouse, además de la construcción de un datamart real a partir de datos emanados de la Web, representan las principales contribuciones de este trabajo.

A diferencia de otros desarrollos para obtener información acerca del uso, contenido y estructura de sitios web, los datos no provienen de métodos invasivos para los visitantes de un sitio, sino de una estrategia reactiva para la construcción de la información. Lo anterior tiene la ventaja de salvaguardar la privacidad de los visitantes, sin mayor pérdida de posibilidades en cuanto a los posibles análisis que se pueden efectuar respecto de un sitio.

1.3 OBJETIVOS

1.3.1 OBJETIVO GENERAL

Diseñar y construir un repositorio de información basado en la arquitectura Data Warehouse para el caso particular de datos originados en la Web.

1.3.2 OBJETIVOS ESPECÍFICOS

1. Revisar el estado del arte en repositorios de información.
2. Diseñar y desarrollar un modelo conceptual para el almacenamiento de información generada a partir de las distintas fuentes de datos.
3. Diseñar e implementar un proceso de limpieza, transformación y carga (ETL) de datos basado en los algoritmos existentes.
4. Aplicar la metodología de construcción de un Web Warehouse a un sitio web real.
5. Diseñar y construir un prototipo de interfaz de consulta para el usuario final.

CAPÍTULO 2 - MARCO CONCEPTUAL

Para abordar y comprender el problema de investigación se requiere del estudio de distintos tópicos que, en este informe, fueron clasificados en cuatro áreas temáticas: 1) World Wide Web, 2) Sistemas de almacenamiento, procesamiento y análisis de datos, 3) Análisis del desempeño de un sitio web y 4) La Web como canal de marketing y ventas.

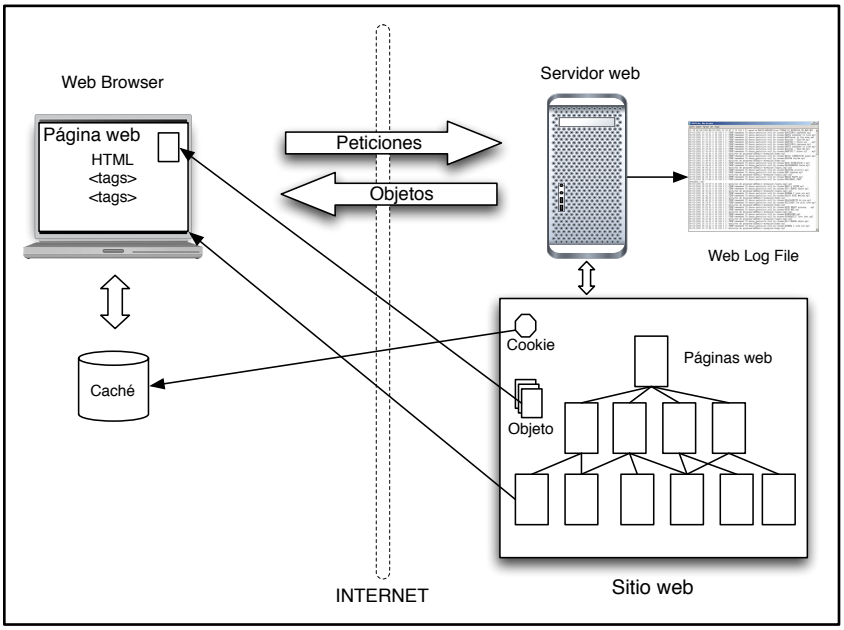
2.1 WORLD WIDE WEB

La World Wide Web [8] nace con la idea de Tim Berners-Lee de generar hipertexto[1] compartido. Esta idea, en la práctica, es un sistema de documentos de hipertexto a los cuales se accede a través de Internet. Los documentos pueden representar texto, imágenes, sonido, etc.

2.1.1 FUNCIONAMIENTO DE LA WEB

Está basado en el paradigma cliente/servidor [20], donde un cliente web pide objetos, es decir, los archivos que conforman un sitio, a un servidor web mediante el protocolo HTTP[2]. Lo anterior comienza en el momento en que el usuario ingresa la

Figura 1: Interacción entre un servidor y un cliente web.



Fuente: Elaboración Propia

[1] "Tecnología que organiza una base de información en bloques distintos de contenidos, conectados a través de una serie de enlaces cuya activación o selección provoca la recuperación de la información" [Díaz et al, 1996]

[2] Hyper Text Transfer Protocol.

dirección de un sitio web en el navegador (URL), o bien, hace clic en algún hipervínculo de otra página. Luego, una petición es enviada al servidor para que, finalmente, el servidor devuelva el objeto pedido por el cliente.

En la Figura 1, se muestra la interacción entre el servidor y el cliente web, donde cada petición de este último queda registrada en un archivo en el servidor (*web log file*). Al cliente web se le devuelve el contenido de la página en un texto en formato HTML[3] que debe ser traducido por el navegador web (*browser*). En él se especifican cuáles son los objetos que componen la página, los que posteriormente le son transmitidos. Finalmente, el navegador presenta el resultado en la pantalla del cliente con todos los objetos que la componen. En algunos casos, el servidor web envía e introduce un archivo llamado *cookie* en el computador del cliente. Éste le permite identificar de alguna manera al usuario y conocer algunos aspectos de su navegación, por ejemplo, reconocer cuando un visitante ya ha visitado anteriormente el sitio. Por último, dado que muchas veces se solicitan los mismos objetos web para ser utilizados en distintas páginas, o bien, se solicita frecuentemente una página en particular, algunos elementos de la navegación se almacenan en memoria del cliente web, de manera de acelerar el proceso de carga de las páginas.

2.1.2 DATOS ORIGINADOS EN LA WEB

Web logs

Los datos relacionados con cada petición de objetos web quedan registrados en el archivo de web log. Cada registro contiene información acerca del comportamiento de navegación de los usuarios, en particular, el tiempo que cada usuario gastó en cada página y la secuencia de páginas que visitó.

Figura 2: Ejemplo de un archivo de log

```
200.83.72.166 - - [31/Mar/2003:23:45:20 -0400] "GET / HTTP/1.1" 200 20160 "-" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)" -
200.83.72.166 - - [31/Mar/2003:23:45:21 -0400] "GET /estilos_.css HTTP/1.1" 304 0 "http://p1.com/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)" -
200.83.72.166 - - [31/Mar/2003:23:45:21 -0400] "GET /images/bgr3.gif HTTP/1.1" 304 0 "http://p2.com/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)" -
200.83.72.166 - - [31/Mar/2003:23:45:21 -0400] "GET /medios/portada/log_HOME.gif HTTP/1.1" 200 6296 "http://p5.com/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)" -
200.83.72.166 - - [31/Mar/2003:23:45:21 -0400] "GET /images/btn_ingresar_home.gif HTTP/1.1" 304 0 "http://p4.com/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)" -
200.83.72.166 - - [31/Mar/2003:23:45:21 -0400] "GET /infoeco/dummy HTTP/1.1" 200 - "http://p5.htm/" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)"
200.28.50.169 - - [31/Mar/2003:23:45:23 -0400] "GET / HTTP/1.1" 200 20160 "-" "Mozilla/4.0 (compatible; MSIE 5.14; Mac_PowerPC)" -
200.28.50.169 - - [31/Mar/2003:23:45:41 -0400] "GET /infoeco/dummy.mopans?/info.htm HTTP/1.1" 200 - "-" "Mozilla/4.0 (compatible; MSIE 5.14; Mac_PowerPC)"
200.246.203.66 - - [31/Mar/2003:23:45:46 -0400] "GET /estilos_.css HTTP/1.0" 304 - "http://www.p5.cl" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)" -
200.30.223.248 - - [31/Mar/2003:23:45:48 -0400] "GET /images/bgr3.gif HTTP/1.1" 200 224 "-" "Mozilla/4.0" -
200.30.223.248 - - [31/Mar/2003:23:45:48 -0400] "GET /images/spacer.gif HTTP/1.0" 200 45 "-" "Mozilla/4.0" -
200.30.223.248 - - [31/Mar/2003:23:45:48 -0400] "GET /medios/portada/log_HOME.gif HTTP/1.0" 200 6296 "-" "Mozilla/4.0"
200.30.223.248 - - [31/Mar/2003:23:45:48 -0400] "GET /images/btn_ingresar_home.gif HTTP/1.0" 200 187 "-" "Mozilla/4.0" -
200.30.223.248 - - [31/Mar/2003:23:45:48 -0400] "GET /infoeco/dummy./info.htm HTTP/1.0" 200 - "-" "Mozilla/4.0" TRANS=vt_infoeco&vt_infoeco=transa/infoeco/info.htm
200.30.223.248 - - [31/Mar/2003:23:45:48 -0400] "GET /images/tab.gif HTTP/1.0" 200 89 "-" "Mozilla/4.0" -
200.246.203.66 - - [31/Mar/2003:23:45:49 -0400] "GET /medios/portada/log_HOME.gif HTTP/1.0" 200 6296 "http://www.p7.cl" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)"
200.246.203.66 - - [31/Mar/2003:23:45:49 -0400] "GET /medios/banner.jpg HTTP/1.0" 200 11130 "http://www.p10.cl" "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)" -
200.28.251.192 - - [31/Mar/2003:23:45:53 -0400] "GET /medios/b_.jpg HTTP/1.1" 200 15134 "http://www.p8.cl" "Mozilla/4.0 (compatible; MSIE 5.5; Windows 98)" -
```

Fuente: Elaboración Propia

[3] Hyper Text Markup Language

La estructura de los datos que conforman los web logs está dada por el estándar definido por la W3C[4]. Por su parte, la estructura del archivo de log tiene generalmente los siguientes campos:

- Dirección IP: corresponde a la dirección de Internet del visitante.
- Identity: información de identificación entregada por los clientes.
- Tiempo: Fecha y hora en que se completó una respuesta a una petición HTTP.
- Tipo de requerimiento, protocolo y archivo pedido: identifica la petición hecha por el cliente y la versión del protocolo de transferencia.
- Status: Código representado mediante un número entero, donde se señala el estado de una petición. Algunos de los más comunes son: 200 ("respuesta exitosa"), 404 ("no se encontró la página web"), 403 ("acceso prohibido a la página").
- Bytes: el número de bytes entregados en una petición.
- Referrer: una línea de texto enviada por el cliente que indica la fuente original desde donde se genera la petición.
- User-Agent: nombre y versión del software que ejecuta la petición.

Los datos que componen el archivo de registro en el servidor representan la principal fuente de información que posee una organización para conocer el comportamiento de los usuarios de un sitio.

Contenido de las páginas web

Lo conforman el conjunto de imágenes, multimedia y texto que componen las páginas de un sitio.

Las páginas web corresponden a documentos escritos en código HTML, estructurados mediante etiquetas o *tags* que son interpretadas por el navegador web. Esta interpretación es gradual, es decir, no requiere de compilación. Esto tiene entre sus consecuencias que, en caso de existir un problema de conectividad mientras se transmite una página, se cargan todos los objetos que se alcanzaron a leer.

Estructura de hipervínculos

La conexión entre las distintas páginas de un sitio, como también con páginas fuera del mismo, son otra fuente importante de información respecto a un sitio web, debido a que dan cuenta de su estructura interna y de sus lugares de fuga.

[4] Consorcio World Wide Web, preocupado de desarrollar estándares web. (www.w3c.es)

Esta relación se representa mediante una etiqueta particular en el código HTML, que permite que un usuario se traslade hacia la dirección web asociada a ese objeto al hacer clic sobre él.

2.1.3 INFORMACIÓN A PARTIR DE LOS DATOS ORIGINADOS EN LA WEB (WEB DATA)

A partir de los datos provenientes del uso de un sitio y las páginas que lo componen, se puede obtener información relevante acerca del desempeño del sitio y, luego del análisis experto de la información, hacer los cambios necesarios para mejorarlo.

Los ámbitos en donde se puede recabar información útil para conseguir lo anterior son:

- Información acerca de la navegación en un sitio mediante la reconstrucción de las sesiones.
- Información respecto del contenido del sitio, analizando el peso relativo de ciertos conceptos en la totalidad del texto.
- Información respecto a los intereses de los usuarios observando el tiempo gastado en cada una de las componentes de su sesión.

Reconstrucción de sesiones

Al proceso de separar la actividad de los todos los usuarios de un sitio en sesiones individuales se le llama sesionización [17].

Dependiendo de los métodos utilizados para la reconstrucción, se pueden distinguir dos estrategias de sesionización:

1) **Estrategia proactiva**, que identifica usuarios mediante el uso de cookies que son enviadas al navegador del visitante cuando éste ingresa por primera vez a un sitio. De esta manera se puede identificar cuándo un usuario ingresa nuevamente y así diferenciarlo del resto y reconstruir su sesión. Esta estrategia posee algunas desventajas relacionadas con el respeto de la privacidad de los usuarios de un sitio y la facilidad de detección y desactivación de las cookies.

2) **Estrategia reactiva,** que utiliza exclusivamente los datos contenidos en los registros de peticiones en el servidor web para ejecutar la reconstrucción de sesiones.

Se debe considerar algunas fuentes de error al reconstruir sesiones basados en los archivos de registro:

- **Uso de concentradores o NAT**, como *routers, firewalls*, etc. Estos dispositivos ocultan la dirección IP real de una red mediante una dirección externa idéntica para todos los equipos que se conectan a través del mismo dispositivo. Luego, se puede tener muchos registros asociados a una misma dirección IP pero que, en realidad, corresponden a muchos usuarios distintos.
- ***Web Crawlers* o *Spider robots*,** que corresponden a programas que recorren los sitios que componen la Web de manera automatizada, con el fin de guardar información respecto de ellos. Son utilizados principalmente por buscadores web como Google, Yahoo o Altavista para indexar las distintas páginas y agregarlas a sus bases de datos. Al no corresponder a usuarios humanos, no son relevantes para el análisis del uso de un sitio web.

Existen heurísticas para mejorar la reconstrucción de sesiones considerando las dificultades anteriores. Por ejemplo, se considera que una sesión no dura más de 30 minutos [39] (heurística orientada al tiempo). Además, se asume que un usuario llega a una página mediante un hipervínculo encontrado en otra página, por lo que si no hay conexión entre las páginas previamente visitadas y la página analizada, se considera parte de otra sesión (heurística orientada a la navegación).

A su vez, existen condiciones mínimas para que una sesión sea considerada como real [40]. Sea *L* un set de registros en el servidor y $R = \{r_1,...,r_n\}$ el set inicial de sesiones encontradas en *L* luego del proceso de sesionización.

Las condiciones mínimas para que r_i sea una sesión real son:

1) Estar compuesta por objetos pedidos durante la sesión ordenadas por tiempo. Luego,

2) $\forall r_i \in R, \forall j = 2,...,\text{largo}(r_i), r_{i,j}.\text{timestamp} > r_{i,j-1}.\text{timestamp}$

3) Sólo objetos pedidos en L pueden aparecer en R, i.e.,
$\bigcup_{r_i \in R} (\bigcup_{j=1}^{\text{largo}(r_i)} r_{i,j}) = L$

4) Cada petición en L pertenece exactamente a una sesión de R, i.e.,
$\forall r_i \in R, \forall j = 1,...,\text{largo}(r_i), \neg \exists i' = i, j' / r_{i,j} = r_{i',j'}$

De esta manera, el proceso de sesionización se puede caracterizar mediante la siguiente secuencia [43]:

1) Selección de los registros cuya URL apunta a objetos web que poseen contenido de texto.
2) Agrupación de registros por dirección IP y agente.
3) Ordenamiento de los grupos por hora y fecha.
4) Eliminación de sesiones con duración mayor a 30 minutos.

Análisis del contenido de un sitio web: *Vector Space Model*

Para entender cuáles son los conceptos que más atraen a los visitantes se requiere del análisis del contenido de las páginas web. Dicho contenido puede ser organizado de manera no estructurada (texto plano), semi-estructurada (e.g. HTML) y estructurada (e.g. tablas, bases de datos) [31].

Para facilitar el análisis del contenido, los documentos son usualmente representados usando el *vector space model* [1, 4, 35], que es, básicamente, la representación vectorial del contenido texto asociado a pesos relativos en el documento.

El primer paso es extraer del texto las palabras que aportan significado al texto y llevarlas a su raíz semántica (e.g. las palabras “soy”, “eres” y “son” se transforman en “ser”). A este proceso se le llama *stemización*.

Luego se procede a la construcción de los vectores: Sea R el número de palabras distintas extraídas de un set de documentos y Q el número de documentos que componen dicho set. La representación vectorial del sitio web puede ser una matriz M de RxQ donde cada componente $m_{i,j}$ representa el peso de la palabra i en el documento j. Dicho peso debe reflejar, además, la importancia relativa de la palabra en el documento entero. Para ello se ocupa la frecuencia inversa en el documento (IDF), definida como $IDF = log(Q/n_i)$, donde n_i es el número de documentos en que aparece la palabra i. A su vez, se puede incorporar el hecho de que, en el caso de una página web, algunas palabras poseen más importancia relativa debido a que forman parte del título, están destacadas, o bien, forman parte de un hipervínculo [44]. Para ello, se construye un vector SW de dimensión R en donde cada componente sw_i representa el peso adicional de la *i-ésima* palabra. Estas palabras se pueden distinguir gracias a las etiquetas HTML que caracterizan cada uno de los casos mencionados.

De esta manera, el peso $m_{i,j}$ se define como:

$$m_{ij} = f_{ij} * (1 + sw_i) * \log(Q / n_i)$$

Donde $f_{i,j}$ es el número de veces que aparece la palabra i en el documento j.

User Behavior Vector

Con el objeto de registrar la secuencia de navegación de un visitante, además de algunos datos de cada una de sus decisiones, se construye un vector de comportamiento (UBV) [45] para cada sesión:

$$v_i = [(p_1,t_1),...,(p_n,t_n)]$$

Donde p_i es la i-ésima página visitada por la sesión s y t_i el tiempo en que el visitante permaneció en ella.

La construcción de este vector facilita la manipulación de los datos relevantes para el análisis de las sesiones, permitiendo acercarse a una comprensión respecto de las preferencias de los visitantes.

2.2 SISTEMAS DE ALMACENAMIENTO, PROCESAMIENTO Y ANÁLISIS DE DATOS.

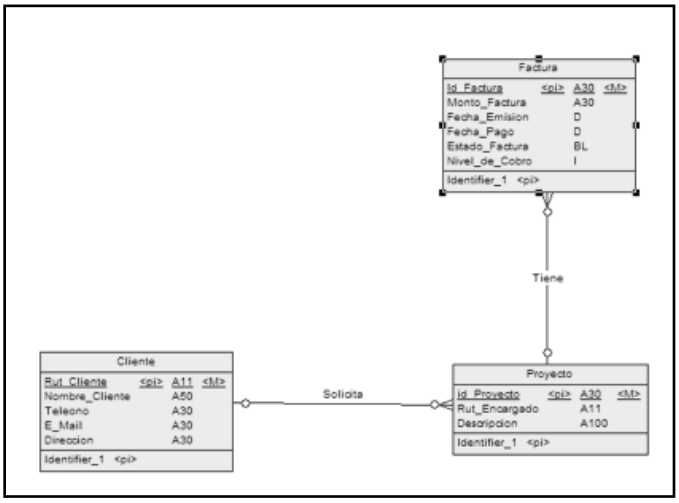
Con el aumento en la capacidad de almacenamiento y velocidad de procesamiento de datos ocurrido a comienzos de la década de los 80, la posibilidad de utilizar los datos generados por los sistemas operacionales para apoyar la toma de decisiones se hizo cada vez más asequible para las empresas y, en consecuencia, se ha transformado crecientemente en una necesidad para mantener una posición competitiva en muchas industrias o bien, en una fuente de nuevas ventajas competitivas para una organización.

2.2.1 MODELO RELACIONAL DE ALMACENAMIENTO DE DATOS

El enfoque con que usualmente se almacenan los datos operacionales está basado en el Modelo Entidad Relación introducido por el Dr. Edgar F. Codd en 1970 [15], que permite, mediante una abstracción del negocio, diseñar un modelo de datos orientado a eliminar redundancias y responder cualquier tipo de pregunta respecto de los datos.

En la Figura 3 se observa un ejemplo de modelamiento relacional para el caso de un sistema de facturación. Consta de tres entidades: cliente, proyecto y factura, cada una con sus atributos. Además, se observan las relaciones de “uno a muchos” entre las entidades cliente - proyecto y entre proyecto - factura. Esto puede interpretarse como: “un cliente puede estar asociado a muchos proyectos” y “hay muchas facturas asociadas a cada proyecto” respectivamente.

Figura 3: Ejemplo de modelamiento entidad-relación.



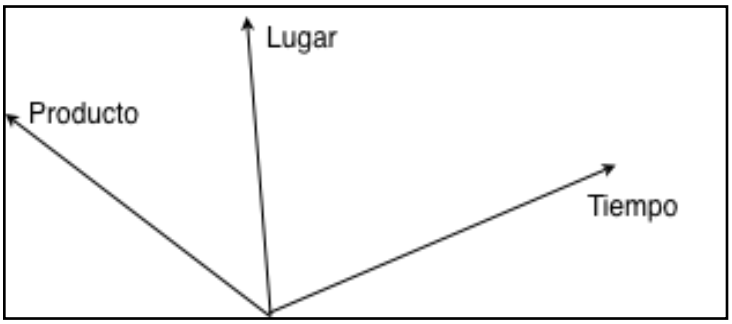
Fuente: Elaboración Propia

A pesar de que este enfoque satisface eficientemente las necesidades de manejo de datos de los sistemas operacionales [15], no resulta ser el más apropiado para la generación de información de niveles más agregados. Ésta última requiere del cruce de numerosas tablas y la agregación de muchos datos emanados desde distintos sistemas, lo que, debido a la baja redundancia y enfoque hacia el "día a día" del modelo relacional, genera en un lento desempeño, en donde una consulta puede tomar hasta días en ser contestada con la velocidad de procesamiento actual.

2.2.2 MODELAMIENTO MULTIDIMENSIONAL

Los usuarios finales de los sistemas de apoyo a la toma de decisiones (DSS[5]) piensan en múltiples dimensiones [41]. Por ejemplo, para el sistema de facturación de la Figura 3 se desean saber "cuantos productos del tipo A se vendieron en la ciudad B durante el segundo semestre del año 2006". En la consulta anterior las dimensiones que se distinguen son: tiempo, lugar y producto (Figura 4).

Figura 4: Dimensiones de una consulta multidimensional.



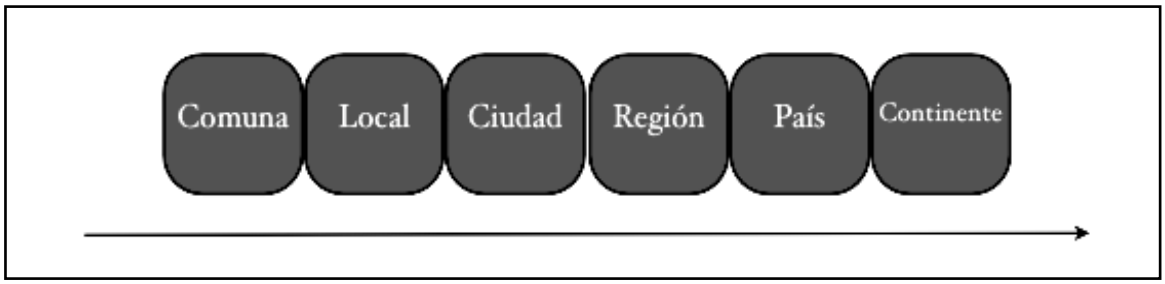
Fuente: Elaboración Propia

[5] *Decision Support System*.

En 1993, el mismo Codd propone el concepto de On Line Analytical Processing (OLAP), que apunta a responder las consultas de los usuarios finales en tiempos "razonables" [16]. Para lograr lo anterior, se hace necesario utilizar un enfoque multidimensional, mediante el procesamiento de los datos, de manera de lograr satisfacer los requerimientos de información aprovechando la dimensionalidad de las preguntas de los *end-users*. Lo anterior sirve como base para el modelamiento multidimensional de datos (MDM) [2]

Las dimensiones están generalmente asociadas a jerarquías. Éstas permiten responder preguntas para los distintos niveles de agregación de cada dimensión. En la Figura 5 se observa un ejemplo de distintas jerarquías que pueden estar asociadas a la dimensión "Lugar". De esta manera, siguiendo con el ejemplo anterior, se pueden hacer las siguientes consultas: "cuántos productos del tipo A se vendieron en cierta comuna/ciudad/región durante el segundo semestre del año 2006"

Figura 5: Distintas jerarquías de la dimensión "Lugar"



Fuente: Elaboración Propia

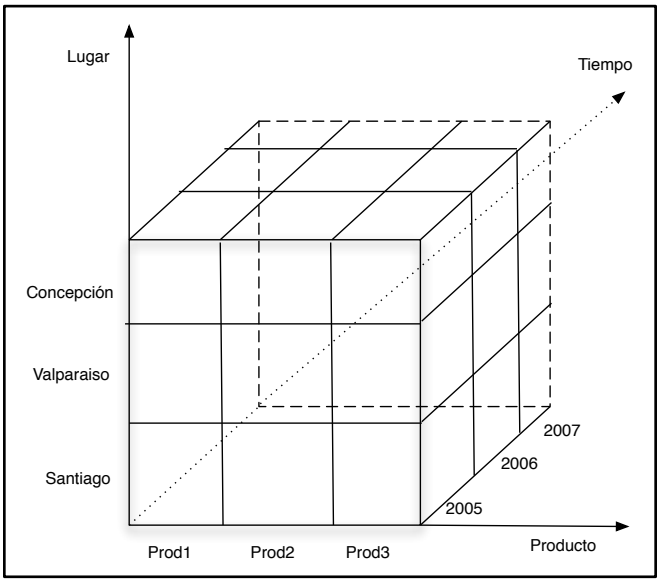
Para la implementación de un modelo multidimensional se pueden usar dos técnicas: cubo o estrella [43]. A su vez, se debe definir el grano que se utilizará, es decir, cuál es la mínima cantidad de información que se debe almacenar en el modelo multidimensional para poder responder adecuadamente a los requerimientos de los usuarios. Se relaciona con las entradas que posee el modelo físico de datos, pues el grano define el nivel de agregación o atomicidad de los registros que se almacenarán finalmente. Por ejemplo, si un usuario requiere información de ventas agregada diariamente y la granularidad del modelo es "ventas por semana", éste no responderá adecuadamente sus consultas. En ese caso se requiere un grano de "ventas diarias".

Modelo cubo

Consiste en representar el modelo como un "cubo de información" (como el de la Figura 6), sobre el cual se pueden hacer consultas sobre sus distintas dimensiones. Ésta requiere de un sistema administrador multidimensional de bases de datos (MDBMS).

Este modelo tiene la ventaja de ser muy rápido y eficiente para responder consultas de muchas dimensiones. Sin embargo, posee la fuerte desventaja de requerir de muchos recursos para sustentar una alta dimensionalidad. Lo anterior se suma al hecho de que se requiere para su implementación de una MDBMS[6], que poseen un alto costo y su uso no está muy masificado en las organizaciones.

Figura 6: Ejemplo de cubo de información para un sistema de ventas y facturación.



Fuente: Elaboración Propia

Sobre el cubo se pueden ejecutar las siguientes operaciones:

- Slicing: selecciona una dimensión del cubo (“tajada”).
- Pivoteo: rota el cubo y muestra una cara particular.
- Dicing: selecciona una o más dimensiones del cubo (saca un “pedazo” del cubo)
- Drill-down: muestra un nivel de jerarquía menor, es decir, el detalle de un punto de acumulación.
- Roll-up: muestra un nivel de jerarquía mayor, es decir, va hacia un nivel más agregado dentro de una dimensión.

A esta opción de hacer OLAP se le suele denominar MOLAP (OLAP multidimensional).

[6] Multidimensional Data Base Management System.

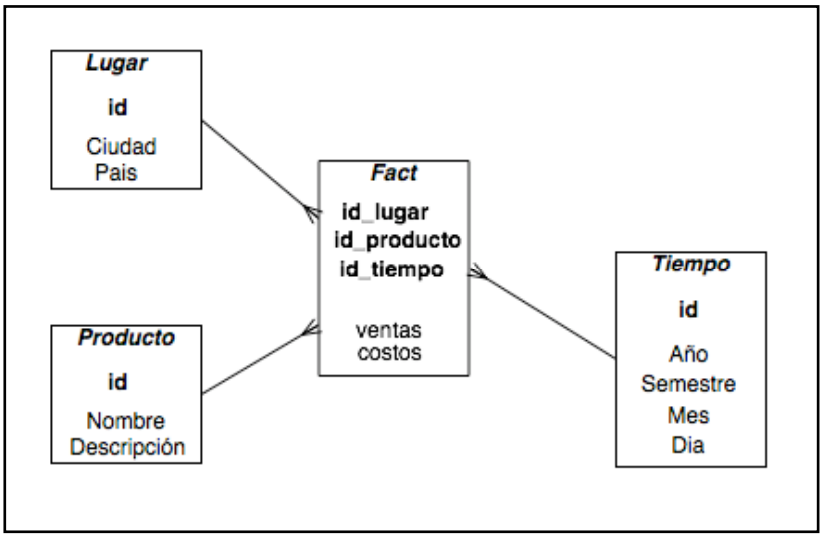
Modelo Estrella

Consiste en una representación multidimensional de datos utilizando la nomenclatura del modelo Entidad-Relación, pero sin considerar las restricciones de éste en cuanto a normalización y redundancia.

Está compuesto por una tabla central (*Fact*) y un conjunto de tablas dimensionales. Cada tupla que compone la tabla Fact posee como identificador el conjunto de llaves primarias de las dimensiones. A su vez, cada tupla posee atributos asociados a las distintas combinaciones posibles de llaves dimensionales. Si alguna de las dimensiones posee entidades asociadas, el modelo posee el nombre de *snowflake*.

En la Figura 7 se muestra un modelo estrella para un datamart de ventas, en donde se pueden identificar las dimensiones de tiempo, lugar y producto, como también la tabla Fact que entrega valores totales de costo y ventas a cada una de sus entradas.

Figura 7: Ejemplo de modelamiento estrella para data mart de ventas.



Fuente: Elaboración Propia

Para este modelo se requiere un sistema administrador de bases de datos relacional, como SQL Server, Oracle, MySql, etc. Éstos últimos son ampliamente usados por las organizaciones para el manejo de sus bases de datos operacionales, lo que representa una importante ventaja por sobre las MDBMS. Además, poseen gran flexibilidad de código y un lenguaje estandarizado (SQL).

A esta opción se le denomina ROLAP (OLAP relacional).

2.2.3 DATA WAREHOUSING

El término Data Warehousing fue acuñado por primera vez por Bill Inmon, quien, en su definición, estableció que es “una colección de datos orientados a temas, integrados, no-volátiles y variables en el tiempo, organizados para soportar necesidades empresariales” [26].

Por su parte, otra voz influyente, Ralph Kimball, define un Data Warehouse como “una colección de datos en forma de una base de datos, que guarda y ordena información que se extrae directamente de los sistemas operacionales y datos externos” [28]

A diferencia de los datamart, éstos integran información de muchas áreas de una organización, extrayendo datos de todos los sistemas operacionales. Sin embargo, poseen, en términos de su diseño y construcción, una secuencia común [12, 14, 28, 36]:

1. Análisis de las necesidades de un usuario final (*end-user*).
2. Selección de las fuentes de información.
3. Desarrollo del modelo lógico, donde las opciones más utilizadas son los modelos estrella y cubo.
4. Preparación de un prototipo para el usuario final, de manera de calibrar las necesidades de información con el desarrollo final.
5. Elegir un sistema administrador de bases de datos (SABD).
6. Construcción del modelo físico de datos, es decir, implementación del modelo lógico en el SABD.
7. Almacenar la información utilizando un proceso de extracción, transformación y carga de datos (ETL) y, posteriormente, evaluar el modelo.
8. Afinar el desempeño, haciendo modificaciones en la estructura interna de datos que mejoren los tiempos de repuesta.

El Proceso de Extracción, Transformación y Carga de Datos (ETL)

Los datos necesarios para generar información de apoyo a las decisiones provienen de múltiples fuentes, cada una con un formato particular; éstos, además, se deben procesar de manera de ser transformados en información útil para ser cargada en un repositorio. Todo este proceso, llamado ETL [21, 42] requiere de especial atención en términos de su diseño y planificación, de manera de asegurar la confiabilidad de la información resultante.

Extracción

La principal complejidad de esta etapa tiene relación con que cada uno de los sistemas operacionales almacena los datos en distintas arquitecturas, como archivos, bases de datos relacionales u otros [43].

Debido a lo anterior, se debe preparar minuciosamente la estrategia de extracción, considerando las posibilidades de incorporar nuevas fuentes, una calendarización adecuada, etc.

Transformación

Esta etapa resulta ser la más compleja de todo el proceso de ETL [43].

Entre sus complejidades están:

- Que las distintas fuentes recopiladas posean las mismas unidades de medida. Por ejemplo, en una fuente los costos pueden estar en dólares y en otra fuente en pesos chilenos.
- Generar los valores que serán almacenados en el repositorio. Esto considera la agregación de datos básicos para consolidar las distintas jerarquías.

Para hacer frente a las dificultades de esta etapa, además de salvaguardar la integridad de los datos de origen, resulta conveniente la creación de una Data Staging Area (DSA). Ésta es una etapa intermedia en donde se cargan los datos para ser transformados y, luego, seguir a la etapa de carga.

Carga

Dependiendo del formato final en que deben ser cargados los datos, ésta puede ser la etapa más sencilla [43]. Debido a que la DSA se encuentra generalmente en el mismo servidor que el datamart, la carga puede ser ejecutada usando las herramientas y lenguaje de la base de datos del servidor. En caso contrario, es decir, que la DSA se encuentre en un servidor distinto, se debe definir un protocolo de intercambio que explicite el mecanismo de transmisión de la información entre la DSA y el datamart.

2.3 ANÁLISIS DEL DESEMPEÑO DE UN SITIO WEB

A pesar de que la Web se ha transformado, entre otras cosas, en un canal de marketing de creciente importancia, muchas organizaciones han puesto el foco en el tráfico que posee un sitio web por sobre la medición del desempeño del sitio respecto de sus objetivos estratégicos, relacionados con aumentar la fidelización, agregar mayor valor a los productos y servicios, transformar a los visitantes en clientes, aumentar la venta cruzada, entre otros posibles objetivos. El hecho es que el número de visitantes de un sitio no necesariamente se correlaciona con el número de clientes reales ni con

la calidad de las relaciones con ellos [24], por lo que se debe considerar una mayor diversidad de dimensiones de análisis del desempeño de un sitio, de manera de poder tener más posibilidades para mejorarlo.

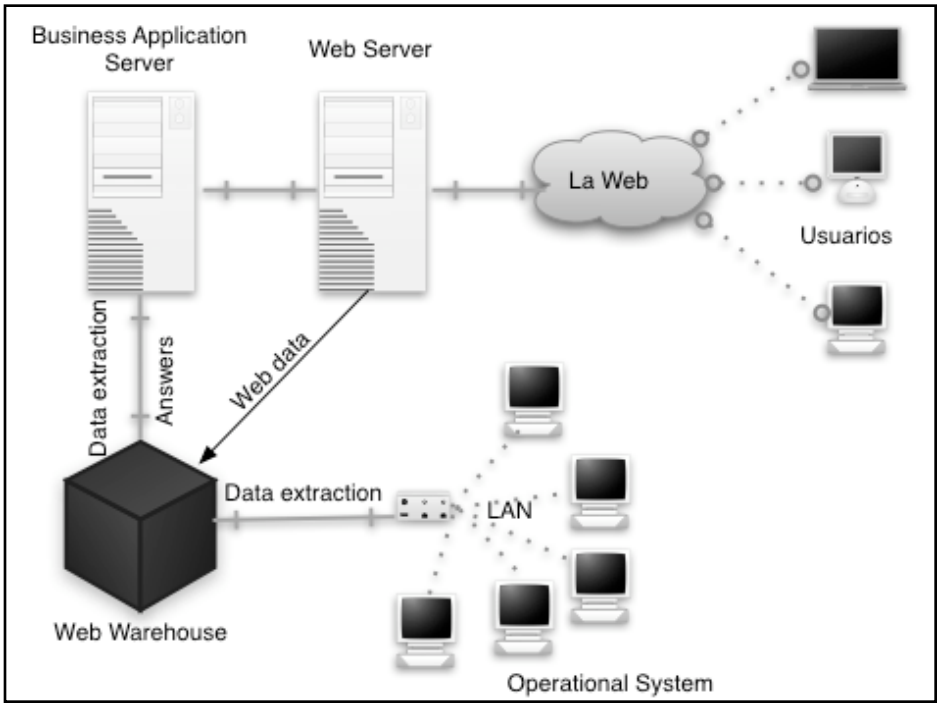
Para lograr lo anterior existen dos grandes enfoques de análisis de la utilización de un sitio web:

- **Análisis de indicadores de desempeño**: en este enfoque se utiliza estadística descriptiva relacionada con los distintos datos emanados de la web, como también análisis multidimensional de datos, es decir, OLAP sobre una arquitectura de Data Warehouse poblada con información construida a partir de web data.
- **Web mining**: utilizando técnicas y algoritmos de minería de datos aplicados a datos de la web. Se puede hacer minería del contenido (*web content mining*), comportamiento de los visitantes (*web usage mining*) y de la estructura de un sitio (*web structure mining*).

2.3.1 WEB WAREHOUSING

Corresponde a la aplicación de la arquitectura de data warehousing a web data [10], con el objeto de almacenar todo lo relacionado a la navegación sobre un sitio web. En la Figura 8 se puede observar una arquitectura genérica para su implementación, utilizando un servidor de aplicaciones para la operación del webhouse e integrando datos de los sistemas operacionales no relacionados con la web. Se

Figura 8: Arquitectura genérica de un web warehouse [43]



Fuente: Elaboración Propia

observa, además, que el web warehouse extrae los web data directamente del servidor web.

Sin embargo, los web data poseen algunas diferencias con los datos generados por transacciones de venta tradicionales relacionadas al Customer Relationship Management (CRM): en el caso del web Warehouse, el concepto es más amplio, e involucra a la relación con usuarios que no necesariamente son clientes. Al proceso de transformar a usuarios en clientes se le puede llamar User Relationship Management (URM), y corresponde a uno de los mayores objetivos de un web warehouse. Cabe destacar que, en la mayoría de los casos, no se posee información respecto del visitante: sólo se puede reconstruir su comportamiento, lo que, unido a datos emanados de otros sistemas operacionales de una organización, pueden construir una buena idea acerca de cuáles son las preferencias de un usuario.

2.3.2 WEB INFORMATION REPOSITORY (WIR)

Normalmente, cuando se construye un repositorio de información consolidada y limpia, se busca satisfacer requerimientos de información de un usuario final. Otra posibilidad es la alimentación de algoritmos de minería de datos.

Para ello, un repositorio de información web (WIR) se concentra en almacenar web data procesada, específicamente, información proveniente de web logs, contenido de texto y estructura de hipervínculos de un sitio.

Algunas ejemplos de preguntas que puede hacer un usuario final respecto del uso de un sitio web son [25]:

1. ¿Cuánto tiempo, en promedio, dura una sesión?
2. ¿Cuántas visitas tiene cada página en un periodo de tiempo?
3. ¿Cuál es la página más visitada en un periodo de tiempo?
4. ¿Cuánto tiempo se gasta un usuario en cada página que compone su sesión?

Para responder lo anterior, entre otras posibles preguntas, se debe definir tanto el grano del repositorio, como también las dimensiones que se considerarán. La opción que se debe implementar depende del enfoque de la aplicación final. Además, el proceso de ETL debe estar alineado con los objetivos del WIR.

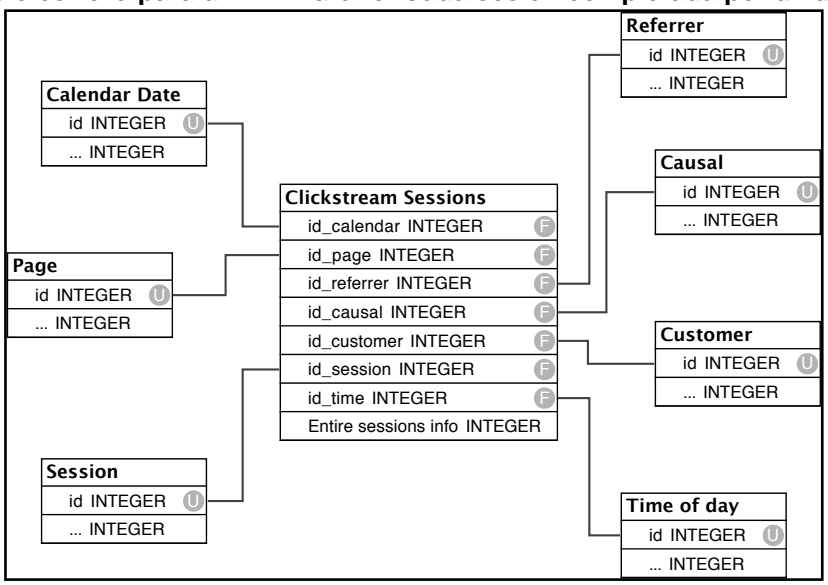
Modelos propuestos

En [27], Kimball y Merz proponen dos modelos genéricos con aplicaciones finales y granos distintos. En la Figura 9 se observa un modelo enfocado al análisis de

sesiones completas. En la Figura 10, los autores proponen un modelo con un grano asociado a cada página solicitada por los usuarios.

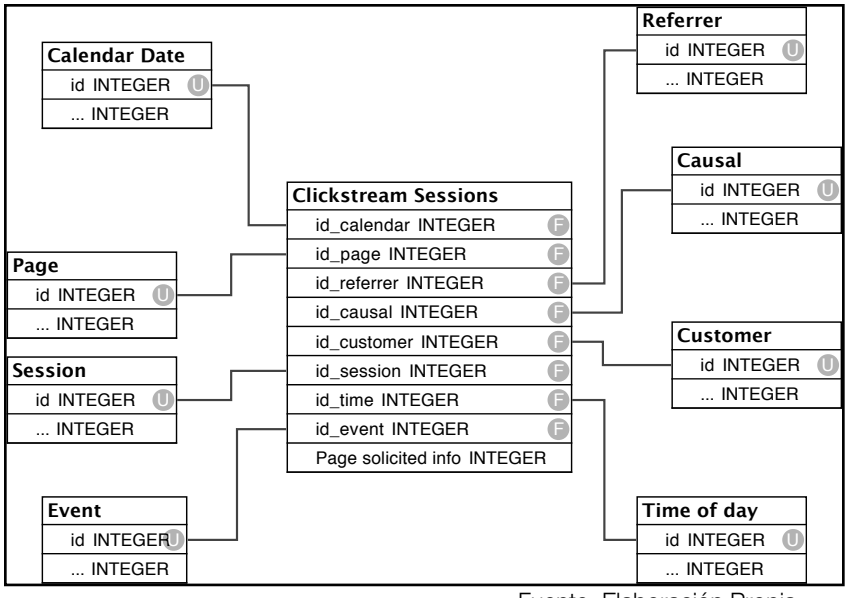
Por su parte, Velásquez y Palade [43] proponen un modelo genérico con un grano asociado a cada objeto web visitado. En la Figura 11 se observan sus dimensiones y atributos de cada tabla dimensional.

Figura 9: Modelo estrella para un WIR. Grano: Cada sesión completada por un usuario. [27]



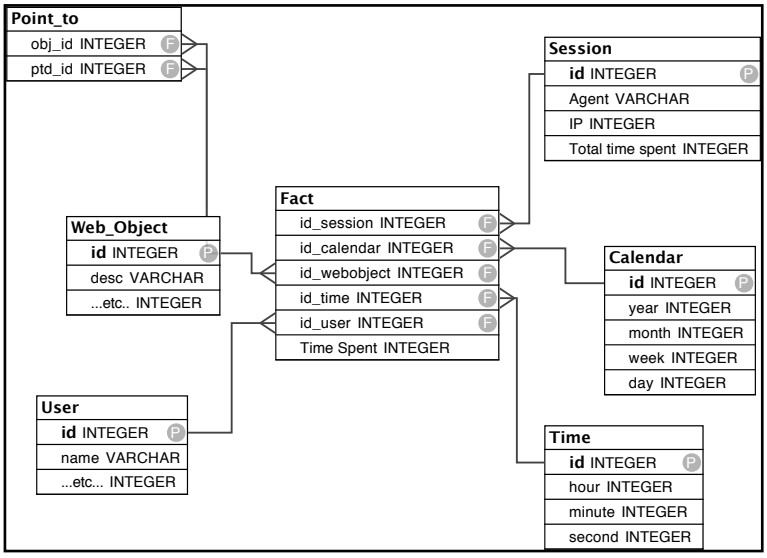
Fuente: Elaboración Propia

Figura 10: Modelo estrella para un WIR. Grano: Cada página solicitada por un usuario. [27]



Fuente: Elaboración Propia

Figura 11: Modelo estrella para un WIR. Grano: Cada objeto solicitado por un usuario [43]



Fuente: Elaboración Propia

Proceso de ETL para un WIR

El proceso de extracción, transformación y carga para el caso de un WIR consta, principalmente, de los siguientes procesos [27]:

1. **Filtrado de registros no necesarios**, de manera de reducir el tamaño de las transacciones al máximo posible, sin comprometer la integridad de los datos necesarios para llenar el datamart de acuerdo a su granularidad.
2. **Identificar sesiones** verificando además la consistencia lógica en términos del temporalidad de los eventos asociados.
3. **Identificar usuarios** si es posible, pareando una sesión encontrada con un usuario pre-existente. En caso de no ser posible, asociar la sesión a un usuario anónimo.
4. **Consolidar los datos en un formato único** que sea capaz de alimentar los programas de carga del repositorio.

A lo anterior se suma los distintos procesos relacionados con el procesamiento del contenido de las páginas de manera de poder ser analizado y cruzado con la información de uso.

Extracción y limpieza de datos

En esta primera etapa del proceso de ETL, se distinguen las siguientes actividades:

a) Eliminación de registros asociados a *Spyder Robots* y *Web Crawlers*.

b) Eliminación de peticiones de objetos, manteniendo los *pages views*. Cada registro de log es una petición de un cliente web al servidor. Esto incluye imágenes, documentos, etc. asociados a una página web que fue pedida por un usuario.
c) Eliminación de registros incoherentes o con datos erróneos.

Transformación de los datos

Es esta etapa los datos son procesados de manera de poder ser cargados correctamente en el repositorio. Las actividades asociadas a esta etapa son:

a) Sesionización.
b) Selección de palabras y *stemización*.
c) Construcción del *vector space model* asociado al contenido de texto de cada página.
d) Construcción del *user behaviour vector* asociado al comportamiento de los usuarios.

Carga de datos

Esta etapa debe mantener la coherencia entre los datos ya transformador y las tablas finales del modelo de datos en el que se almacenarán. Un programa se encarga de la carga periódica de los datos transformados en el repositorio.

2.4 La Web como canal de marketing y ventas

La principal razón por la que se requiere conocer las preferencias y necesidades de los visitantes de un sitio web es, precisamente, para mejorar su experiencia de navegación y transformarlo finalmente en cliente de la organización. Por ello, se presentan a continuación los conceptos de CRM, Comercio electrónico y ECRM, que permiten justificar la relevancia del problema planteado en esta investigación desde el punto de vista del marketing.

2.4.1 CUSTOMER RELATIONSHIP MANAGEMENT (CRM)

En un contexto competitivo, las organizaciones tienen múltiples variables sobre las cuales pueden tomar acción con el fin de mejorar sus resultados y obtener una mejor posición relativa a sus competidores. Considerando el hecho de que captar un cliente nuevo es más caro que mantener a uno pre-existente, sumado a que es más costoso venderle un producto adicional a un cliente antiguo que a uno nuevo [34], el manejo de la relación con los clientes de una empresa puede trasformarse en una buena opción de lograr mejores resultados.

El concepto de CRM tiene relación con la adopción de una estrategia orientada al cliente, de manera de construir relaciones duraderas y provechosas mediante la entrega de productos y servicios altamente personalizados, satisfaciendo de mejor manera las necesidades de cada clientes, e integrando acciones de marketing, ventas y atención al cliente [23].

De acuerdo a [9], una estrategia de CRM involucra cinco objetivos:

- **Captación de nuevos clientes**, transformando clientes potenciales en regulares. Generalmente, los clientes regulares son reticentes a optar por un competidor debido a su lealtad con la organización y, además, prestan mayor atención al buen trato que se les da que a los cambios de precios de los productos [38].
- **Venta cruzada**. Mediante la comprensión de las necesidades de los clientes regulares, las organizaciones se pueden anticipar a lo que los consumidores demanden. Para ello se requiere de un dialogo constante entre la empresa y el cliente [34].
- **Mantención de los clientes actuales**. En mercados saturados de oferta, la generación de transacciones recurrentes con el mismo consumidor contribuye a generar rentabilidad para la organización [38].
- **Segmentación de clientes**, de manera de comprender qué mensaje enviar a cada segmento de clientes y elegir el canal adecuado para ese mensaje [39].
- **Generación de rentabilidad a partir de los clientes actuales**, profundizando en la relación y comprensión de sus necesidades.

Un elemento crucial relacionado al CRM es la integración de múltiples canales comunicación con los clientes, debido a que crecientemente los consumidores optan por utilizar varios canales distintos [37]. De acuerdo a [32], los cinco tipos de canales que se utilizan para la comunicación con los departamentos de marketing, ventas y atención al cliente son (Figura 12):

- Vía Telefónica, mediante *call centers*.
- Marketing directo, cofmo campañas de correo y entrega de información directa.
- Fuerza de venta, es decir, personas que se comunican directamente con los consumidores.
- Locales de venta, en donde los clientes pueden obtener información respecto de los productos.
- Comercio electrónico, que ofrece atención en todo horario y que no involucran empleados en las transacciones.

Figura 12: Relación de los dptos. de Marketing, Ventas y Atención y canales de comunicación con los clientes.

Fuente: Elaboración Propia

2.4.2 COMERCIO ELECTRÓNICO

Con la aparición de la Web y su posterior penetración en la vida diaria de las personas, una de sus aplicaciones más interesantes para las organizaciones es la de ocuparla como canal de ventas, marketing y, en general, de comunicación con los clientes. Debido a lo anterior, el comercio electrónico ha crecido enormemente, canalizando durante el año 2005 ventas por U$4.300 millones en Latinoamérica, de las cuales un 4% corresponde a Chile [3].

Según [46] se puede definir comercio electrónico como "la automatización de transacciones comerciales usando computadores y tecnologías de información". Dependiendo de los agentes participantes, éste puede ser *Business to Business* (B2B), *Business to Consumer* (B2C), *Consumer to Business* (C2B) o bien, *Consumer to Consumer* (C2C) [22].

Algunas de las ventajas del comercio electrónico, en particular, del comercio B2C, son:

- **Menores costos de transacción que plataformas tradicionales**, siendo en algunos casos hasta diez veces menor [33].

- **Flexibilidad y transparencia**, dado que los consumidores pueden hacer órdenes en cualquier momento del día, y lo hacen con mayor información que por los canales tradicionales, pues pueden comparar información respecto de los productos [5].
- **Aumento de la competencia**, que posee un efecto positivo para los consumidores, generando menores precios.
- **Posibilidades de customizar los productos y servicios,** utilizando la enorme cantidad de información emanada de las transacciones electrónicas [18].
- **Nuevos servicios utilizando la plataforma de Internet** como, por ejemplo, la entrega de mayor información a los clientes respecto de los productos, fechas de entrega, stock, etc.

2.4.3 ELECTRONIC CUSTOMER RELATIONSHIP MANAGEMENT (ECRM)

Consiste en la aplicación del concepto de CRM al comercio electrónico. Debido a que la utilización de la Web facilita el logro de los objetivos de la adopción de una estrategia orientada al consumidor, existe una enorme oportunidad de generar relaciones duraderas y fructíferas con los clientes, disminuir los costos, mejorar las ventas y hacer más efectivos los esfuerzos de marketing [13] mediante la intensificación del uso de nuevas tecnologías, en particular, mediante el uso de la Web.

Uno de los principales desafíos en ese sentido, es la de construir y alimentar bases de conocimiento respecto de la historia de relación con el cliente, unificando información de todas las áreas de negocio de la organización para la agregación de valor mediante la generación de ofertas atractivas, beneficios especiales a los clientes más rentables, entre otras posibles opciones.

CAPÍTULO 3 – METODOLOGÍA

La metodología que se utilizó para el diseño y construcción del Web Information Repository (WIR) fue, fundamentalmente, la que se propone para el desarrollo de repositorios de información para Data Warehousing, adaptándola al problema particular de información relacionada a el uso, contenido y estructura de un sitio web.

3.1 DISEÑO DEL MODELO CONCEPTUAL DE ALMACENAMIENTO DE INFORMACIÓN

Los distintos modelos conceptuales de datos que componen el WIR fueron construidos como modelos de Entidad-Relación (ER) [15]. Sin embargo, cada uno de estos modelos posee un enfoque distinto de acuerdo a su función:

- **Modelo de datos del sitio web**: debe representar todos los datos que emanan de un sitio web, es decir, estructura, contenido y logs. Para este caso se diseñó un modelo ER tradicional que los almacena pre-procesando algunas de sus entradas. En una arquitectura de Data Warehouse representa la Data Staging Area.

- **Modelo de información del WIR**: debe representar la información construida a partir de los datos de la DSA. Se diseño mediante un modelo snowflake [29] a partir de lo propuesto por [43, 27]. Esta es una variación del modelo ER que nos permitirá utilizar algebra relacional de bases de datos y un lenguaje formal estándar, como SQL [43].

3.2 DISEÑO E IMPLEMENTACIÓN DEL PROCESO DE ETL

Extracción de datos

Al construir los modelos de datos se distinguieron las fuentes y tipos de datos de un WIR. Además, como ya se mencionó, se utilizó una DSA en donde se reúnen las distintas fuentes de datos y se genera una primera limpieza de los mismos. Lo anterior permite ejecutar un proceso de extracción de datos seguro, pues no se manipulan los datos directamente en sus fuentes.

Limpieza de los datos

De acuerdo a lo mencionado en [26,27], los principales problemas encontrados en las fuentes de datos tienen que ver con la inconsistencia e irrelevancia de algunas de sus componentes. En el contexto de este trabajo, se ejecutó una limpieza de acuerdo

a lo propuesto en [43, 27]. de manera de hacer frente a dichos problemas. Para el caso de las fuentes de datos en este caso particular, se eliminaron:

- Sesiones no correspondientes a usuarios humanos
- Sesiones inválidas (muy cortas, muy largas o incongruentes).
- Palabras del contenido en forma de texto que no tienen significado propio, o bien, que cumplen función de ilativo.
- Filtrar logs, eliminando llamadas a gráficos, animaciones y otros contenidos no textuales, dejando así sólo peticiones de páginas web.

Transformación de los datos

Para el caso de datos emanados de un sitio web y la navegación de sus usuarios, se distinguen dos grandes procesos de transformación que se implementan de maneras distintas:

- Reconstrucción de sesiones de navegación (sesionización). Se utilizó una estrategia reactiva [40] mediante una heurística orientada al tiempo [6, 7, 17].
- Representación del contenido de las páginas. Se utilizó una representación vectorial del contenido de texto llamada *Vector Space Model* [1, 4, 35], que permite visualizar el contenido de un documento como un set de palabras que poseen pesos relativos distintos.

3.3 APLICACIÓN A UN SITIO WEB REAL

En base al modelo conceptual desarrollado en las etapas anteriores, se construyeron las bases de datos necesarias para la operación del repositorio. Además, se adaptaron a los requerimientos de un sitio web real asociado a la industria bancaria, manteniendo sólo la información que se pudo construir a partir de los datos disponibles. El sitio escogido posee un alto nivel de tráfico de usuarios y gran cantidad de páginas, por lo que resultó adecuado para la ejecución de pruebas tanto de los modelos de datos como del proceso de ETL.

3.4 CONSTRUCCIÓN DEL PROTOTIPO DE LA INTERFAZ DE CONSULTA

Se revisaron las herramientas existentes en el mercado para el análisis del uso de sitios web, como Google Analitycs, WebTrends, etc. que poseen estrategias proactivas, es decir, utilizan *cookies*. A partir de dicha revisión y el levantamiento de requerimientos, se sintetizaron las características necesarias que debería tener un diseño final y se construyó el prototipo.

CAPÍTULO 4 – MODELO TEÓRICO

Sobre la base de los modelos propuestos por la literatura, en el siguiente capítulo se propone un modelo conceptual para la construcción de un webhouse. El modelo propuesto consta de cuatro elementos: especificación de los requerimientos, modelos de almacenamiento, proceso de ETL y presentación de la información.

4.1 REQUERIMIENTOS GENERALES

Para la construcción de un sistema se debe, en una primera etapa, efectuar una especificación de requerimientos. Para el caso del modelo de webhouse se muestran a continuación: los requerimientos de información, los perfiles de usuario, los casos de uso y, por último, los procesos de negocio involucrados.

4.1.1 REQUERIMIENTOS DE INFORMACIÓN

Un repositorio de información tiene como principal finalidad el responder consultas de usuarios humanos a partir de un conjunto de datos.

En la sección 2.3.2 se presentaron cuatro ejemplos de preguntas que un repositorio de información web puede responder. En general, se pueden distinguir interrogantes de tres tipos: relacionadas con las sesiones, relacionadas con la conversión por objetivos y relacionadas al contenido del sitio web.

Consultas relacionadas a las sesiones

Las sesiones están asociadas directamente al comportamiento de los visitantes de un sitio. En consecuencia, resulta de gran interés responder interrogantes respecto a las sesiones y sus características.

Algunos ejemplos de consultas relacionadas a las sesiones pueden ser:

1. ¿Cuántas páginas, en promedio, visitan las sesiones en un periodo de tiempo?
2. ¿Cuáles son los sets de páginas más frecuentes que componen una sesión en un periodo de tiempo?
3. ¿Cuáles son las páginas menos visitadas en un periodo de tiempo?
4. ¿Cuáles páginas concentran la mayor cantidad del tiempo en promedio de las sesiones?
5. ¿Cuántos bytes en total se transmitieron en las sesiones de un determinado tiempo?
6. ¿A qué hora se produce la mayor cantidad de sesiones?
7. ¿Qué día de la semana genera mayor cantidad de sesiones?

Consultas relacionadas a conversión por objetivos

La conversión por objetivos es uno de los indicadores más relevantes para medir el desempeño de un sitio, pues indica cuántos usuarios llegaron a una página que para la organización resulta de más importancia para sus resultados. Ejemplos de objetivos son las páginas de cierre exitoso de una transacción, registro exitoso de un nuevo usuario, entrega de datos de contacto (e.g. mail, dirección o teléfono) para informar acerca de nuevas promociones, etc.

Algunos ejemplos de consultas relacionadas a conversión por objetivos son:

1. ¿Cuál fue el porcentaje de conversión en un periodo determinado?
2. ¿Cuál es el objetivo que concentró mayor conversión por periodo?
3. ¿En qué momentos (horarios y/o fechas) se produce el mayor porcentaje de conversión?
4. ¿Cuántas páginas en promedio toma una sesión en alcanzar un objetivo?

Consultas relacionadas al contenido de un sitio web

El contenido de un sitio, por su parte, es la principal razón que motiva a un usuario a visitarlo.

Como se mencionó en la sección 2.1.3, el contenido de texto de un sitio web se representa utilizando el *vector space model*. De esta manera, se calcula el peso de una palabra en la totalidad del contenido del sitio.

Algunos ejemplos de consultas relacionadas al contenido de un sitio son:

1. ¿Cuáles son las palabras con mayor/menos peso relativo en el sitio?
2. ¿Cuáles son las palabras que componen las páginas que concentran la mayor cantidad de visitas?
3. ¿Cuáles son las palabras con las que llegan los usuarios a un sitio web, es decir, cuáles son las búsquedas que atraen a un visitante?
4. ¿Cuál es el peso de las palabras especiales (i.e. destacadas) en el sitio?

4.1.2 PERFILES DE USUARIO

Se distinguen, al menos, dos perfiles distintos para el uso de la información entregada por el repositorio: webmaster y marketing.

Cada uno de ellos deberá tener distintos privilegios y tipos de reportes de acuerdo a sus necesidades particulares de información.

Perfil administrador / webmaster

Quien toma decisiones respecto del contenido y estructura de un sitio, llamado comúnmente *webmaster*, es el principal usuario de una plataforma de información como es el WIR. A dicho perfil corresponden todos los requerimientos de información respecto de tráfico, navegación, contenido, e incluso, conversión por objetivos.

Para este perfil, se debe entregar privilegios para el acceso a toda la información contenida en el repositorio. Para ellos, se propone que este perfil pueda acceder a un cubo OLAP cuyas dimensiones son todas las presentes en el modelo de datos. De esta manera, podrá hacer todo tipo de consultas respecto a número de sesiones, bytes transmitidos y tiempo de permanencia por página de las sesiones. Sumado al análisis OLAP, este usuario debe tener acceso a los pesos de las palabras en el contenido del sitio, generado a partir del *vector space model*, como también información respecto de los sets de páginas más recurrentes en las sesiones y el porcentaje de conversión por cada objetivo.

Perfil usuario de marketing

El usuario de marketing tiene como principal finalidad analizar la efectividad de las campañas y promociones que se implementan, además de medir el cumplimiento de los objetivos que se tienen para este canal relacionados con su área.

En ese sentido, el perfil de usuario de marketing deberá tener acceso a la información respecto de las campañas y las visitas generadas a partir de ellas, como también, al número de sesiones exitosas, es decir, que cumplen con un escenario predefinido como exitoso, generadas a partir de cada campaña.

Otro punto relevante para este perfil es el de los usuarios de un sitio. Si se cuenta con dicha información, puede ser de gran interés para este perfil conocer cuáles son los usuarios más antiguos, los que más utilizan el sitio y los mejores en términos de la conversión por objetivos.

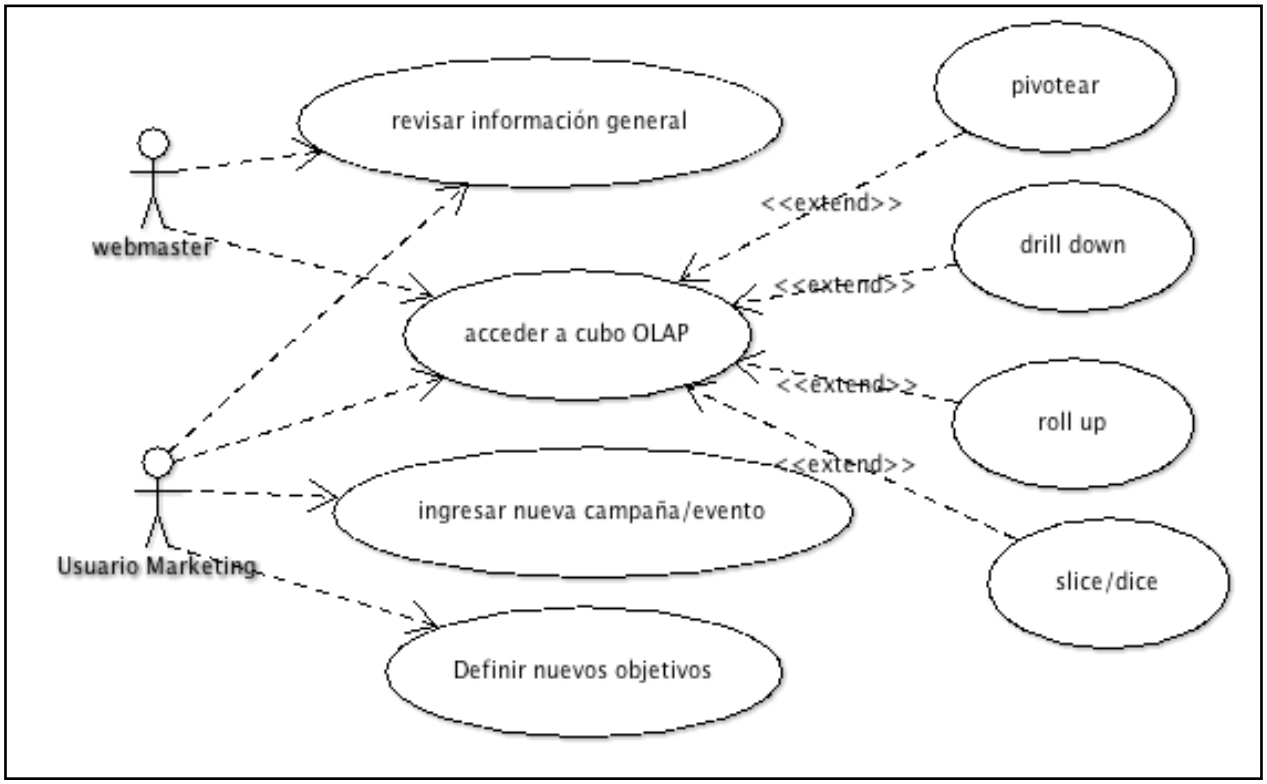
Para el análisis OLAP, deberá contar con un cubo que entregue los indicadores mencionados anteriormente cruzados por las dimensiones temporales.

Por último, es necesario que este usuario tenga la posibilidad de los objetivos del sitio que generan conversión, de manera de poder medir temporalmente el desempeño del sitio para cada uno y tomar decisiones sobre la base de esa información.

4.1.3 DIAGRAMA DE CASOS DE USO

De acuerdo a lo mencionado en este capítulo, el diagrama general de casos de uso para los requerimientos del sistema es el presentado en la Figura 13. En él se ven reflejadas las actividades que cada perfil de usuario puede realizar en su interacción con el repositorio de información. Se incluyen además las actividades relacionadas con el análisis de los cubos OLAP, mencionadas en la sección 2.2.2.

Figura 13: Diagrama de casos de uso.



Fuente: Elaboración Propia

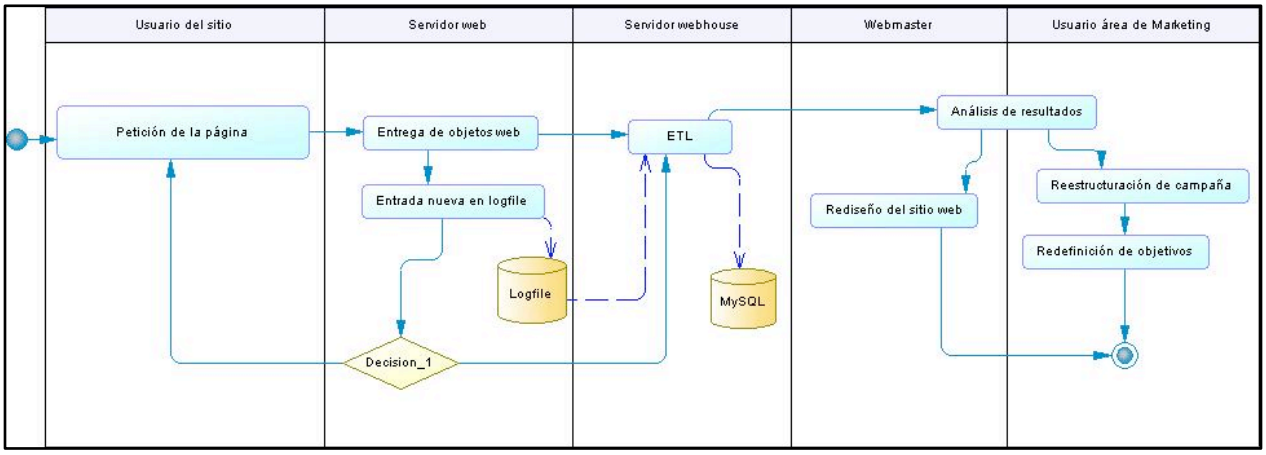
4.1.4 PROCESOS DE NEGOCIO

Los procesos de negocio que componen el análisis del desempeño de un sitio web mediante un webhouse son, principalmente, tres: 1) petición y entrega de una página web, 2) proceso de ETL y 3) Análisis de la información y toma de decisiones.

En la Figura 14 se observan los distintos procesos involucrados y los roles que los ejecutan. Los actores involucrados son los visitantes del sitio web además de los dos perfiles de usuario de la información del repositorio, es decir, un web master y un usuario del área de marketing.

El proceso comienza con la petición de una página web por parte de un visitante. Un servidor web le entrega la página junto con los objetos que la componen y, al mismo tiempo, registra en un archivo de log todas las peticiones hechas al servidor y sus resultados. Posteriormente, el servidor en donde opera el webhouse se encarga de extraer los datos, transformarlos en información y cargarlos en la base de datos. Luego, los distintos perfiles de usuario pueden acceder al análisis de dicha información, tanto a los indicadores de desempeño predefinidos como a los cubos OLAP que se hayan construido. Por último, cada usuario toma decisiones asociadas a su perfil de acuerdo a la información que recabó.

Figura 14: Diagrama de actividades involucradas en la operación de un webhouse.



Fuente: Elaboración Propia

4.2 MODELOS DE ALMACENAMIENTO

Como se describió en la sección 2.3.2, los datos emanados de un sitio web con los que se construye la información que se almacena en el repositorio poseen tres fuentes principales: web logs, contenido de las páginas y estructura de hipervínculos.

Dada la multiplicidad de fuentes, se debe recopilar los datos de cada una y aglutinarlos para su posterior limpieza y procesamiento. Para efectuar este proceso se utiliza la *Data Staging Area*, que sirve como etapa intermedia en el proceso de ETL.

A su vez, se debe modelar el repositorio donde se almacenará la información limpia y procesada, de manera que pueda responder a los requerimientos de información.

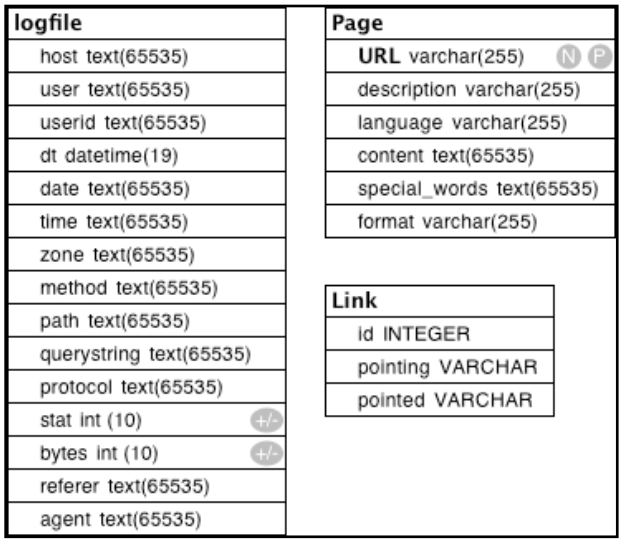
4.2.1 MODELAMIENTO DE LA DATA STAGING AREA

Para el caso de un WIR, la data staging area debe cumplir con los siguientes requerimientos:

- Almacenar los datos útiles de los logs, esto es, sólo los correspondientes a usuarios humanos y sesiones válidas.
- Almacenar las páginas con su contenido ya pre-procesado, es decir, luego de la *stemización*.
- Almacenar la estructura de hipervínculos incluyendo enlaces externos.

Considerando lo anterior, se propone el diseño conceptual de una DSA genérica que posee tres grandes entidades de datos: página web, link y web log file (Figura 15).

Figura 15: Tablas de la DSA.



Fuente: Elaboración Propia

4.2.2 MODELAMIENTO ESTRELLA DEL REPOSITORIO

Para el almacenamiento de la información procesada y limpia proveniente de la DSA, se diseñó un modelo *snowflake* con siete dimensiones.

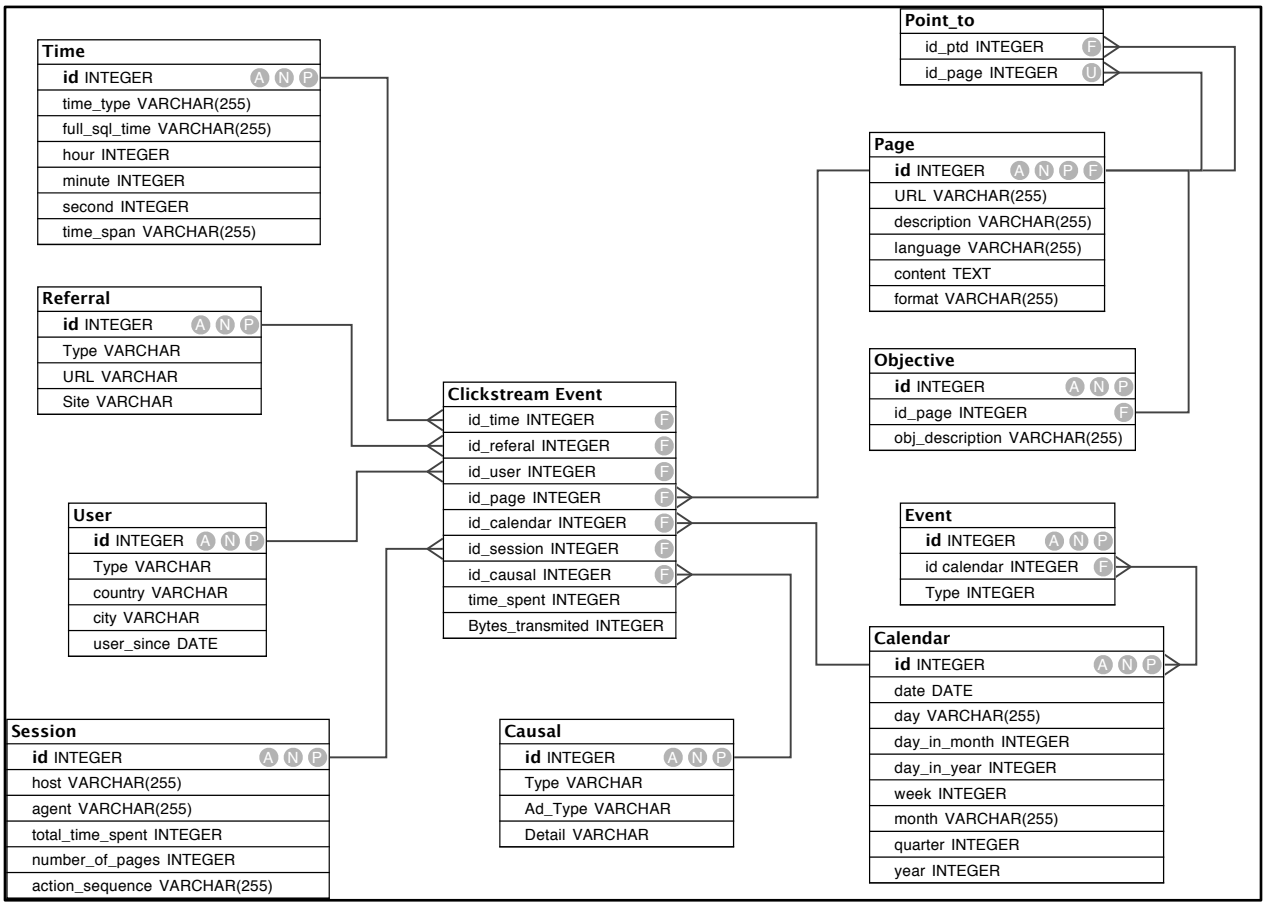
En la Figura 16 se observa el modelo completo, compuesto por la tabla fact y las siete dimensiones. Además, se observa las tablas adicionales *point_to, event y objective*, asociadas a las dimensiones, que hacen que el modelo sea del tipo *snowflake*.

Dimensión Tiempo

Esta dimensión representa la hora del día en que ocurrió el evento almacenado en la tabla Fact. Además de la posibilidad de requerir una hora o un rango en particular, se pueden definir rangos de tiempo más funcionales para el análisis, como: hora de almuerzo, hora de apertura y cierre, periodos de punta, etc.

Por otra parte, esta dimensión puede ser construida a distintos niveles dependiendo de las necesidades de información (e.g. minutos, segundos, centésimas de segundo). Sin embargo, debido al gran número de registros que puede significar almacenar eventos a niveles de centésimas o milésimas de segundo, es recomendable utilizar para la construcción de la información hasta el nivel de los segundos [27].

Figura 16: Modelo snowflake genérico para un WIR.



Fuente: Elaboración Propia

Al considerar una granularidad de un segundo, esta dimensión constaría de 86400 filas, lo que permite generar varios tipos de análisis que pueden resultar muy provechosos.

De esta manera, los principales atributos con los que cuenta esta tabla son:

Tabla 1: Atributos de la dimensión "Tiempo".

Nombre	Tipo/descripción	Rango/ejemplo
Id	número entero auto-incremental	desde 0 hasta 85399
Full SQL time	tiempo en formato HH:MM:SS	desde 00:00:00 hasta 23:59:59
Hour	número entero referente a la hora	desde 0 hasta 23
Minute	número entero referente al minuto	desde 0 hasta 59

Second	número entero referente al segundo	desde 0 hasta 59
Time span	cadena de caracteres respecto del rango predefinidos de tiempo	e.g. hora de almuerzo, hora de punta, hora de cierre y apertura

Dimensión Página

La dimensión "Página" almacena cada una de las distintas páginas que componen un sitio web. Es considerada de "cambio lento" debido a que usualmente esta tabla permanece inalterada hasta que se agrega una nueva página al sitio o alguna de las pre-existentes se modifica.

Cabe destacar que en esta tabla se almacena el contenido pre-procesado, esto es, *tokenizado* y *stemmizado*. De esta manera se puede cruzar un conjunto de eventos con el contenido sobre el cual dichas sesiones navegaron, pudiendo medir la relevancia e interés que genera cierto tipo de contenidos en relación con el resto.

Los principales atributos que componen la tabla "Página" son los siguientes:

Tabla 2: Atributos de la dimensión "Página".

Nombre	Tipo/descripción	Rango/ejemplo
Id	número entero auto-incremental	desde 0 hasta N
URL	cadena de caracteres con la URL de la página.	e.g. "www.ejemplo.com/index.html"
Description	cadena de caracteres con la descripción de la página.	e.g. "página de inicio"
Language	cadena de caracteres con el idioma de la página.	e.g. "Español", "Inglés", etc.
Content	campo de texto compuesto por el contenido procesado de la página.	e.g. "Contenido página prueba contacto horario"
Format	cadena de caracteres que describe el formato de la página.	e.g. "HTML", "PHP", etc.

A esta dimensión se le asocian las tablas llamadas "Objetive" y "Point to". En ellas se registran qué paginas definen que una sesión es exitosa (e.g. la página de pago de un producto) y la estructura de hipervínculos del sitio respectivamente. La definición de objetivos es fundamental para calcular la conversión por objetivos, principal métrica referente al desempeño de un sitio web.

Dimensión Calendario

Esta dimensión está asociada a la fecha en que ocurre un evento almacenado en la tabla fact. Se almacena aquí una entrada por cada día de cada año en el que se

registran eventos. Asimismo, se definen las jerarquías que posteriormente ocupara el cubo OLAP para mostrar los resultados.

Las principales dimensiones de esta dimensión son:

Tabla 3: Atributos de la dimensión "Calendario".

Nombre	Tipo/descripción	Rango/ejemplo
Id	número entero auto-incremental	desde 0 hasta (nº de años*365)
Date	fecha en formato DD/MM/AA.	desde 01/01/año_inicial hasta 31/12/año_final
Day	cadena de caracteres referente al dia de la semana.	“lunes”, “martes”,…, “Domingo”
Day in month	número entero referente al día del mes.	desde 1 hasta 31
Day in year	número entero referente al día en el año.	desde 1 hasta 365
Week	número entero referente a la semana en el año.	desde 1 hasta 52
Month	cadena de caracteres respecto del mes.	“enero”,“febrero”,…, “diciembre”
Quarter	número entero referente al trimestre.	desde 1 hasta 4
Year	número entero referente al año.	e.g. “2008”, “2009”, etc.

Se le asocia a esta dimensión una tabla llamada “Evento”, donde uno puede asociar fechas de calendario a eventos particulares (e.g. “18 de septiembre” asociado al evento “fiestas patrias”). De esta manera, se pueden hacer análisis de acuerdo a eventos relacionados a la industria para la que se construye el repositorio.

Dimensión Sesión

Como se mencionó en 2.1.3, la actividad de un usuario se enmarca dentro de una sesión. Ella almacena, entre otras cosas, información respecto a las páginas visitadas y tiempo que el usuario permaneció en ellas.

La importancia de esta dimensión radica en que lo almacenado en ella es pieza clave del análisis de desempeño de un sitio, pues a partir de las sesiones se pueden definir tipos de usuarios, segmentar, clasificar las sesiones como exitosas o no-exitosas de acuerdo a la conversión por objetivos, entre otras posibilidades.

Los atributos que componen esta dimensión son:

Tabla 4: Atributos de la dimensión "Sesión".

Nombre	Tipo/descripción	Rango/ejemplo
Id	número entero auto-incremental	desde 0 hasta Nº de sesiones encontradas
Host	cadena de caracteres con la dirección IP del visitante.	e.g. “201.168.8.1”

Agent	cadena de caracteres con la descripción del visitante (browser, sistema operativo, etc)	e.g. “Mozilla/5.0 (Macintosh; U; Intel Mac OS X; es-ES; rv:1.8.1.12)”
Total_time_spent	número entero con el tiempo que duró la sesión.	e.g. “20 [seg]”
Number_of pages	número entero con el número de páginas que compone una sesión.	desde 1 hasta Nº de páginas de la sesión que más páginas visitó
Action_Sequence	campo de texto que contiene las páginas que componen la sesión.	e.g. “index.htm – contacto.htm”

Dimensión Causal

En esta dimensión se registran las distintas campañas y/o promociones que generan que un usuario visite el sitio. Un ejemplo de lo anterior puede ser la promoción de un producto a través de un banner pagado en otro sitio. Al registrar esta causal y asociarla a una sesión generada por un clic en ese banner, se puede tener una métrica del éxito de una campaña publicitaria.

Dimensión Usuario

Algunos sitios poseen sistemas de registro de usuarios, lo que permite identificar a los usuarios en el momento que estos ingresan. De esta manera, se puede almacenar información del historial de cada usuario, como también, hacer un seguimiento a sus compras o peticiones de servicios.

Debido a lo anterior, se incorpora una dimensión de usuario al modelo de datos, de manera de poder ejecutar consultas respecto a cada tipo de usuario, o bien, a algún usuario en particular.

Dimensión *Referral*

En los archivos de registro presentes en el servidor web, existe un campo llamado *referrer* relacionado con la procedencia del usuario que pide una página, es decir, registra el lugar donde se presionó el hipervínculo que generó la visita al sitio.

Este dato posee gran importancia en el análisis, entre otras cosas, debido a que la visita puede proceder de un buscador, en cuyo URL del *referral* se almacena la consulta con la cual se generó el hipervínculo a la página. Por ejemplo, si el *referral* de una petición al es “http://www.url_buscador.com/search?hl=es&q=club+deportivo”, se distingue que se procede de un buscador, que la consulta fue en idioma español y se utilizó la frase “club deportivo”. De esta manera, se puede identificar qué buscan quienes visitan el sitio.

Tabla Fact: clickstream event

La tabla *fact* es la entidad donde se registran los eventos que se analizarán de acuerdo a las distintas dimensiones. La granularidad en este caso es “cada petición de página de un usuario”, es decir, cada vez que un usuario acceda a una página, se registrará en esta tabla asociado a todas las dimensiones mencionadas anteriormente.

Las medidas de cada registro que la compone son los indicados en la Tabla 5.

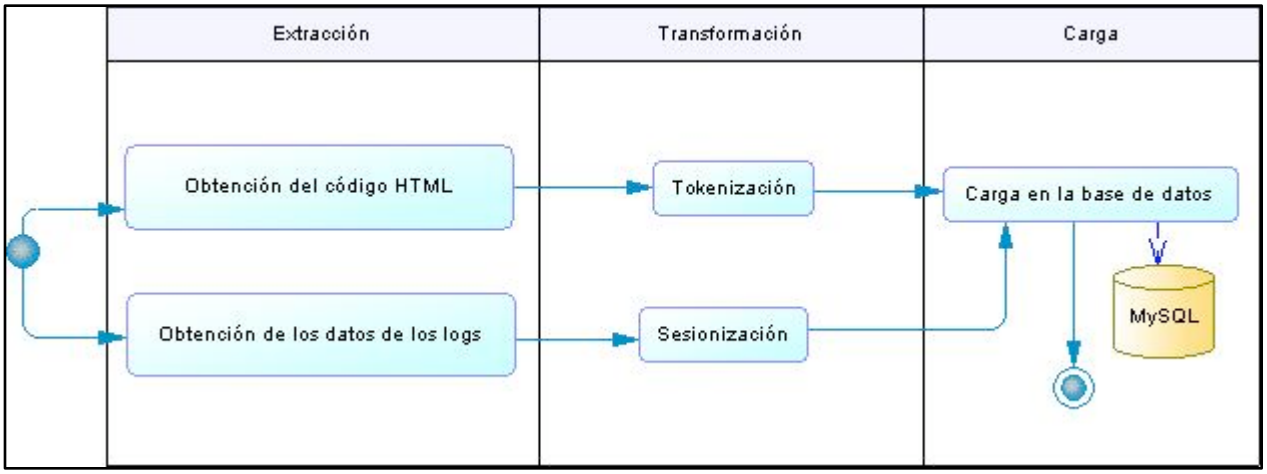
Tabla 5: Medidas de la tabla "Clickstream event".

Nombre	Tipo/descripción	Rango/ejemplo
Bytes_transmited	número entero con la cantidad de bytes transmitidos desde el servidor al cliente.	e.g. “234 [bytes]”
Time_spent	número entero con la cantidad de tiempo en segundos que el visitante permaneció en una página.	e.g. “25 [segundos]”

4.3 DISEÑO DEL PROCESO DE ETL

Estando resuelto el modelo de datos en donde se almacenará la información del repositorio, se debe diseñar un correcto proceso de extracción de los datos desde sus fuentes, su transformación en información orientada a responder las consultas específicas de negocio y su posterior carga en la base de datos (Figura 17). El proceso de ETL para poblar un WIR genérico se construyó en base a lo planteado en la sección 2.3.2.

Figura 17: Flujo del proceso de ETL.



Fuente: Elaboración Propia

4.3.1 EXTRACCIÓN

Fuentes de datos

Las fuentes de datos de un repositorio de información web son principalmente tres: web logs, contenido de texto y estructura de hipervínculos de un sitio. De acuerdo al formato y estructura de cada una de las fuentes, se debe diseñar una estrategia particular para obtener los datos necesarios.

1) Web logs

Los archivos de logs están, generalmente, en archivos de texto como a se describió en la sección 2.1.2. De estos archivos se debe extraer cada registro desde cadenas de caracteres separadas por un salto de línea. Luego, se debe extraer cada uno de los miembros que componen cada línea, para almacenarlos en una tabla construida con los atributos necesarios.

2) Contenido de texto

Para obtener el contenido de una página web, éste debe ser extraído directamente desde el código HTML. Debido a que este lenguaje se estructura en base a etiquetas que no son parte del contenido, éstas deberán ser limpiadas. Además, debe ser eliminada toda la información referente al estilos y scripts en el código. De esta manera, se obtiene un set de palabras que componen el contenido de un texto.

3) Estructura de hipervínculos

Los datos acerca de la estructura de hipervínculos de un sitio se encuentra, al igual que el contenido, dentro del código HTML de las páginas que lo componen.

Figura 18: Código y presentación de un hipervínculo HTML.

Código HTML

```
…
<a href="http://www.example.com/">Link a página</a>
…
```

Presentación

Link a página web

Fuente: Elaboración Propia

En la Figura 18 se observa la representación de un hipervínculo en el código HTML y la forma en que un usuario generalmente lo observa en una página web. Cabe destacar que el hipervínculo del ejemplo lleva hacia la página “http://www.example.com”.

Descripción del proceso

a) Extracción de estructura de links y contenido de texto

La primera etapa para la extracción de los datos es la obtención de la páginas que componen el sitio. Para ello se debe determinar cuál será la página de partida para recorrer el sitio. Es recomendable establecer la página de incio de un sitio, comúnmente llamada *home*, como comienzo de la extracción, pues desde ella se puede acceder a todas las páginas que componen el sitio.

Posteriormente, se obtiene el código HTML de dicha página. Desde él se obtendrán la estructura de hipervínculos del sitio y el contenido de texto de sus páginas. Posteriormente se crea un arreglo de hipervínculos en donde se almacenarán los hipervínculos detectados en el código de las páginas. En base a este arreglo se ejecutará la extracción del contenido de texto de las páginas.

Luego de haber recorrido la totalidad de las páginas extrayendo los hipervínculos que las componen, se comienza a extraer el contenido de texto de cada página presente en el arreglo. Esto se hace, primero, eliminando todas las palabras que estén entre etiquetas de scripts (<scripts> </scripts>) y de estilos (<style> </style>). Estas palabras no constituyen contenido de la página. Los scripts son programas incluidos en el código que permiten la automatización de tareas para la creación de utilidades. Por su parte, las palabras que estén etiquetadas como *style* tienen por finalidad establecer el formato visible de las palabras y objetos que componen una página.

El paso siguiente corresponde a la eliminación del resto de las etiquetas, de manera de obtener sólo el contenido de texto de cada página.

Por último, se deben eliminar caracteres no textuales como: paréntesis, puntos, comas, comillas, etc.

b) Extracción de los datos del archivo de log

Para la extracción de los datos respecto de la navegación en un sitio web, se debe tener acceso al archivo de registro de actividad del servidor web con sus clientes, llamado *weblog file*. Como se planteó en la sección 2.1.2, estos datos se encuentran generalmente en un archivo de texto en donde cada registro está separado del siguiente con un salto de línea. Cada uno de estos registros debe ser separado de

manera de obtener cada uno de los datos con los que se completará la tabla de la base de datos relacional.

Cabe destacar que en esta etapa de extracción se debe filtrar todo registro que no aporte información relevante para el análisis, como son las visitas de *crawlers*

4.3.2 TRANSFORMACIÓN

Una vez realizada la extracción de los datos, éstos deben ser transformados en la información que servirá de base para la toma de decisiones respecto del contenido y estructura del sitio web. Para el caso de los web data, la transformación se conforma principalmente de dos grandes procesos: sesionización y *tokenización*.

Sesionización

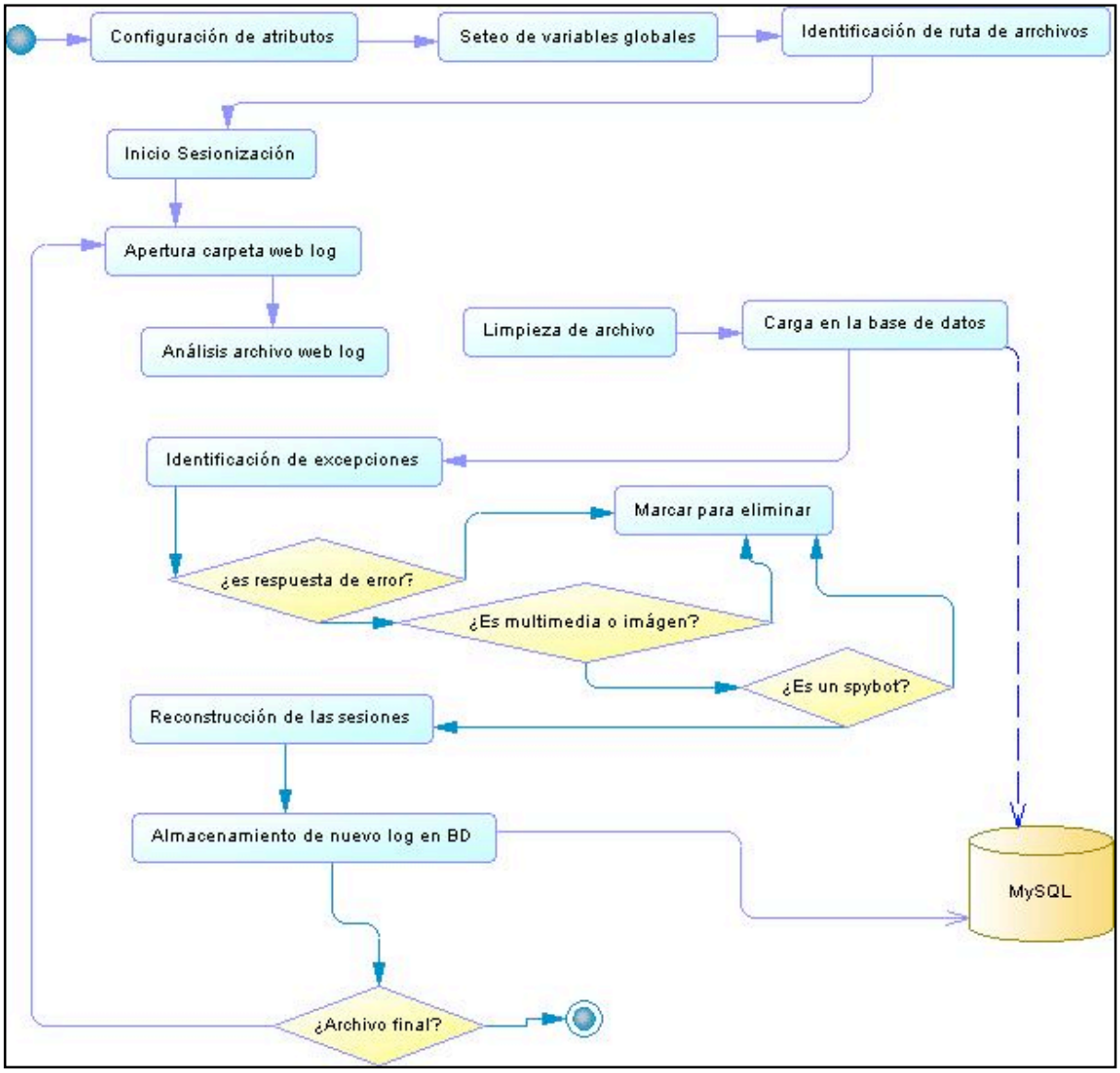
La reconstrucción de las sesiones a partir de las entradas del archivo de log se hace sobre la base de las heurísticas mencionadas en la sección 2.1.3. Además, cabe recordar que en la etapa de extracción el archivo de log fue limpiado de entradas erroneas o relacionadas a *crawlers*, robots, o bien, a objetos que no fueran páginas web. En la Figura 19 se puede observar el flujo completo del proceso de sesionización.

Tokenización

La representación de un texto como un set de palabras se denomina tokenización. Este proceso se puede subdividir en tres etapas que lo conforman:

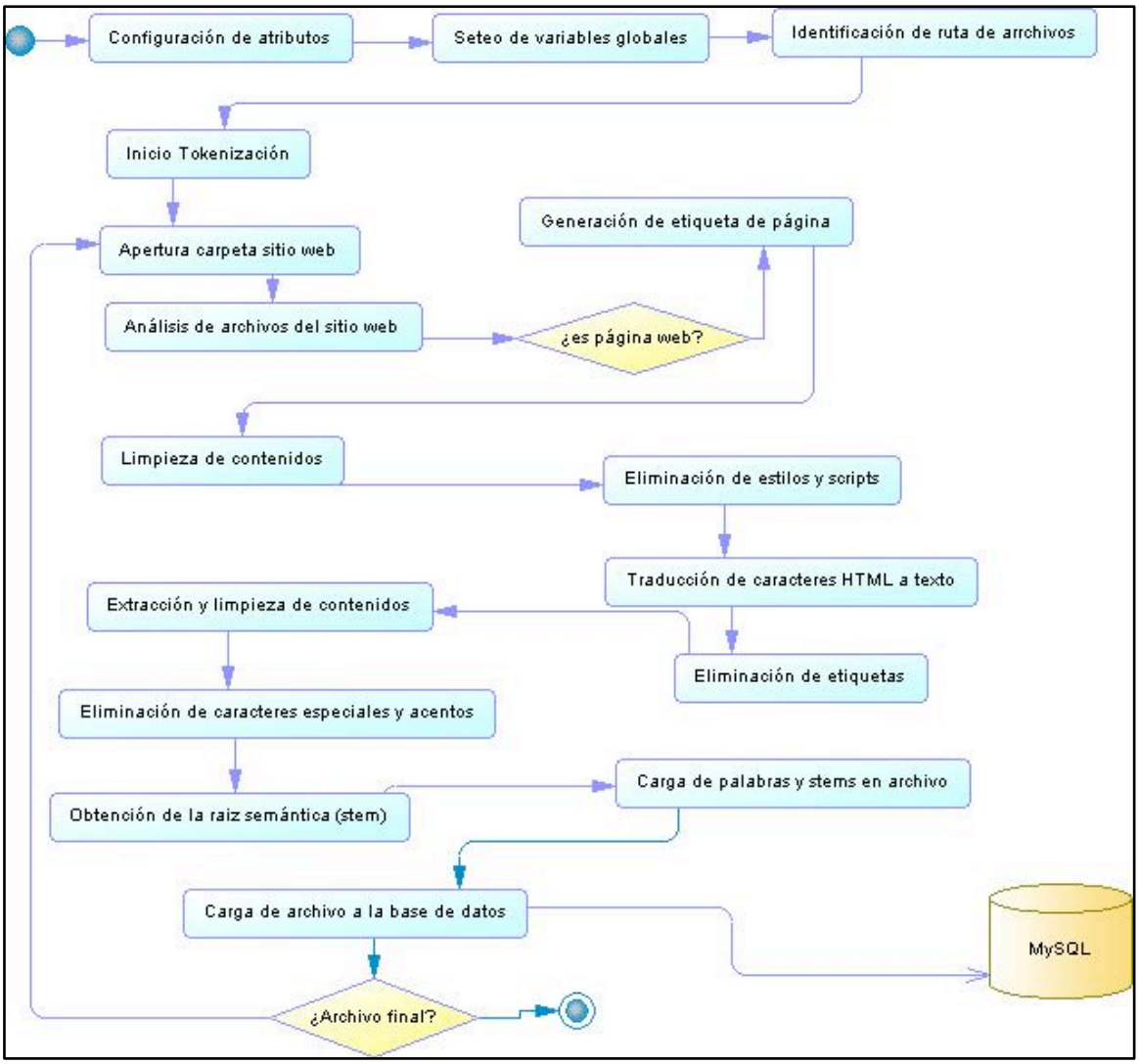
- **Etapa 1: cargar las páginas que serán tokenizadas**. Se debe determinar cuál es la carpeta contenedora de las páginas que se analizar, de manera de comenzar a leer cada uno de los archivos y comenzar a procesarlos.
- **Etapa 2: Extraer el contenido de cada página.** Se almacena en una variable el código HTML de cada página, que será posteriormente filtrado. Se ejecuta la limpieza de caracteres especiales, código en forma de *script* o *style*, además de la eliminación de etiquetas HTML. De esta manera, se tiene una variable que contiene un set de palabras filtradas a las cuales se debe procesar para llevarlas a su raiz.
- **Etapa 3: Transformar el contenido en un set de palabras.** Se aproxima cada una de las palabras a su raiz semántica mediante la *stemización*.

Figura 19: Flujo del proceso de sesionización [19].



Fuente: Elaboración Propia

Figura 20: Flujo del proceso de tokenización [19].



Fuente: Elaboración Propia

De esta manera, se obtiene el contenido de cada una de las páginas transformado en un set de palabras llevadas a su raíz. Esto nos permitirá analizar el peso relativo de cada una dentro del contenido del sitio en su totalidad. En la Figura 20 se puede observar el flujo completo del proceso de *tokenización*.

4.3.3 CARGA DE LOS DATOS

Una vez que los datos ya están procesados, corresponde cargarlos en el repositorio final desde donde se alimentan los reportes.

Para ello, el diseño de la carga comienza con el llenado de las dimensiones para luego cargar la tabla *fact* que, en este caso, corresponde a los *clickstream events.*

4.4 PRESENTACIÓN DE LA INFORMACIÓN

Para completar el diseño genérico del webhouse, se deberá determinar la manera en que la información se le presentará a los distintos usuarios.

Dicha información se puede clasificar en dos tipos: indicadores y cubos, y a continuación se definirá la interacción de cada una con los perfiles definidos para el uso del repositorio.

4.4.1 INDICADORES

Los indicadores son valores numéricos obtenidos a partir de la medición de variables de interés para la organización.

De acuerdo a los requerimientos de información presentados en 4.1, se pueden establecer algunos indicadores que deben formar parte del webhouse:

1) Número de sesiones encontradas por periodo.
2) Número de páginas visitadas en promedio por las sesiones.
3) Bytes transmitidos por las sesiones de un periodo.
4) Tiempo promedio de duración de las sesiones de un periodo.

4.4.2 RANKINGS

Otro elemento que puede aportar para el análisis, es el uso de rankings, es decir, listas ordenadas de elementos sobre los cuales se tiene información.

Algunos rankings que satisfacen requerimientos de información son:

1) Peso relativo de las 20 palabras más importantes.
2) Sets de páginas más recurrentes.
3) Páginas más visitadas por las sesiones por período.

4.4.3 CUBOS DE INFORMACIÓN

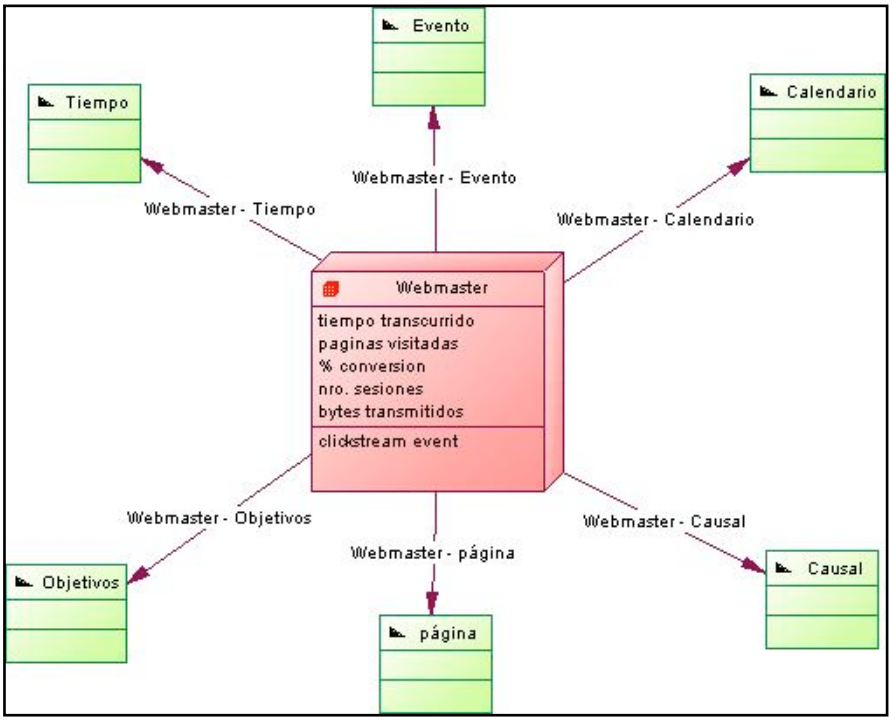
Como se describió anteriormente, este diseño conceptual propone la construcción de un cubo para cada perfil de usuario.

Para el caso del usuario del administrador o webmaster, se propone un cubo formado por las siguientes dimensiones (Figura 21):

1) Tiempo
2) Calendario
3) Objetivo
4) Causal
5) Usuario
6) Evento

Por su parte, las medidas que se pueden obtener a través de las dimensiones son, para este usuario, cinco: tiempo transcurrido, páginas visitadas, porcentaje de conversión, número de sesiones y bytes transmitidos.

Figura 21: Cubo de información para el perfil webmaster.



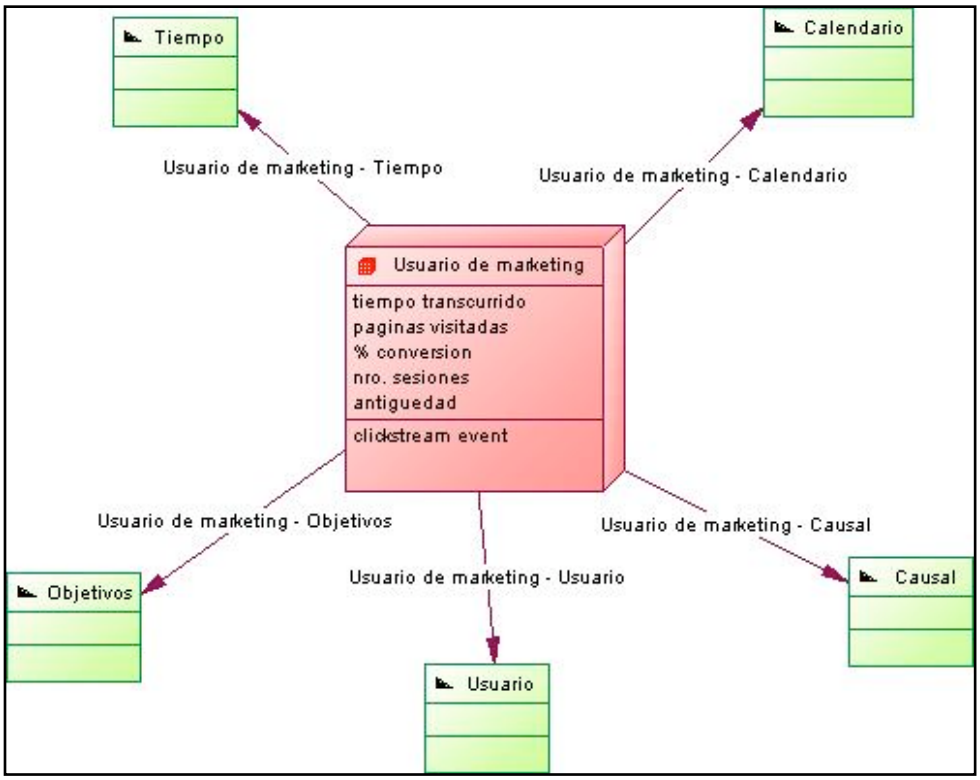
Fuente: Elaboración Propia

Para el caso del usuario del área de marketing, se propone un cubo formado por las siguientes dimensiones (Figura 22):

1) Tiempo
2) Calendario
3) Objetivo
4) Causal
5) Usuario

Por su parte, las medidas que se pueden obtener a través de las dimensiones son, para este usuario, cinco: tiempo transcurrido, páginas visitadas, porcentaje de conversión, número de sesiones y antigüedad del usuario. Cabe destacar que se requiere de datos acerca de usuarios registrados para la conformación de este cubo.

Figura 22: Cubo de información para el usuario de marketing.



Fuente: Elaboración Propia

CAPÍTULO 5 - CONSTRUCCIÓN DEL WIR

El modelo conceptual propuesto en el capitulo anterior se aplicó para un sitio web real con un alto volumen de tráfico de visitantes, compuesto por 212 páginas escritas en español. Éste pertenecía a una organización de la industria bancaria cuyas transacciones se efectuaban exclusivamente de manera electrónica. Se debió adaptar el modelo genérico de datos y el proceso de ETL para el caso particular de este sitio y al tipo de datos de los que se disponía. Por ejemplo, no se tuvo acceso a datos respecto de usuarios registrados del banco pues esto supone violación al secreto bancario. Sólo se accedió a datos de visitantes anónimos que navegaron por la página entre los meses de enero y abril del año 2003.

Para la implementación, se debió definir una arquitectura del sistema, además de la necesidad de contar con diversas componentes de software que entregaran la funcionalidad requerida para cada una de las actividades (ETL, generación de indicadores y cubos, además de la presentación de dicha información) .

Para ello se definió un diseño lógico del sistema y una arquitectura adecuada para su funcionamiento. Sobre esa base se probó el modelo conceptual.

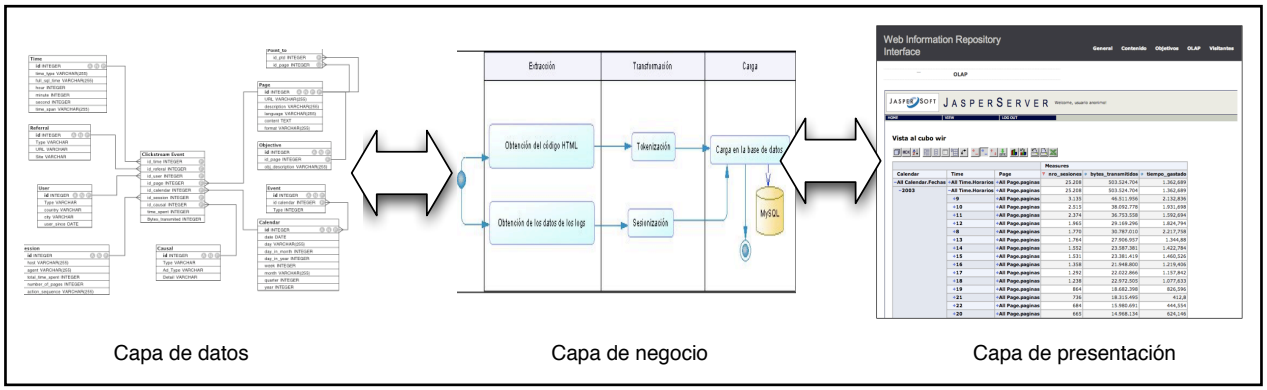
5.1 ARQUITECTURA

La definición de la arquitectura comprende la elaboración de un marco de diseño y desarrollo que contenga las componentes que dan forma al sistema, como también, sus interrelaciones. Estas componentes pueden ser de hardware, software, comunicación o seguridad.

Un enfoque de diseño y construcción de aplicaciones ampliamente utilizado es el que separa su implementación en tres tipos de servicios: datos, lógica y visualización. Esta separación supone una arquitectura de tres capas, una para cada tipo de servicio (capa de datos, capa media y capa de presentación). Para cada una de ellas se debe definir componentes de hardware y software que aseguren el correcto funcionamiento del sistema, como también, la conectividad entre cada uno de los módulos que lo componen.

La utilización de una arquitectura de tres capas posee múltiples ventajas por sobre otro tipo de arquitecturas. En particular, resulta beneficiosa su utilización en sistemas que requieren de flexibilidad respecto de los usuarios que lo utilizarán y, por su parte, flexibilidad para el acceso a distintas fuentes de datos, incluso permitiendo el uso de más de un tipo de motor para su obtención.

Figura 23: Diseño de tres capas para un webhouse.



Fuente: Elaboración Propia

Cabe destacar que la implementación de una arquitectura de tres capas no necesariamente implica que éstas deben estar separadas físicamente, es decir, dos capas pueden estar en un mismo servidor, pero desde el punto de vista lógico, fueron desarrolladas de manera separada.

En la Figura 23 se observa una representación de las tres capas de un webhouse: La capa de datos corresponde al modelo estrella y DSA. Por su parte, la capa de negocio corresponde a todo el procesamiento de los datos e información de la capa de datos que será presentada en la siguiente capa, en donde se muestra la información requerida por un usuario final.

Fuente: Elaboración Propia

Cabe destacar que, para cada una de las capas, existen múltiples alternativas de software para implementar los módulos que las componen. Para este trabajo, se optó por la utilización de herramientas de código abierto, que, al no tener costo de licencia, poseen una gran aceptación y comunidad de usuarios, lo que facilita el desarrollo.

5.1.1 CAPA DE DATOS

Un ámbito crucial para el funcionamiento de un repositorio de información es la capa de datos, encargada del almacenamiento de los datos e información que servirán de entrada para el procesamiento en las siguientes capas y, finalmente, la generación de reportes.

Dado lo anterior, se distingue que, en el caso particular que se aborda en esta investigación, la capa de datos debe responder a dos tipos de requerimientos:

a) requerimientos de datos. En esta capa se almacenarán los datos que sirven de entrada para el proceso de ETL.

b) requerimientos de información. También se debe almacenar la información resultante del proceso de ETL, que servirá de base para los reportes construidos para el usuario final del sistema.

Ambos requerimientos, dado el enfoque ROLAP del modelo propuesto, requieren de un motor de base de datos relacional y de un servidor que almacene los datos e información que se necesiten.

Componentes de hardware

Para la capa de datos se necesita un servidor capaz de albergar la totalidad de los datos e información que implica este sistema, como también, su crecimiento futuro. Además, el servidor debe satisfacer las consultas provenientes de las otras capas respecto de los datos e información que contiene.

Componentes de sofware

Se debe definir un motor de base de datos que satisfaga correctamente las necesidades de un webhouse basado en un enfoque relacional, es decir, debe responder de manera satisfactoria y rápida consultas en lenguaje SQL a un modelo relacional, que es donde opera la data staging area, y a uno snowflake, lugar donde opera el repositorio de información.

Entre los motores de bases de datos de código abierto, MySQL y PostgreSQL poseen ventaja por sobre sus alternativas, contando con excelentes niveles de funcionalidad. Siendo ambas buenas alternativas, MySQL responde con mayor rapidez a grandes consultas, además de satisfacer todos los requerimientos necesarios para la correcta implementación de este trabajo.

Por ello, se optó por la utilización de MySQL versión 5 para la implementación de la capa de datos en términos de software.

5.1.2 CAPA MEDIA

La capa media es la encargada del procesamiento de los datos que se presentarán en la siguiente capa.

Para el caso de un webhouse, la capa media debe establecer una conexión eficiente con la base de datos relacional, implementar el proceso de ETL, además de implementar los cubos de información con todo el procesamiento que ello conlleva.

Componentes de hardware

Para esta capa se necesita un servidor que sea capaz de ejecutar el proceso de ETL, además de los requerimientos de la capa de presentación, esto es, entregar los cálculos de indicadores necesarios y soportar la operación de los cubos de información.

Componentes de sofware

Para la implementación del ETL, se optó por utilizar el lenguaje PHP. Esto incluye implementar los módulos de sesionización, *tokenización* y *stemización*, procesamiento de los datos y la carga. Dicho lenguaje, al igual que MySQL, es prácticamente un estándar para implementación de la capa media en el caso más recurrente: aplicaciones web.

Para la operación de los cubos de información, que requieren de un servidor de aplicaciones Java, se utilizó Tomcat. Gracias a este software se podrán ejecutar las aplicaciones necesarias para generar los cubos y todo lo que esto conlleva respecto del procesamiento de datos.

Sobre Tomcat se ejecuta una aplicación encargada de la gestión de usuarios de los distintos reportes y cubos de análisis llamada Jasperserver que, a su vez, puede ejecutar consultas ROLAP sobre los cubos gracias a Mondrian, un motor de este tipo de consultas[7]. Se eligió ésta por sobre otras alternativas, debido a que Jasperserver entrega la funcionalidad requerida para la aplicación (generación de reportes y cubos, manejo de perfiles de usuario, etc) sin ser demasiado compleja, además de ser una herramienta de código abierto (Tabla 6).

Tabla 6: Cuadro comparativo de alternativas para generación de reportes y cubos.

Herramientas / Características	Jasperserver	Desarrollo propio sobre Mondrian	Pentaho BI suite
código abierto	√	-	√
Tiempo estimado de implementación	1 semana	1-2 meses	2 semanas
Cubos OLAP	√	Se deben desarrollar.	√
Reportes	√	Se deben desarrollar.	√
Perfiles de usuario	√	Se debe desarrollar.	√
Complejidad	Media	-	Alta

Fuente: Elaboración Propia

[7] ver Anexo C: Análisis de herramientas para olap.

Por último, para dar soporte a las operaciones de la interfáz se utilizó Wordpress, un gestor de contenidos escrito en PHP que permite la creación rápida de interfaces web.

5.1.3 CAPA DE PRESENTACIÓN

Para la presentación se optó por la interfaz de salida por defecto de Jasperserver, es decir, un navegador web (e.g. Firefox, Internet Explorer, Opera). De esta manera, prácticamente ningún usuario deberá instalar software adicional para generar reportes pues estos son accesados vía web, por lo que el procesamiento ocurre en el resto de las capas, que son ejecutadas en el servidor web y no en el computador del usuario.

Además, como se mencionó anteriormente, se utilizó Wordpress para la presentación de los indicadores y cubos OLAP.

5.2 MÓDULOS DE LA INTERFAZ[8]

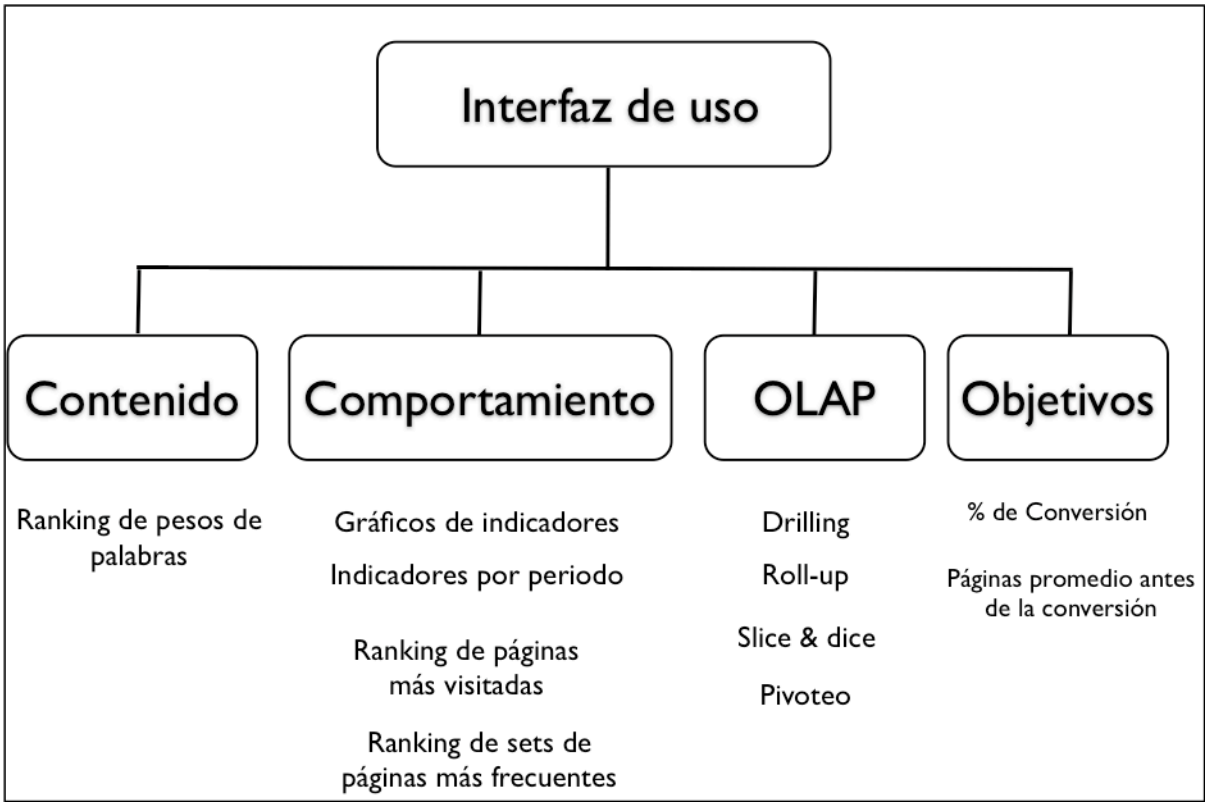
Se diseñó una interfaz para la interacción con los usuarios del WIR que, como se muestra en la Figura 24, consta de cuatro módulos[9]:

1) Módulo de análisis de las visitas, en donde se pueden observar indicadores de un rango de fechas como: tiempo promedio de duración de sesiones, número de sesiones analizadas, bytes transmitidos promedio, páginas visitadas por sesión en promedio. En la Figura 25 se muestra una imagen de este módulo.
2) Módulo de análisis del contenido de las páginas que componen el sitio, en donde se puede observar un ranking de palabras de acuerdo al peso relativo de cada una en el sitio.
3) Módulo de análisis de la conversión por objetivos, donde se puede observar el porcentaje de conversión por cada objetivo y el número de páginas promedio para alcanzarlo.
4) Módulo de navegación a través de los cubos utilizando Jasperserver, en donde se pueden ejecutar todas operaciones asociadas a OLAP (Figura 26).

[8] ver Anexo 67A: imágenes de la interfaz de la aplicación.

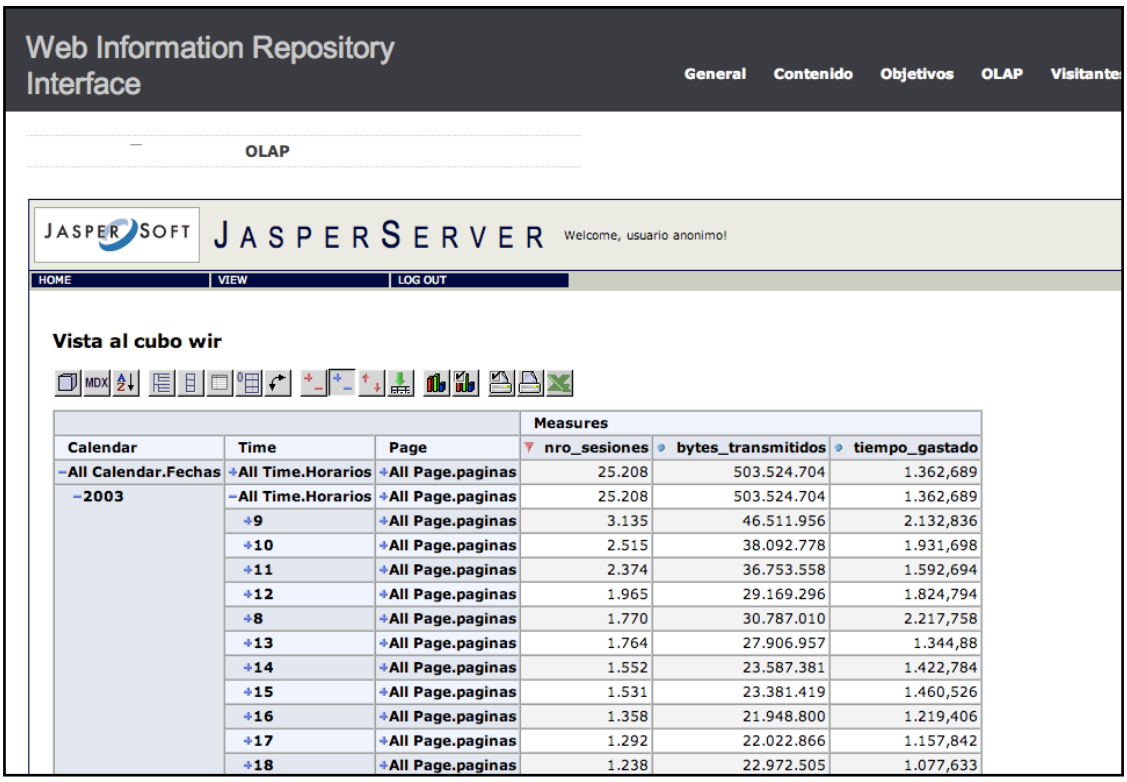
[9] Los módulos se construyeron obsevando características de herramientas disponibles en el mercado (Google Analytics, Webtrends, entre otras).

Figura 23: Estructura de la interfaz de interacción con el webhouse



Fuente: Elaboración Propia

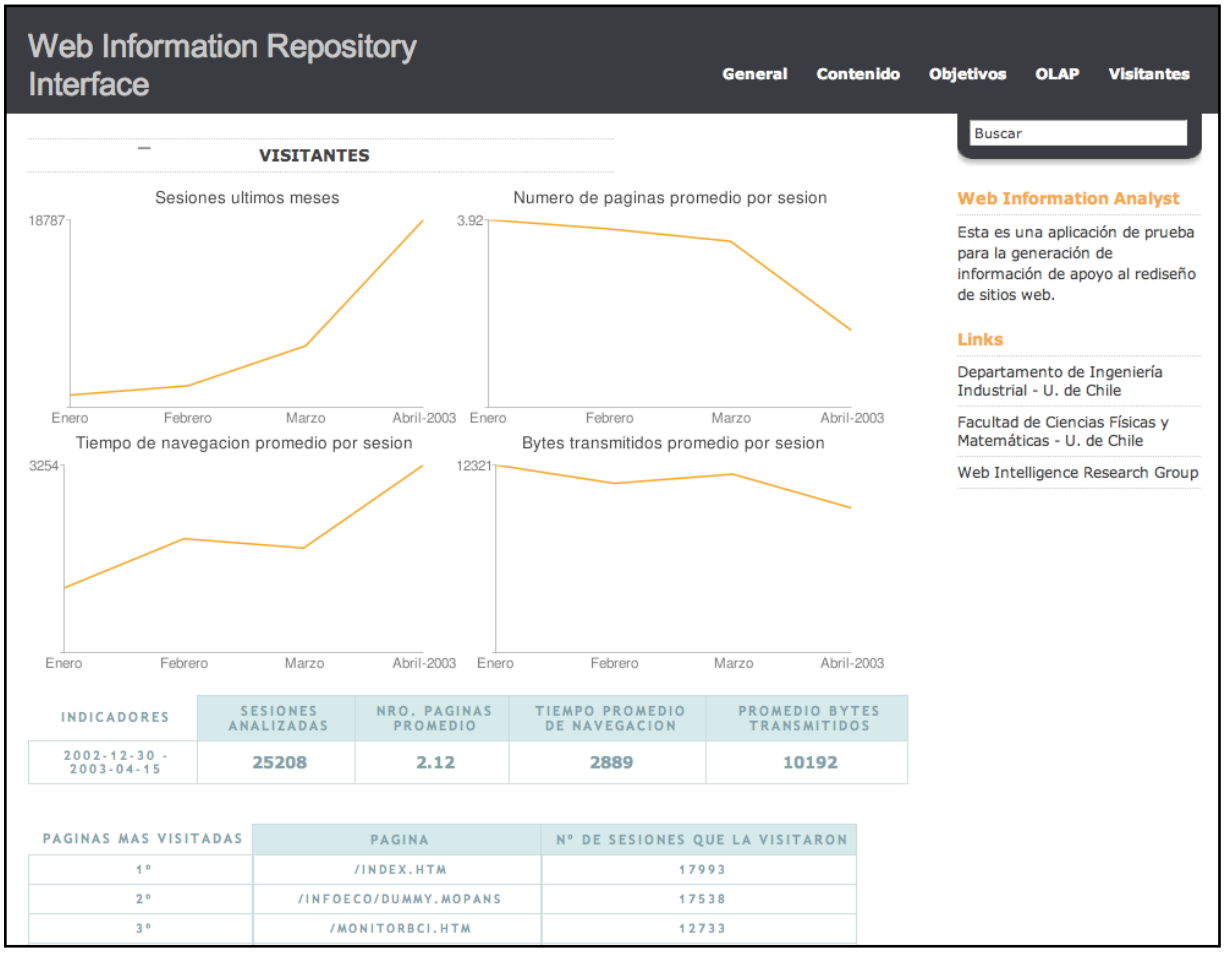
Figura 24: Módulo de análisis del cubo OLAP



			Measures		
Calendar	Time	Page	nro_sesiones	bytes_transmitidos	tiempo_gastado
−All Calendar.Fechas	+All Time.Horarios	+All Page.paginas	25.208	503.524.704	1.362,689
−2003	−All Time.Horarios	+All Page.paginas	25.208	503.524.704	1.362,689
	+9	+All Page.paginas	3.135	46.511.956	2.132,836
	+10	+All Page.paginas	2.515	38.092.778	1.931,698
	+11	+All Page.paginas	2.374	36.753.558	1.592,694
	+12	+All Page.paginas	1.965	29.169.296	1.824,794
	+8	+All Page.paginas	1.770	30.787.010	2.217,758
	+13	+All Page.paginas	1.764	27.906.957	1.344,88
	+14	+All Page.paginas	1.552	23.587.381	1.422,784
	+15	+All Page.paginas	1.531	23.381.419	1.460,526
	+16	+All Page.paginas	1.358	21.948.800	1.219,406
	+17	+All Page.paginas	1.292	22.022.866	1.157,842
	+18	+All Page.paginas	1.238	22.972.505	1.077,633

Fuente: Elaboración Propia

Figura 25: Módulo de análisis de las visitas



INDICADORES	SESIONES ANALIZADAS	NRO. PAGINAS PROMEDIO	TIEMPO PROMEDIO DE NAVEGACION	PROMEDIO BYTES TRANSMITIDOS
2002-12-30 - 2003-04-15	25208	2.12	2889	10192

PAGINAS MAS VISITADAS	PAGINA	Nº DE SESIONES QUE LA VISITARON
1º	/INDEX.HTM	17993
2º	/INFOECO/DUMMY.MOPANS	17538
3º	/MONITORBCI.HTM	12733

Fuente: Elaboración Propia

5.3 PRUEBAS Y ANALISIS DE LOS RESULTADOS

La operación del sistema se testeó en un servidor de prueba, con una carga de alrededor del 10% de los registros de los cuatro meses de uso. La operación fue satisfactoria en general, salvo la carga de la dimensión “sesión” en el proceso de ETL, que tomó un tiempo elevado comparado con el resto de las dimensiones (se demoró dos horas sobre los minutos que demoró cargar el resto de las tablas). Esto se puede deber a la necesidad de construir información (como “set de páginas visitadas” o “tiempo de duración”) que requiere de analizar para cada sesión la totalidad de la tabla.

Una vez concluido el proceso de ETL, es decir, con las tablas del repositorio cargadas, se procedió a la utilización de la aplicación.

Para los datos cargados, se encontraron los siguientes resultados (Tabla 7):

Tabla 7: Resultados encontrados con los datos de prueba.

Nº total de sesiones encontradas	25208
Nº promedio de páginas visitadas por cada sesión	2,12
Promedio de bytes transmitidos	10192
Página más visitada: index.htm	index.htm
Palabra con mayor peso: el nombre del banco.	el nombre del banco
Conversión por objetivo “hazte cliente”	0,478% recorriendo 5 páginas promedio.
Conversión por objetivo “contáctenos”	0,028% recorriendo 2 páginas en promedio.
Conversión por objetivo “preséntanos un amigo”	0,028% recorriendo 5,4 páginas promedio.
Hora de mayor tráfico	9:27 AM
Hora de menor tráfico	5:11 AM
Total de bytes transmitidos	503.524 megabytes.

Fuente: Elaboración Propia

5.3.1 ANALISIS DE LOS RESULTADOS Y PROPUESTAS DE REDISEÑO

Los datos encontrados arrojan una bajísima conversión por objetivos, que en total completa un 0,528%, es decir, de un total de 200 personas que ingresan al sitio, sólo una completa una secuencia exitosa para la organización. Además, para alcanzar el objetivo más exitoso, “hazte cliente”, se recorren 5 páginas en promedio. Si consideramos que el promedio de páginas visitadas por una sesión son 2,12 , se debería hacer más visible y alcanzable dicho objetivo, de manera que los usuarios completen sesiones exitosas en un mayor porcentaje.

Resulta coherente que la palabra con mayor peso sea el nombre del banco. Sin embargo, aparecen en el listado palabras como “mapa” (en el lugar 17) que no tienen mucha relación con el contexto de una organización bancaria. Otro punto que destaca respecto del contenido es la alta figuración de la palabra “plazo” y “comisión” en el undécimo y duodécimo lugar respectivamente, que resaltan aspectos negativos para la imagen del cliente por sobre las buenas características de los productos ofrecidos. A pesar de ser necesarias este tipo de palabras en el contenido, pues representan información relevante respecto de los productos, el hecho de que posean tanto peso puede ser un elemento interesante de ser contrastado con la estrategia de posicionamiento de la institución.

En la Figura 27 puede verse la “nube” de keywords, que representa gráficamente el peso de las palabras más importantes (a mayor peso, mayor tamaño en la nube).

Figura 26: Nube de keywords del sitio web de un banco (observación: se eliminó el nombre de la institución).

Fuente: Elaboración Propia

Por último, mencionar que el hecho de que la hora de mayor visita sea las 9:27 puede indicar que, generalmente, las personas ingresan al sitio al llegar a sus lugares de trabajo, lo que, eventualmente, podría ser aprovechado asociando la página de la institución a elementos que comúnmente son revisados a esa hora: precios de acciones, noticias más relevantes del mundo financiero, o incluso, el clima para el resto del día.

CAPITULO 6 – CONCLUSIONES

Crecientemente, los sitios web han adquirido mayor importancia como canal de marketing y ventas gracias a la alta penetración que ha tenido Internet en los hábitos de las personas. Las oportunidades a través de este canal son múltiples y, en un entorno altamente competitivo, resulta impensado que no sea incluido de manera protagónica dentro de la estrategia de las instituciones.

Debido a lo anterior, la necesidad de comprender las preferencias y necesidades de los clientes se ha trasladado también a la Web, exigiendo afinar los criterios con que se diseña y mejora la relación con los clientes que la utilizan.

A pesar de que existen múltiples herramientas para el análisis del tráfico de un sitio, el enfoque predominante para la construcción de esta información es proactivo, es decir, mediante la utilización de métodos que invaden la privacidad de los visitantes, pues requieren de la introducción de código en sus computadores de manera de poder hacer seguimiento de sus acciones.

Mediante la construcción de un repositorio de información web que utilice exclusivamente datos provenientes de las interacciones de los visitantes con el servidor web y datos emanados del código de las páginas, se puede construir la información necesaria para el análisis del comportamiento de los visitantes de un sitio, del contenido y de la estructura del mismo. Utilizando dicha información se puede avanzar hacia una comprensión de las preferencias de quienes visitan un sitio, generando una importante fuente de ventajas competitivas respecto de quienes no efectúan este tipo de análisis. Otros beneficios pueden relacionarse con el aumento de la venta cruzada de productos o, en el caso de un sitio de la industria bancaria, descongestionar las sucursales físicas.

Dentro de los posibles enfoques de análisis, en esta investigación se implementó el análisis estadístico de los datos provenientes de un sitio. Utilizando la arquitectura data warehouse y la metodología para construcción de este tipo de repositorios, se lograron los objetivos planteados para esta investigación, siendo la implementación del modelo la etapa más compleja y que requirió mayor dedicación.

De la aplicación del modelo conceptual desarrollado, se obtuvieron las siguientes conclusiones generales:

1) **La importancia de la conversión por objetivos como indicador del desempeño de un sitio web**. El monitoreo de este indicador debe ser el centro de cualquier análisis estadístico de los datos de un sitio, pues es una medida del

cumplimiento de los objetivos específicos de este canal. Una correcta definición de objetivos, sumado a su incorporación como prioridad en el diseño de un sitio puede generar un aumento en la efectividad de la Web como canal de ventas, con todos los beneficios en términos de menores costos y atención sin restricción horaria que ésta entrega. Puede considerarse la posibilidad de agregar una dimensión al modelo relacionado con ella, que almacene aspectos relevantes, como, por ejemplo: porcentaje histórico de conversión, máxima conversión, justificación de por qué una página es objetivo, etc.

2) **La estrategia reactiva no sólo permite generar de manera satisfactoria la información necesaria para el análisis de un sitio, sino que, además, posee la enorme ventaja de permitir la comparación temporal de distintas configuraciones del sitio.** Al construir un repositorio con arquitectura data warehouse se asegura la disponibilidad de la información para ser comparada en el tiempo. Esto permite acercarse a una medida objetiva acerca de si un rediseño mejoró o no los indicadores relevantes para la organización.

3) **Para operar en toda su potencialidad, se debe agregar directamente datos al repositorio que construyan las dimensiones no generadas a partir de los datos de provenientes de la Web.** Este es el caso de las dimensiones referidas a eventos en el tiempo que pueden explicar cambios en los indicadores de desempeño. Este es el caso, también, si se desea efectuar una medición de la efectividad de campañas publicitarias a través de este canal (banners, links pagados, etc).

4) **Se debe construir sitios web "bien formados" para el webhousing,** de manera de poder construir toda la información necesaria para el análisis. Esto implica optar por una menor cantidad de elementos que dificultan la construcción de la información, como pop-ups, animaciones flash, etc. Esto no excluye la incorporación de elementos multimedia, pero deben estar complementados con una buena política de incorporación de metadatos respecto de cada objeto.

Como recomendaciones respecto del diseño e implementación de una aplicación de estas características, podemos mencionar:

1) **Contrastar la información construida a partir de datos obtenidos con una estrategia reactiva con información generada con métodos proactivos.** De esta manera, se puede medir la calidad con que se están reconstruyendo las sesiones y, de esta manera, calibrar de mejor manera los algoritmos.

2) **Probar la aplicación con la totalidad de los datos de un sitio, considerando un tiempo prolongado.** La aplicación construida en esta aplicación operó correctamente con los datos de prueba correspondientes a tres meses de operación. Para avanzar hacia el diseño final de una aplicación de estas características se debe ejecutar cargas mayores que aseguren un buen desempeño en escenarios reales de sitios con alto nivel de tráfico.

3) **Implementar la integración con datos operacionales de otras áreas de la organización**, de manera de enriquecer los posibles análisis.

4) **Almacenar las conclusiones de diseño y contenido** obtenidas a partir de la información, de manera de no volver a cometer los mismos errores y crear conocimiento para la organización. Se puede, incluso, generar una Wiki[10] con dicha información.

5) **Avanzar hacia la creación de gráficos a partir de la información enfocados al análisis de un usuario final**. Los gráficos generados en este desarrollo carecen de la flexibilidad requerida para este tipo de aplicaciones.

Como posibilidades futuras en la línea de esta investigación, se distinguen las siguientes:

1) **Extensión del modelo genérico a objetos web, incluyendo imágenes, archivos de audio, video, etc,** de manera de incorporar elementos que generan mucha atracción para los visitantes y que cada vez están más presentes en los sitios debido al aumento de las velocidades de navegación.

2) **Enriquecimiento del análisis incorporando métodos de minería de datos que operen sobre la información del repositorio,** de manera de encontrar conocimiento no trivial proveniente de la gran cantidad de datos que genera el uso, contenido y estructura de un sitio web.

3) **Efectuar una comparación del desempeño del modelo desarrollado en esta investigación con el de otros modelos,** por ejemplo, considerando una tabla fact con un grano más grande, que almacene otros eventos, etc.

---

[10] Plataforma colaborativa que permite creación, edición y eliminación de contenido por múltiples usuarios.

4) **Re-evaluar supuestos de la bibliografía que pueden estar obsoletos** como, por ejemplo, el asumir que una navegación dura máximo 30 minutos.

5) **incorporar las nuevas tendencias de diseño e implementación de páginas web** (AJAX[11], estructura de blogs, streaming, etc) al estudio de las preferencias de los usuarios. Éstas son cada vez más utilizadas y agregan múltiples desafíos respecto del tipo de información que se genera de las nuevas formas de navegación e interacción con las organizaciones: el usuario ahora también es capaz de generar contenido.

La tendencia apunta a que la utilización de la Web será cada vez un elemento más central en las empresas, organizaciones y personas, por lo que, con esta investigación, se pretendió aportar hacia el avance en la comprensión de las preferencias y necesidades de los visitantes, de manera de generar organizaciones más eficientes, competitivas y que satisfagan de mejor manera a quienes son su razón de ser: sus clientes.

[11] Complemento de JavaScript asincrónico con XML.

BIBLIOGRAFÍA

[1] AAS, K. Y EIKVIL, L. Febrero, 1999. Text categorisation : A Survey. Technical report, Norwegian Computing Center.

[2] AGRAWAL, P. y GUPTA, A. Noviembre , 1997. Modeling multidimensional databases. In Procs. 13th Int. Conf. Data Engineering ICDE, pp. 232-243.

[3] AMERICAECONOMIA-VISA. 2006. Informe sobre Comercio Electrónico en America Latina. Disponible en URL: http://pdf.americaeconomia.com/RepositorioAmeco/Ediciones/E_56/V_28/S_164/A_1572/325_Esp_ecomerce.pdf. Fecha de consulta: 07/03/2008.

[4] BAEZA-YATES, R. y RIBEIRO-NETO, B. 1999. Modern Information Retrieval. Addison-Wesley.

[5] BAKOS, Y. Agosto, 1998. The Emerging Role of Electronic Marketplace on the Internet. Communications of the ACM, 41(8). pp. 35-42.

[6] BERENDT, B., MOBASHER, B. y SPILIOPOULOU, M. Agosto, 2002. Web Usage Mining for e-business applications. Tutorial, ECMML/PKDD Conference.

[7] BERENDT, B. y SPILIOPOULOU, M. 2001. Analysis of navigation behavior in web sites integrating multiple information Systems. The VLDB Journal, 9. pp. 76-82.

[8] BERNERS-LEE et al. 1994. The world wide web. Communications of ACM, 37(8). pp. 76-82.

[9] BERSON, A., SMITH, S. y THEARLING, K. 2000. Building Data Mining Applications for CRM. McGraw-Hill, New York, USA, 1st Edition.

[10] BHOWMICK, S.S. et al. 1998. Web warehousing: Design and issues. In Procs. Int of ER Workshops. pp. 93-104.

[11] CATLEDGE, L.D. y PITKOW, E. 1995. Characterizing Browning behaviors on the world wide web. Computers Networks and ISDN System, 27. pp. 1065-1073.

[12] CAVERO, J.M. et al. 2004. Managing data mining technologies in organizations: techniques and applications, chapter: A multidimensional data wearehouse development methodology. Idea Group. pp. 188-201.

[13] CHATRANON, A. et al. 2001. Customer Relationship Management (CRM) and E-Commerce. In Proc. 1st International Conf. on Electronic Business, Hong Kong, China.

[14] CHENOWETH, T., SCHUFF, D y ST. LOUIS, R. 2003. A method for developing dimensional data marts. Communications of the ACM, 46(12). pp. 93-98.

[15] CODD, E.F. 1982. Relational database: A practical foundation for productivity. Communications of the ACM, 25(2). pp. 109-117.

[16] COLLIAT, S. 1996. Olap, relational and multidimensional database Systems. SIGMOD Record, 25(3). pp. 64-69.

[17] COOLEY, R., MOBASHER, B. y SRIVASTAVA, J. 1999. Data preparation for mining World wide web browsing patterns. Journal of Knowledge and Information System, 1. pp. 5-32.

[18] EC-REPORT. 2004. The European e-Business Report. A Portrait of e-Business in 10 Sectors of the EU Economy. Third Synthesis Report of the e-Business W@tch, European Commission.Disponible en URL: http://www.ebusiness-watch.org/key_reports/ documents/EBR04.pdf. Fecha de acceso: 07/03/2008.

[19] FERNANDEZ, J. I. 2007. Mejorando el contenido textual de un sitio web mediante la identificación de sus website keywords. Memoria para optar al título de Ingeniero Civil Industrial, Facultad de Ciencias Físicas y Matemáticas, Universidad de Chile.

[20] GALLAUGHER, J. y RAMANATHAN, S. 1996. Choosing a client/Server architecture: a comparison of two-tier and three-tier systems. Information Systems Management Magazine, 13(2). pp. 7-13.

[21] GOLFARELLI, M. y RIZZI, S. 1999. Designing the data warehouse: key stops and crucial issues. Journal of Computer Science and Information Management, 2(1). pp. 1-14.

[22] HARMSEN, S. 2001. Strategy in the Context of eCommerce. Directed Studies 42590, School of Business, Carleton University, Ottawa, Canada. http://www.svenharmsen.de/PDFs/590-eStrategy.pdf. Fecha de acceso: 09/03/2008.

[23] HIPPNER, J. 2004. CRM –Grunlagen, Ziele und Konzepte. In Hajo Hippner and Klaus D. Wilde editors, *Grunlagen des CRM. Konzepte und Gestalung*, Wiesbaden, Alemania. pp. 13-41.

[24] HOCHSZTAIN, E. et al. Mayo, 2003. A Framework to Integrate Business Goals in Web Usage Mining. First International Atlantic Web Intelligence Conf. AWIC 2003. Madrid, España. pp. 28-36

[25] HU, X. y CERCONE, N. 2004. A data warehouse/online analytical processing Framework for web usage mining and business intelligence reporting. International Journal of Intelligent Systems, 19(7). pp. 585-606.

[26] INMON, W.H. 1996. Building the data warehouse (2da edición). John Wiley and Sons, New York.

[27] KIMBALL, R. y MERX, R. 2000. The data webhouse toolkit. Wiley Computer Publisher, New York, USA.

[28] KIMBALL et al. 1998. The data warehouse lifecycle toolkit: expert methods for designing, developing and deploying data warehouses. John Wiley and Sons, New York, USA.

[29] LEVENE, M. y LOIZOU, G. 2003. Why is the snowflake schema a good data warehouse design? Information Systems, 28. pp. 225-240.

[30] MAIER, T. 2006. Modeling ETL for web usage analisis and further improvements of the web usage analisis process. Tesis para optar al grado de doctor. University Eichstät-Ingolstadt.

[31] PAL, S.K., TALWAR, V. y MITRA, P. Septiembre, 2002. Web mining in soft computing Framework: Relevance, state of the art and future directions. IEEE Transaction on Neural Networkd, 13(5). pp. 1163-1177.

[32] PAYNE, A. , Febrero, 2003. The Multi-Channel Integration Process in Customer Relationship Management. White Paper, Cranfield School of Management, Cranfield University, Cranfield, UK.

[33] PRICE WATERHOUSE. Technology Forecast Report, 1996.

[34] PEPPARD, J. Junio, 2000. Customer Relationship Management (CRM) in Financial Services. European Management Journal, 18(3). pp. 312-327.

[35] SALTON, G., WONG, A. y YANG, C.S. Noviembre, 1975. A vector space model for automatic indexing. Comunications of the ACM archive, 18(11). pp. 613-620.

[36] SEN, A. y SINHA, P. 2005. A comparison of data warehousing methodologies. Comunications of the ACM, 48(3). pp. 79-84.

[37] SCHÖGEL, M. SCHMIDT, I. y SAUER, A. 2004. Multi-Channel Management im CRM – Prozessorientierung als zentrale Herausforderung. Management von CRM Projekten. Handlungsempfehlungen und Branchenkonzepte, pp. 107–122.

[38] SCHUMACHER, J. y MEYER, M. 2004. Customer Relationship Management strukturiert dargestellt. 1st Edition. Springer, Heidelberg, Alemania.

[39] SRIVASTAVA, J. et al. Mayo, 2002. A case for Analytical Customer Relationship Management. Advances in Knowledge Discovery and Data Mining, Proc. of the 6th Pacific-Asia Conf. PAKDD 2002. Taipei, Taiwan. pp. 14-27.

[40] SPILIOPOULOU et al. 2003. A Framework for the evaluation of session reconstruction heuristics in web-usage análisis. INFORMS Journal on Computing, 15. pp. 171-190.

[41] TESTE, O. 2001. Towards conceptual multidimensional design in decision support Systems. Fifth East-European Conf. on Advances in Databases and Information Systems. Vilnius, Lithuania. pp. 25-28.

[42] VASSILIADIS, P., SIMITSIS, A. y SKIADOPOULOS, S. 2002. On the logical modeling of ETL processes. In Proc. 14th Int. Conf. on Advanced Information Systems Engineering, London, UK. pp. 782-786.

[43] VELASQUEZ, J.D. y PALADE, V. 2008. Adaptive web site: a knowledge extraction form web data approach. IOS Press, Netherlands.

[44] VELASQUEZ, J.D. et al. 2005. Towards the identification of keywords in the web site content: A methodological approach. International Journal of Web Information Systems, 1(1). pp. 11-15.

[45] VELASQUEZ, J.D. et al. Octubre, 2003. Using the kdd process to support the web site reconfiguration. In Proc. IEEE/WIC Int. Conf. on Web Intelligence, Halifax, Canada. pp. 511-515.

[46] WESTLAND, J.C. y CLARK, T.H.K. 1999. Global Electronic Commerce: Theory and Case Studies. The MIT Press, Cambridge, MA, USA, 1st edition.

A IMÁGENES DE LA INTERFAZ DE LA APLICACIÓN

Figura 27: Pantalla de bienvenida de la aplicación.

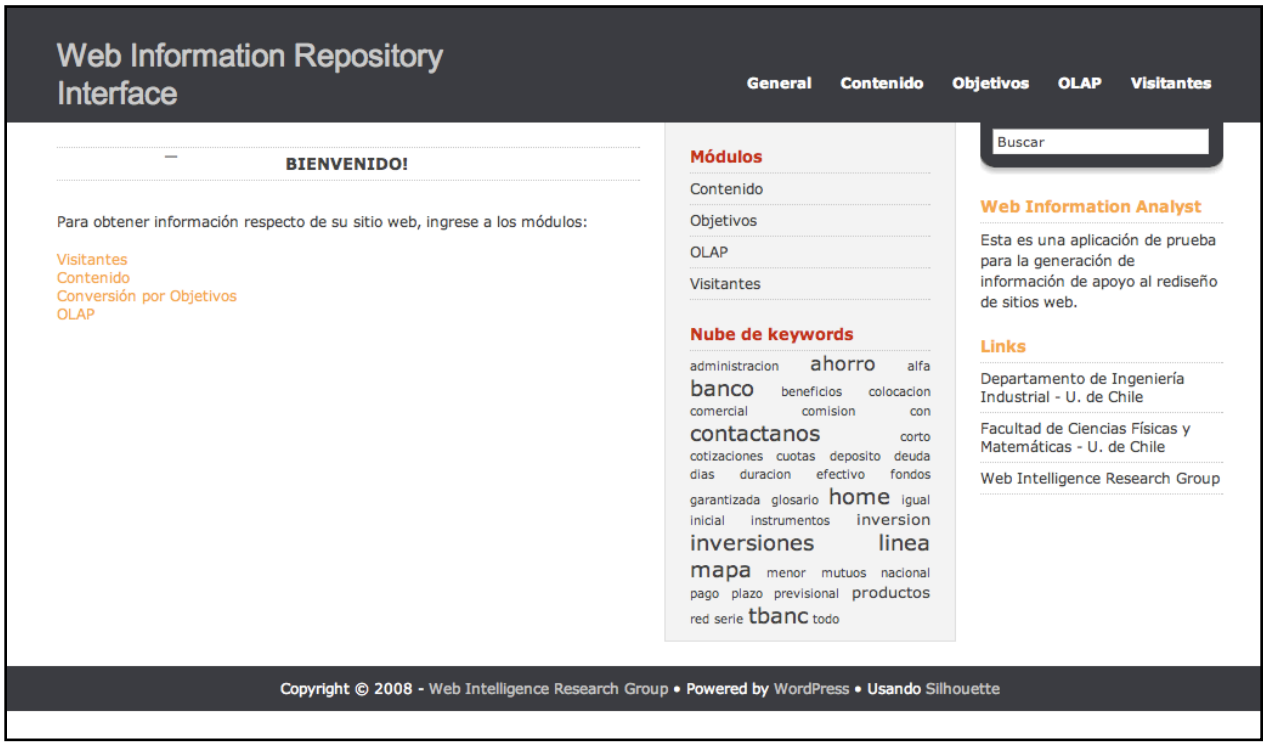


Figura 28: Módulo de contenido de la aplicación.

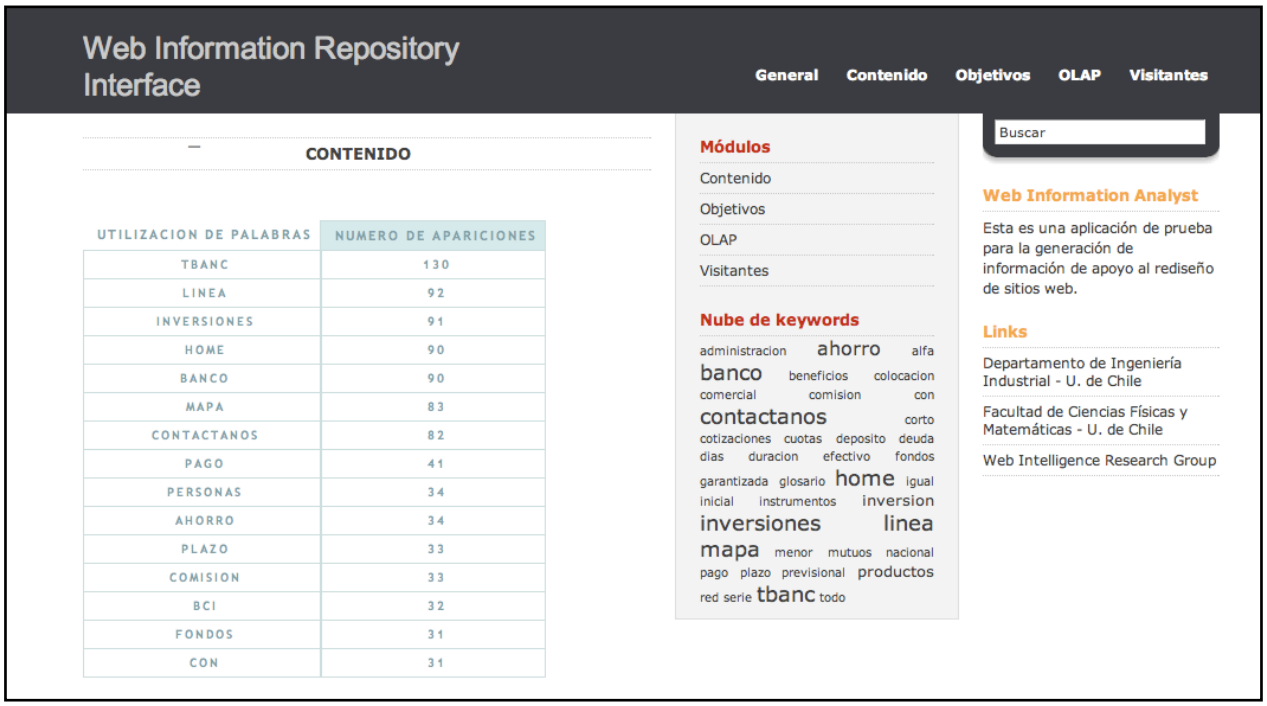


Figura 29: Modulo de entrada al análisis de visitas (rango de fechas).

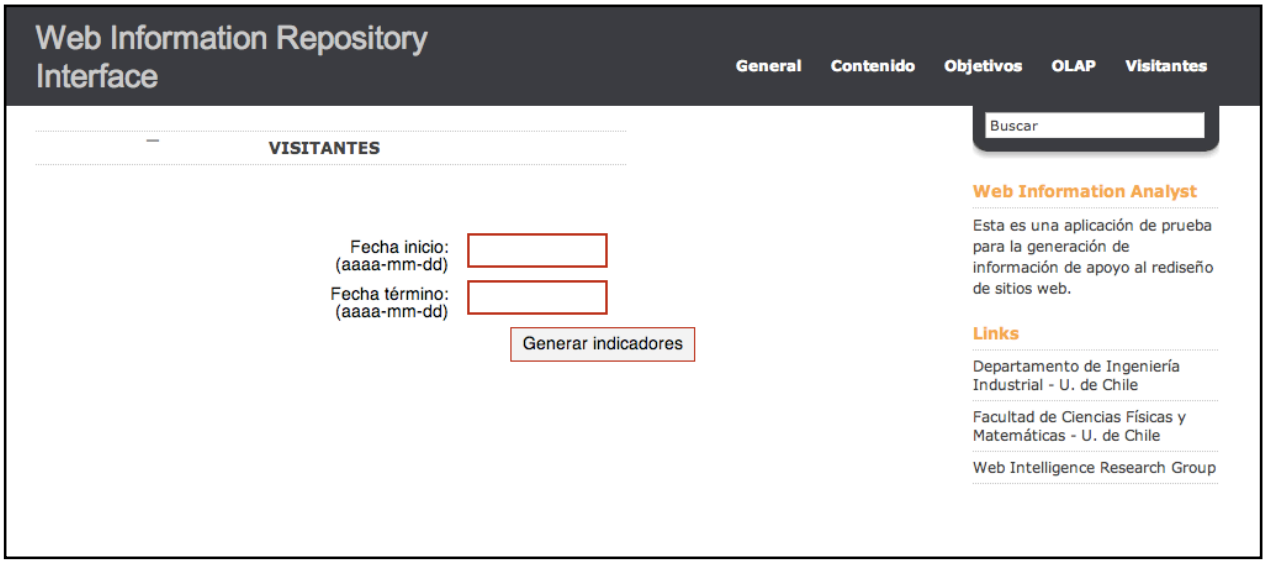
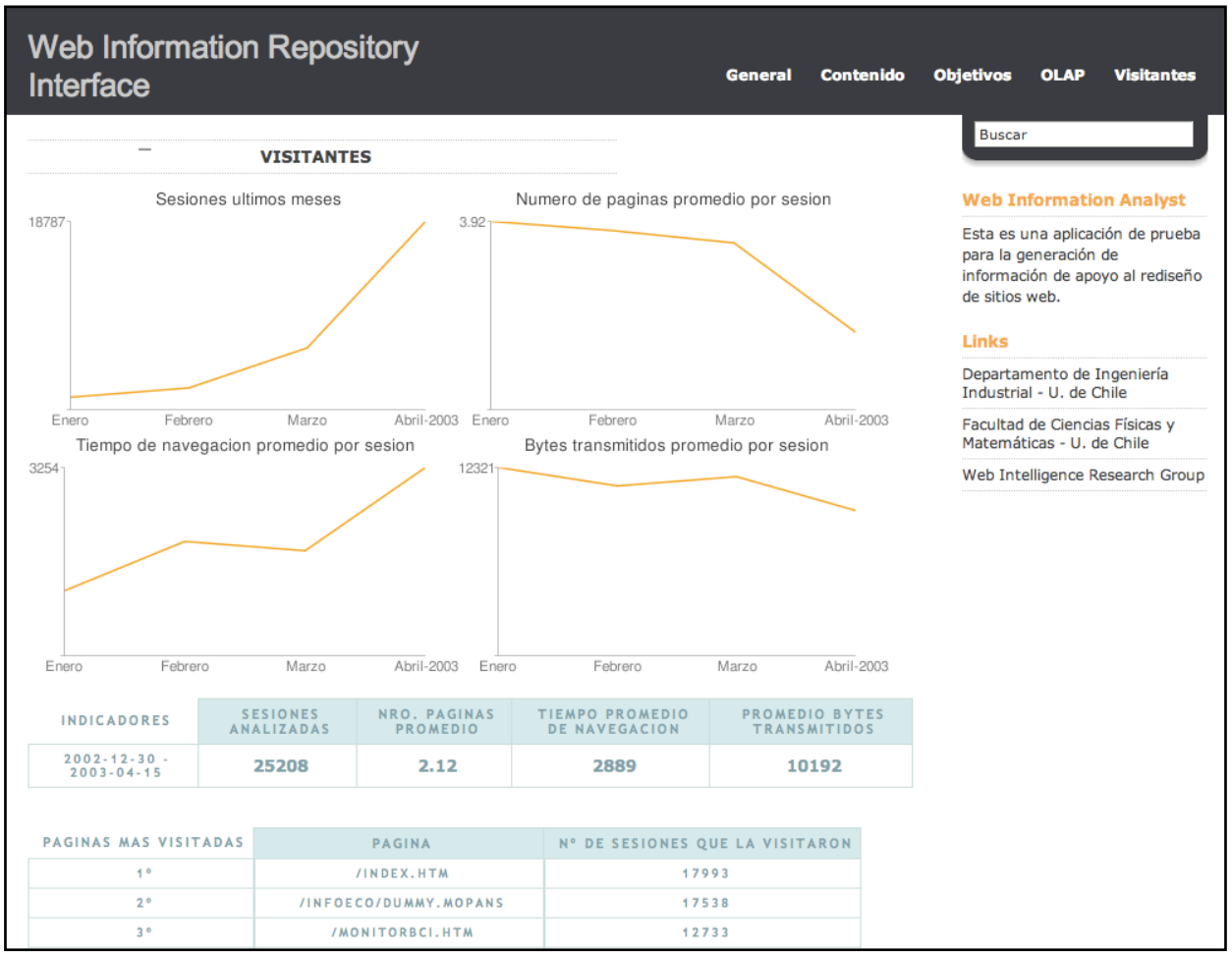


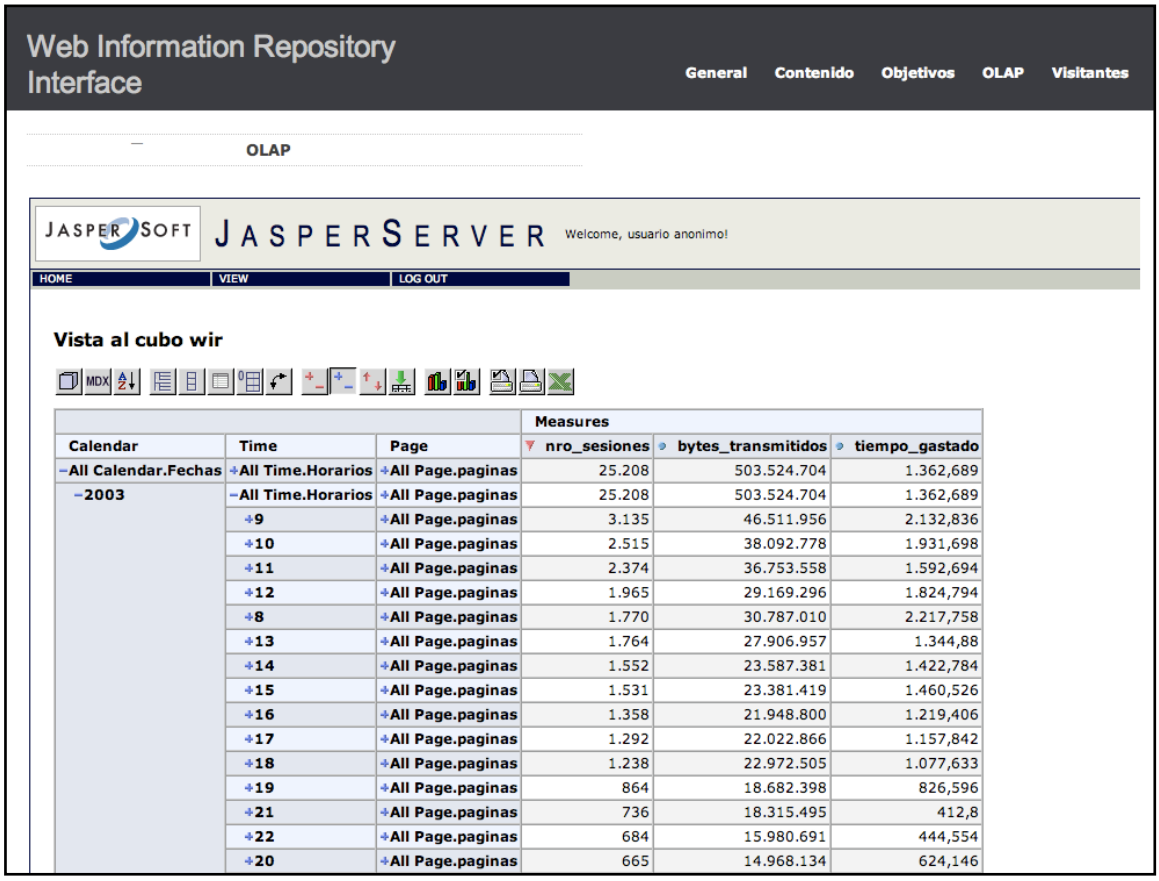
Figura 30: Módulo de análisis de sesiones de la aplicación.



INDICADORES	SESIONES ANALIZADAS	NRO. PAGINAS PROMEDIO	TIEMPO PROMEDIO DE NAVEGACION	PROMEDIO BYTES TRANSMITIDOS
2002-12-30 - 2003-04-15	25208	2.12	2889	10192

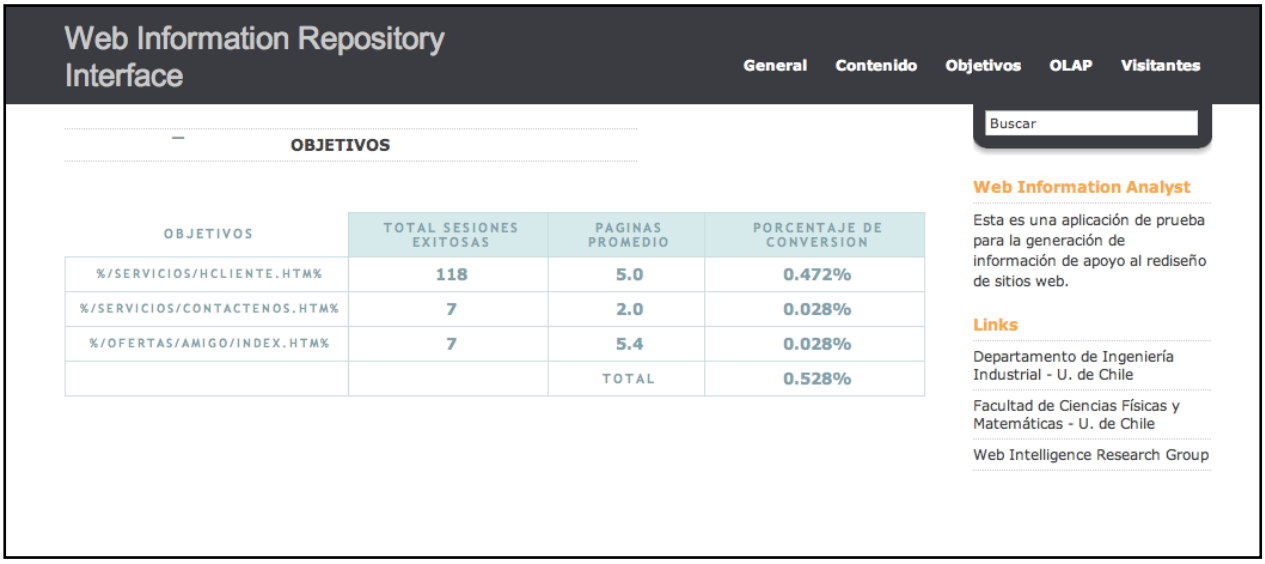
PAGINAS MAS VISITADAS	PAGINA	Nº DE SESIONES QUE LA VISITARON
1º	/INDEX.HTM	17993
2º	/INFOECO/DUMMY.MOPANS	17538
3º	/MONITORBCI.HTM	12733

Figura 31: Módulo de revisión de cubos OLAP.



			Measures		
Calendar	Time	Page	nro_sesiones	bytes_transmitidos	tiempo_gastado
-All Calendar.Fechas	+All Time.Horarios	+All Page.paginas	25.208	503.524.704	1.362,689
-2003	-All Time.Horarios	+All Page.paginas	25.208	503.524.704	1.362,689
	+9	+All Page.paginas	3.135	46.511.956	2.132,836
	+10	+All Page.paginas	2.515	38.092.778	1.931,698
	+11	+All Page.paginas	2.374	36.753.558	1.592,694
	+12	+All Page.paginas	1.965	29.169.296	1.824,794
	+8	+All Page.paginas	1.770	30.787.010	2.217,758
	+13	+All Page.paginas	1.764	27.906.957	1.344,88
	+14	+All Page.paginas	1.552	23.587.381	1.422,784
	+15	+All Page.paginas	1.531	23.381.419	1.460,526
	+16	+All Page.paginas	1.358	21.948.800	1.219,406
	+17	+All Page.paginas	1.292	22.022.866	1.157,842
	+18	+All Page.paginas	1.238	22.972.505	1.077,633
	+19	+All Page.paginas	864	18.682.398	826,596
	+21	+All Page.paginas	736	18.315.495	412,8
	+22	+All Page.paginas	684	15.980.691	444,554
	+20	+All Page.paginas	665	14.968.134	624,146

Figura 32: Módulo de conversión por objetivos de la aplicación.



OBJETIVOS	TOTAL SESIONES EXITOSAS	PAGINAS PROMEDIO	PORCENTAJE DE CONVERSION
%/SERVICIOS/HCLIENTE.HTM%	118	5.0	0.472%
%/SERVICIOS/CONTACTENOS.HTM%	7	2.0	0.028%
%/OFERTAS/AMIGO/INDEX.HTM%	7	5.4	0.028%
		TOTAL	0.528%

B CÓDIGO PROCESO DE CARGA DE DATOS

B.1 LLENADO DIMENSIÓN SESSION

```
<?php
        include 'database/open.php';
        include_once 'database/config.values.php';

        $borratabla=mysql_db_query('WIR', "TRUNCATE TABLE `Session`");
        $tabla=mysql_db_query('web analysis', "SELECT * FROM `weblog_analysis_logclean`");
        $countrows = mysql_num_rows($tabla);

        while($i < $countrows) {

        $result1 = mysql_db_query('web analysis', "SELECT `id_session`, `path`, `agent` ,
`page_nav_time`,`host` FROM `weblog_analysis_logclean` WHERE id_session='$i' ")      ;

        $stringwords ="";
    $countwords = mysql_num_rows($result1);
        echo $countwords;
        $nroword=0;
        $tiemponav=0;
        while($nroword < $countwords)
        {$words=mysql_fetch_row ($result1);
         $stringwords = $stringwords." --> ".$words[1];
         $tiemponav=$tiemponav+$words[3];
         $nroword=$nroword+1;}

  $query2= "INSERT INTO Session (`id`, `host` , `agent`, `total_time_spent`, `number_of_pages`,
`action_sequence`) VALUES ('$words[0]', '$words[4]' , '$words[2]','$tiemponav', '$nroword',
'$stringwords')";
                                    echo "<br>El insert a la BD: $query2";
                                    mysql_db_query('WIR',$query2);
                                    $i=$i+1;
}

?>
```

B.2 LLENADO DIMENSIÓN PAGE

```
<?php
        include 'database/open.php';
        include_once 'database/config.values.php';
        $borratabla=mysql_db_query('WIR', "TRUNCATE TABLE `Session`");
        $tabla=mysql_db_query('web analysis', "SELECT * FROM `weblog_analysis_logclean`");
        $countrows = mysql_num_rows($tabla);
```

```
          while($i < $countrows) {
          $result1 = mysql_db_query('web analysis', "SELECT `id_session`, `path`, `agent` , `page_nav_time`,`host` FROM `weblog_analysis_logclean` WHERE id_session='$i' ")      ;

          $stringwords ="";
    $countwords = mysql_num_rows($result1);
          echo $countwords;
          $nroword=0;
          $tiemponav=0;
          while($nroword < $countwords)
          {$words=mysql_fetch_row ($result1);
           $stringwords = $stringwords." --> ".$words[1];
           $tiemponav=$tiemponav+$words[3];
           $nroword=$nroword+1;}

  $query2= "INSERT INTO Session (`id`, `host` , `agent`, `total_time_spent`, `number_of_pages`, `action_sequence`) VALUES ('$words[0]', '$words[4]' , '$words[2]','$tiemponav', '$nroword', '$stringwords')";
                                    echo "<br>El insert a la BD: $query2";
                                    mysql_db_query('WIR',$query2);
                                    $i=$i+1;
}
?>
```

B.3 LLENADO DIMENSIÓN TIME

```
<?php

          include 'database/open.php';
          include_once 'database/config.values.php';

$hora = 00;
$minuto = 00;
$segundo = 00;

          for( $hora = 00 ; $hora < 24; $hora++ ){
               for( $minuto = 00; $minuto < 60; $minuto++ ){
                    for( $segundo = 00; $segundo < 60 ; $segundo++ ){
                              $instante=$hora.":".$minuto.":".$segundo;
                               $query = "INSERT INTO `Time` ( `full_sql_time`, `hour`, `minute`, `second`) VALUES('$instante','$hora','$minuto','$segundo')";
                               echo "$query<br>";
                                    mysql_query($query);
                                        }
                               }
```

```
                }
?>
```

B.4 LLENADO DIMENSIÓN CLICKSTREAM EVENT

```
<?php
        include 'database/open.php';
        include_once 'database/config.values.php';

$borratabla=mysql_db_query('WIR', "TRUNCATE TABLE `Clickstream Events`");
$i=0;
$result1 = mysql_db_query('web analysis', "SELECT `path` , `date` , `id_session`, `date` , `page_nav_time`,`bytes` FROM `weblog_analysis_logclean`")  ;

$countrows = mysql_num_rows($result1);

        while($i < $countrows) {
    $tuplas=mysql_fetch_row ($result1);
    echo $tuplas[0];
    $arrTime=explode(" ", $tuplas[1]);
    $hora=$arrTime[1];
    $query2= "INSERT INTO  `WIR`.`Clickstream Events` (
`id_page` ,
`id_time` ,
`id_session` ,
`id_calendar` ,
`time_spent` ,
`Bytes_transmited`
)
VALUES (
'$tuplas[0]',  '$hora', '$tuplas[2]',  '$tuplas[3]', '$tuplas[4]', '$tuplas[5]'
);
";
                                    echo "<br>El insert a la BD: $query2";
                                    mysql_db_query('WIR',$query2);
                                    $i=$i+1;
}
?>
```

C ANÁLISIS DE HERRAMIENTAS PARA OLAP

C.1 JASPER ANALYSIS

Características generales

- Motor para OLAP Relacional (ROLAP).
- Operación vía web.
- Incluye MySQL y Tomcat operando.
- Consultas y generación de cubos operan con estándares de la industria: MDX y XML.
- Compatibilidad con muchas RDBMS y sistemas operativos.
- Manejo seguro de usuarios basados en roles y privilegios.

Operaciones de análisis principales

- Drill- down
- Dril-Through
- Slice & dice
- Pivoteo
- Filtros
- Generación de gráficos
- Exportación PDF y Excel
- Cambio de ejes

Versiones

Está en versión Open Source y Professional.

Las diferencias están relacionadas con el soporte técnico, permiso para inclusión en otras herramientas comerciales, programas de actualización, garantía y algunas características avanzadas. Además, en la versión profesional hay soporte certificado multiplataforma, tanto para el sistema operativo, servidor de aplicaciones y RDBMS.

Características avanzadas de la versión pagada

- Interfaz de usuario mejorada, utilizando asincronismo javascript y XML (AJAX), que entrega opciones visuales potentes como, por ejemplo, *drag and drop* en interfaces web.

1. Gráficos interactivos y con posibilidades de *drill.*

2. Incluye herramienta de diseño de cubos OLAP.

3. Incluye módulo de administración de servidor OLAP.

Tabla 8: Comparación entre distintas versiones de JasperAnalysis

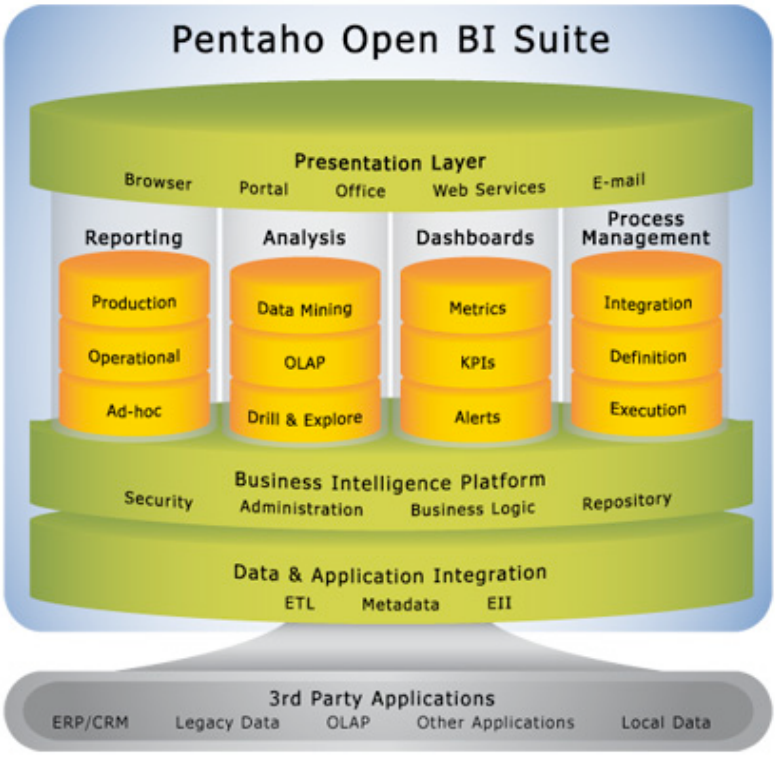
JASPERANALYSIS		PRODUCT EDITION	
		Open Source	Professional
Licensing and Pricing	License Type	GPL	Commercial
	Number of CPUs	N/A	•
	Number of OEM Customers	N/A	•
Benefits	Modifiable Source Code	•	•
	Community Support Forum	•	•
	Internal Use	•	•
	Commercial Embedding Rights		•
	Certified Platform Support		•
	Professional Technical Support		•
	Managed Release Cycle		•
	Legal Indemnity		•
	Advanced Features		•

Fuente: http://www.jaspersoft.com/JasperSoft_JasperAnalysis.html

C.2 PENTAHO BI SUITE 1.6

Características generales

- Motor OLAP relacional (ROLAP)
- Compatible con bases de datos propietarias como Oracle, DB2, SQL Server, NCR y otras.
- Compatible con motores de bases de datos de código abierto como MySQL, PosgreSQL y otros.
- Utiliza XML/A y MDX.
- Exportación a PDF, HTML, Excel, RTF o texto plano.
- Posee un diseñador de reportes (Penaho Report Designer) que utiliza *drag and drop*.
- Interface web basada en AJAX.
- Distribución de reportes basados en roles.



Fuente: www.pentaho.com

Versiones

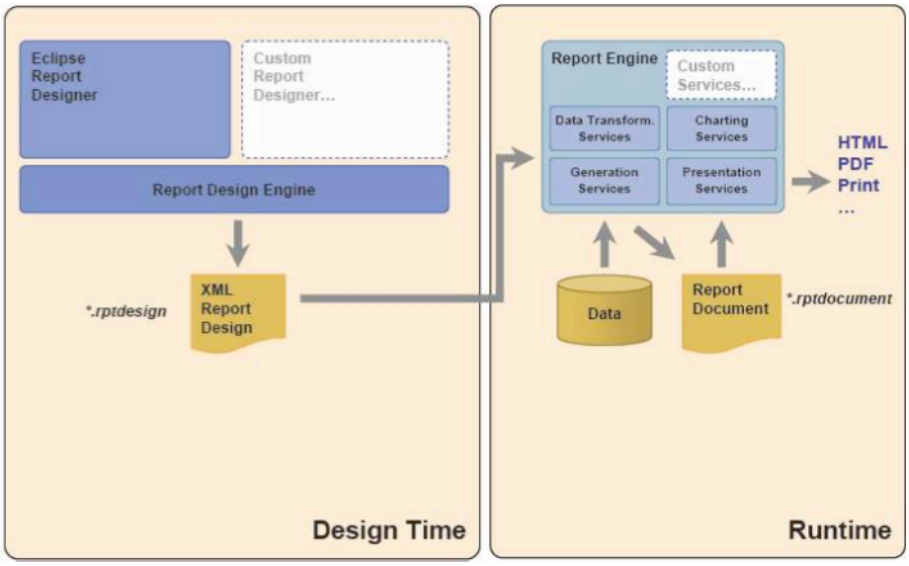
Sólo existe la versión open source.

C.3 ECLIPSE BIRT

Características generales

- Framework integrado con un IDE Eclipse para desarrollo de informes y un runtime para J2EE para verlos.
- Informes encapsulados en xml.
- Posibilidad de ser embebido.
- Open source.

Flujo de creación de reportes



Fuente: http://www.telefonica.net/web2/todobi/Oct07/Reporting_OS.pdf

C.4 CUADRO COMPARATIVO

	Jasper Reports/Server 2.0	Pentaho 1.6	BIRT 2.2
Generación de XML	si	si	si
Extensibilidad (vía API)	si	si	si
Wizards	si	si	si
Previsualizador	si	si	si
Integración Pentaho	si	nativa	si
Drag & Drop	no	si	si
Edición Gráfica	si	si	si
Edición XML	no	si	si
Funciones incorporadas	si	si	si
Soporte a reglas de negocio	si	si	si

	Jasper Reports/Server 2.0	Pentaho 1.6	BIRT 2.2
Gráfico	si	si	si
Lista	si	si	si
Tabla	si	si	si
Tablas dinámicas	si	no	si
texto	si	si	si
texto dinámico	si	si	si
imagen	si	si	si
Grilla	no	no	si
2D	si	si	si
3D	si	si	si
Dial	si	si	no

Fuente: http://www.telefonica.net/web2/todobi/Oct07/Reporting_OS.pdf