



UNIVERSIDAD DE BELGRANO

# Las tesis de Belgrano

Facultad de Tecnología Informática  
Carrera Ingeniería en Informática

“3S: SEO System Suites”  
Suite de herramientas de SW libre para  
facilitar decisiones de SEO

N° 469

Lucas Pablo Hernández

Tutor: Sergio Aguilera

Departamento de Investigaciones  
2010

Universidad de Belgrano  
Zabala 1837 (C1426DQ6)  
Ciudad Autónoma de Buenos Aires - Argentina  
Tel.: 011-4788-5400 int. 2533  
e-mail: [invest@ub.edu.ar](mailto:invest@ub.edu.ar)  
url: <http://www.ub.edu.ar/investigaciones>



## Índice

I.	Introducción .....	5
II.	Objetivos .....	7
III.	Alcance y límites .....	7
IV.	Marco teórico .....	8
o	¿Qué hace Google? .....	8
o	Crawling .....	9
o	Indexing .....	9
o	Serving .....	11
o	Meta Tags .....	12
o	URLs limpias .....	13
o	Links .....	14
o	Redireccionamiento Permanente .....	16
o	Contenido dinámico.....	16
V.	Propuesta de desarrollo de la tesina .....	17
VI.	Práctica .....	21
VII.	Conclusiones .....	37
VIII.	Bibliografía .....	39
IX.	Anexos .....	40
o	Manual de usuario.....	40
o	Diagrama de base de datos .....	41
X.	Glosario .....	42



## I. Introducción

Desde la década del 90, y especialmente desde la creación del buscador Google, los usuarios de Internet alrededor del mundo han ido cambiando su forma de navegar. Desde ese entonces la tendencia de navegar solicitando **direcciones Web** fue disminuyendo, dando lugar a la navegación utilizando **conceptos**. Esta revolución de los buscadores agilizó la evolución de las aplicaciones “punto com”, lo que trajo consigo un cambio trascendental tanto en los negocios como en el diseño y marketing de aplicaciones y sitios Web.

Hoy en día esta tendencia sigue creciendo, tanto que más del 70% de las navegaciones de los usuarios comienzan por un buscador [1].

Todas las empresas alrededor del mundo se vieron obligadas entonces a interesarse e informarse en el modo en que estos buscadores representaban sus sitios, la manera en que los “rankeaban” (“se posicionaban”, estaban posicionados en una búsqueda) y priorizaban ante otros o qué parámetros/atributos de sus sitios eran relevantes. Aquí nació una preocupación nueva para las empresas que querían obtener una ventaja en el mercado a través de la Web: ya no debían preocuparse solo por los contenidos de sus sitios y por la interfaz con el usuario, sino también por su predisposición correcta hacia los diferentes motores de búsqueda.

En la actualidad, los diferentes motores de búsqueda (liderados ampliamente por Google) son muy limitados en brindar recomendaciones o prácticas objetivas sobre como incrementar la posición de un sitio Web frente a otro (siempre y cuando estén en los mismos rubros). A pesar de su tan famosa popularidad, Google no deja de ser una empresa con fines económicos, por lo tanto solo se interesa en brindar información sobre PPC (Pay Per Click, todo lo que involucre publicidades pagas en Google.com). Más allá de esto, alrededor de la Web existen innumerables recomendaciones y técnicas a seguir. Una recomendación no es 100% concreta, y es imposible saber si estas iniciativas funcionan o no hasta meses después de haberlas implementado. Esto significa tiempo y recursos utilizados sin la garantía de que me den un beneficio al final del camino, algo muy difícil de defender al momento de analizar un presupuesto o de entrecruzar sectores técnicos y de negocio dentro de la empresa. Además, los buscadores han encontrado la manera de reconocer y penalizar los intentos por cumplir estas pautas con fines fraudulentos, como por ejemplo el hecho de repetir patrones o incluir contenido irrelevante con el único propósito de “incluir contenido”. También es importante destacar lo imposible que es crear un ambiente para testear este tipo de cambios, ya que no se puede aislar un sitio y a la vez ver como reacciona en un buscador.

La dificultad de estos conceptos no radica en su complejidad técnica, sino en buscar una manera de satisfacer estos requerimientos mínimos sin perder contenido útil del sitio, sin que el sitio pierda atracción visual y/o de contenidos, y dándole un **beneficio** final a los usuarios, como sostiene [2]. Esta es la razón por la cual no siempre funcionan, y por la cual los buscadores nunca revelan sus algoritmos libremente. Quizás el punto más importante es que varias de estas recomendaciones están compuestas por el “**qué** hacer” y no el “**cómo** hacerlo”, “**cómo** organizarse” y “**cómo** obtener información concreta para el ROI (retorno de inversiones: si vamos a realizar algo que necesita un presupuesto y que, además, tardará meses en dar frutos, debemos estar capacitados para indicar cuánto tiempo y recursos llevará recuperar lo invertido)”. Obviamente, nadie de esto de forma gratuita.

Para solucionar estos problemas, nació SEO. Sus siglas (Search Engine Optimization) traducidas al español significan Optimización de los Motores de Búsqueda. Esta disciplina (llamémosla disciplina de aquí en adelante) tiene la particularidad de que no está acoplada ni al desarrollo técnico ni a los negocios. Un analista SEO busca, a partir de estudios de mercado y de negocio, intentar volcar esta información hacia los conceptos que benefician a un sitio en la priorización dentro de un buscador. Vale la pena aclarar también que muchos de estos conceptos son difusos, creando una constante comunicación obligada entre profesionales y consultores con el fin de garantizar que la iniciativa SEO realizada/que está por realizarse no sufra penalizaciones o no vaya en contra de lo denominado “beneficiario para el usuario”, como sostienen líderes de proyectos en Google en [3]. Aquí uno de las características de la disciplina que más me llamó la atención y que me atrajo a aprender e investigar al respecto: la necesidad de tener tanto conocimientos técnicos como de negocios.

Como defienden diversos autores reconocidos de esta nueva disciplina ([4], [5]), no existe una estrategia marcada para encarar un proyecto “amigable” con respecto a SEO debido a que, por ejemplo, el contenido de un sitio de noticias va a ser muy diferente al de un portal de estudiantes, y por lo tanto los conceptos a aplicar también. Estos autores defienden que, ante objetivos diferentes, los puntos de vista deben ser muy diferentes también. Para entender mejor esta premisa vale la pena repetir que, con respecto a iniciativas SEO, sabemos el “**qué** hacer”, y no el “**cómo** hacerlo”. Qué hacer implica conceptos más técnicos, mientras que el cómo implica conceptos tanto técnicos como de negocio. Es por esto que es tan difícil sistematizar una estrategia consistente y reutilizable para el desarrollo orientado a SEO. Es difícil volcar aspectos del mercado y del negocio en una solución técnica. También, desde mi experiencia como analista SEO, puedo confirmar que hay muchos profesionales muy capaces pero con un perfil débil en tecnología. Aquí una clara ventaja (y un gran abanico de posibilidades con las que me encontré) que yo tuve al tener un perfil técnico de ingeniería.

Todas las dificultades previamente mencionadas se potencian bajo una premisa básica aplicable en muchos casos cotidianos: “**es más fácil basarse en palabras que en hechos**”. En nuestro caso puntual de SEO, estas “palabras” provienen de innumerables profesionales, “bloggers” y hasta compañías consultoras de SEO que no pueden probar con **hechos** y/o información a través del tiempo que cierta iniciativa o estrategia a llevar (obviamente, siempre relacionada con SEO) va a ser positiva para nuestro sitio. Como mencioné previamente, la información brindada por Google es escasa o nula (el único escrito oficial es el siguiente [6]); nadie conoce el algoritmo que utiliza Google para posicionar un sitio por encima de otro, por lo tanto deja una franja de prueba y error libre al analista SEO. De este modo encuentro un vacío en las pruebas concretas de que “cierta iniciativa funciona por encima de otra” que pienso aprovechar para la práctica de mi tesina. También es importante destacar que son muy pocas las empresas que tienen un seguimiento concreto de las posibilidades que sus sitios tienen con respecto a SEO. Son pocas las empresas (sólo las más importantes) que saben para qué palabra clave está optimizada cada una de sus páginas, que saben que están aprovechando al máximo todo contenido que poseen (esto último ocurre también en empresas grandes). Como analista SEO he tenido la posibilidad de participar en muchas demostraciones con consultores SEO (los más populares, [7] y [8]), en las cuales ninguno pudo demostrarme con hechos concretos comparaciones históricas de un antes y un después de una implementación, como mencioné previamente.

En conclusión, cualquier analista SEO sabe qué hacer, pero ante una estrategia fallida/victoriosa, son pocos los que pueden mostrar con facilidad datos históricos que demuestren que ESA iniciativa fue la causante de la victoria/derrota.

Mi idea es, a través de mi perfil tanto técnico como de negocios/marketing, crear e implementar una suite de herramientas que:

- **Proporcionen** un sistema base fácil de implementar por cualquier desarrollador que sirva para facilitar las decisiones estratégicas gerenciales a seguir
- **Analicen y encuentren** patrones favorables/desfavorables para alentar/evitar esfuerzos e iniciativas
- Faciliten la inteligencia de negocios por medio del análisis y la automatización
- **Proporcionen** una arquitectura modular que facilita la personalización (dependiendo de la línea de negocios) y la creación de reportes con resultados relevantes
- **Almacenen y ordenen** datos históricos de los esfuerzos realizados
- **Utilicen** los datos/información posibles más cercanos a la realidad
- **Automaticen** todos los puntos previamente mencionados de forma eficiente
- **Se relacionen** directamente con la información pertinente, mejorando la performance del sistema y las posibilidades de análisis y crecimiento

Hoy en día existen diferentes herramientas pagas ([7] y [8]) que, además de ser muy caras, no pertenecen en un 100% a la empresa que compra ese servicio (ejemplo, bases de datos con información histórica en servidores de las compañías que ofrecen el servicio, límite en los seguimientos de palabras clave, etc.) ni son 100% a medida de la misma. Quizás puedan ofrecernos el posicionamiento de una serie de palabras claves que nosotros previamente le solicitemos, pero eso no es suficiente (y menos aún moldeable en un 100% a las necesidades de la empresa). Estoy seguro que en la mayoría de los casos el cliente le va a reclamar al proveedor del servicio “excelente información, pero me gustaría ver esto otro...”.

El antecedente más cercano que desencadenó el desarrollo y la idea general de esta práctica remonta al 2009, y surgió luego de ver cómo la empresa donde trabajo hacía el seguimiento de los esfuerzos realizados. Se utilizaban dos archivos de Microsoft Access (que había que actualizar a mano) con información poco confiable, y ningún líder de proyecto ni manager podía acceder o actualizarlos con los diferentes esfuerzos implementados. Era un gran archivo con mucha información que simplemente no servía. En estos archivos se volcaban las posiciones de ciertas palabras claves mes a mes. Ese era el único seguimiento relacionado con SEO, algo de verdad catastrófico para una empresa internacional. Volcándolo a un tema cotidiano como el fútbol, yo lo compararía con saber cómo salió mi equipo el fin de semana, pero no saber quien jugó bien y mal, quién erró más pases, qué jugada o estrategia resultó, etc. En ese mismo instante muchas ideas de automatización y muchas oportunidades de proyectos surgieron, resultando en las causas suficientes para planear y finalmente iniciar la práctica de esta tesina.

Con respecto a mi experiencia previa, tuve la oportunidad de encarar un proyecto de rediseño de páginas ya creadas (proyecto finalizado). El objetivo de este rediseño es el de hacer las páginas más amigables hacia los buscadores. Fue una ventaja para mí trabajar al lado de analistas SEO (no sólo en la parte de desarrollo sino en etapas tempranas de análisis de requerimientos y del mercado) ya que de ellos aprendí diferentes maneras de enfrentar los problemas relacionados a esta disciplina, y diferentes trucos, estrategias o buenas prácticas para aquello.

Mi motivación y justificación para este proyecto, como sostenía previamente, es mi trabajo actual y mi puesto de analista SEO. Estoy dispuesto a desarrollar herramientas libres que cualquier empresa pueda utilizar no sólo para facilitar sus decisiones estratégicas futuras, sino para prevenir que se cometan errores dos veces y promover las decisiones basadas en hechos y no en meras palabras. En el marketing online se invierten grandes cantidades de dinero, y me parece inaceptable invertirlas sin ningún seguimiento y análisis previo detallado.

## II. Objetivos

Los objetivos que podría diferenciar son:

- Crear una suite de herramientas libres que sirvan para:
  - o Facilitar decisiones estratégicas
  - o Automatizar la generación de reportes
  - o Automatizar la obtención de información relevante relacionada tanto a sitios propios de la empresa como a sus competidores
  - o Automatizar el análisis de los posicionamientos en Google
- La suite estará compuesta por herramientas que:
  - o Guardarán información histórica relevante en una base de datos
  - o Obtendrán los meta tags [9] de un conjunto de URLs
  - o Obtendrán los rankings de un conjunto de palabras clave, como también los posicionamientos de competidores directos
  - o Generarán reportes automatizados, con una arquitectura modular
  - o Validarán si las URLs de mi dominio son visibles por Google

## III. Alcance y límites

Esta tesina estará orientada al buscador Google. Cualquier otro buscador está fuera del alcance. Hoy en día existen tres buscadores que lideran el mercado (obteniendo aproximadamente un 90% del mercado total). Ellos son Bing de Microsoft [10], Yahoo [11] (que ha sido comprado por Bing y cuyo motor de búsqueda está siendo fusionado con el de Microsoft) y Google, siendo Google no solo el más popular y utilizado sino el poseedor del algoritmo más evolucionado y avanzado. Esto es fácil de comprobar ya que los resultados de búsqueda de Google son los más cercanos a lo que el usuario está buscando. Las tres razones por las cuales el alcance de esta tesina está totalmente orientado a Google son:

- o El tráfico que ingresa a través de Google puede hasta quintuplicar el tráfico que ingresa a través de Bing y Yahoo juntos
- o El algoritmo de Google es el más evolucionado y avanzado, resultando también en el más demandante. Se estima que existen tres veces más factores de posicionamiento en Google que en sus competidores
- o Bing y Yahoo tienden a copiar los comportamientos del algoritmo de Google, por lo tanto es correcto concluir que una iniciativa para posicionarme mejor en Google también ayudará a mis posicionamientos tanto en Bing como en Yahoo

No incluirá el funcionamiento de los robots de búsqueda e indexación de Google, sino que estará 100% orientado a los resultados finales. No nos interesa cómo Google encuentra y guarda la información, sino dónde y cómo la va a mostrar en el futuro.

Todo lo referido a campañas de “link building” [12] estará fuera del alcance de la tesina, ya que estos conceptos implican más conocimientos de marketing que no son pertinentes a nuestra carrera.

En varias oportunidades utilizaré conceptos en inglés, sencillamente por que se utilizan esos nombres en el mercado. De todas formas, aclararé su significado en español (pero por comodidad, utilizaré el nombre apropiado).

## IV. Marco Teórico

En esta sección de la tesina es importante definir los conceptos básicos de SEO para posteriormente poder entender la práctica. De forma breve detallaré las diferentes iniciativas y aspectos a tener en cuenta para que un sitio esté lo mayormente optimizado posible, como también trataré de explicar qué es lo que hace que un sitio esté mejor posicionado que otro.

### ¿Qué hace Google?

Si tuviera que definir qué es lo que hace Google, diferenciaría 3 pasos básicos [13]:

- Crawling (“rastreo”) [14]
- Indexing (“indexación”) [15]
- Serving (“servicio”) [16]

Utilizaremos estos tres pasos para desarrollar tanto el marco teórico como el proyecto de tesina.

En el primer paso, el rastreo, se analiza toda la web a través de robots propietarios de Google. Un robot es un pequeño programa que navega a través de todos los sitios existentes recopilando e indexando absolutamente toda la información (contenidos, títulos, links, etc.). En los comienzos de Google sólo se indexaba texto, pero hoy en día se indexa contenido de imágenes, animaciones Flash, y otras tecnologías. Para entender esto es pertinente saber qué es indexar. Indexar refiere a la acción de registrar ordenadamente información para elaborar su índice [17].

Básicamente, tiene como propósito la elaboración de un índice que contenga la información de forma ordenada, con la finalidad de obtener resultados de forma sustancialmente más rápida y relevante al momento de realizar una búsqueda. Es por ello que la indexación es un elemento fundamental en cualquier motor de búsqueda, no sólo Google. Todo buscador debe tener eficiencia y velocidad. El poderío de procesamiento que se necesita para indexar toda la web y el poderío necesario para luego brindar los resultados de una búsqueda de forma rápida, son los dos pilares que hacen que Google no de información sobre sus algoritmos. Ellos son la clave de su éxito y los que lo diferencian del resto.

Entonces de esta forma, cuando se realiza una consulta (el usuario ingresa una palabra clave o un conjunto de palabras clave al buscador), Google se dirige al índice para localizar los elementos deseados, arrojando así resultados precisos y rápidos (etapa de servicio). Sin un índice el motor de búsqueda debería



escanear el contenido de cada página web de forma individual cada vez que se iniciara una búsqueda, lo cual considerando el volumen de la información existente en Internet, tardaría más que minutos y demandaría equipos informáticos muy potentes (en otras palabras, imposible e ineficiente). Teniendo los sitios indexados y ordenados es más fácil para Google la etapa de servicio, brindando entonces contenido relevante ante la búsqueda del usuario.

Sabiendo ya qué es lo que hace Google, podemos utilizar el sentido común para entender el trabajo de un analista SEO. Entonces, estos son los interrogantes que cualquier analista debe hacerse al comenzar su trabajo (partiendo desde la base). Vale la pena aclarar que un analista no debe “atacar” cada fase de forma separada, sino que es un constante trabajo en conjunto:

## Crawling

*Analista: “¿Sabe Google que existo? ¿Pudo acceder a mis sitios y encontrarnos?”*

Utilicemos el sentido común. Si Google no sabe que existimos, entonces todo el trabajo realizado es en vano. He visto muchos casos en los cuales se ha invertido en optimizar una página, cuando en verdad Google todavía no la reconocía.

Aunque la respuesta obvia a esta pregunta podría ser “imposible saberlo, solo Google lo sabe”, esa es una respuesta incorrecta. Hay una manera para indicarle a Google que existo. El propietario de un sitio puede suministrarle a Google una “lista” de las páginas que nos interesa que indexe, junto con alguna información adicional que haga la visita del robot más eficaz (ejemplo, frecuencia de la actualización de cada página). Esta “lista” se denomina “sitemap”.

Un sitemap [18] es un archivo XML que contiene toda esta información: todas mis páginas por las cuales quiero que los robots de Google pasen. El archivo debe alojarse en el servidor donde están alojadas mis páginas, aunque también es recomendado brindárselo directamente a Google a través de herramientas de webmaster, como “Google Webmaster Tool”.

Tener un sitemap ordenado, validado y correcto todavía no nos garantiza que existamos para Google, ya que hay otros factores tanto técnicos (por ejemplo, errores de HTML, tamaño excesivo de cada página, contenido duplicado, etc.) como de arquitectura (por ejemplo, páginas que sólo son enlazadas en el sitemap, páginas muertas, etc.) que son igual de importantes. Si Google no puede indexar nuestro sitio, es lo mismo que si no nos conociera... es más, quizás es peor ya que representa una incomodidad o problema que nosotros le estamos causando a los robots de Google, poniendo en riesgo nuestra reputación general para/con el buscador.

## Indexing

*Analista: “¿Puede indexar nuestro sitio?”*

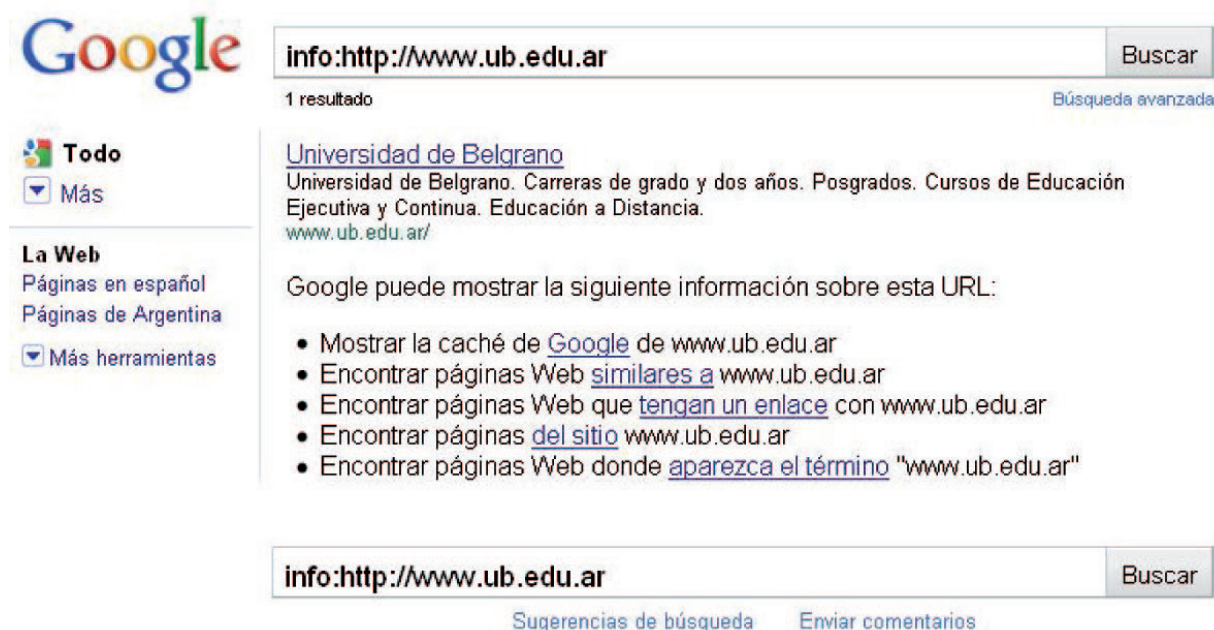
Con la pregunta de “¿sabe Google que existo?” respondida, ya debemos empezar a preocuparnos por aspectos técnicos del sitio. Recordemos que los robots de Google son programas, por lo cual su forma de analizar el sitio no es mágica, sino que validará paso a paso nuestro sitio. Si el sitio está corrupto técnicamente (problemas técnicos como etiquetas HTML mal cerradas, loops que nunca terminan, código JavaScript que no compila e infinidad de otros factores), los robots de Google jamás podrán recorrerlo por completo y, en la mayoría de los casos, terminarán por descartar la página entera. Mientras más limpio técnicamente este nuestro sitio, con más facilidad se va a indexar y menos problemas le traerá a los robots de Google. Hay ciertos errores que pueden ser divisados en la herramienta de Webmasters de Google, pero no todos.

Teniendo las dos preguntas previas contestadas y chequeados, es momento de saber si Google ya nos reconoce o no. La manera más sencilla es la de consultar al buscador directamente, realizando la siguiente búsqueda: “info:<URL>”, que incluye un parámetro especial “info”. Google brinda la posibilidad de añadir diferentes parámetros a las consultas [19] (por ejemplo, si buscamos “manual.java filetype:pdf”

los resultados de la búsqueda serán archivos en formato pdf). Básicamente le consulta a Google si “reconoce y/o tiene algún tipo de información sobre la URL en cuestión”, si tiene conocimiento de su existencia (ergo está indexada).

(Utilizaremos la URL <http://www.ub.edu.ar> para el ejemplo positivo, y la URL <http://www.urlInexistente.com.ar> para el ejemplo negativo)

### Ejemplo positivo



The screenshot shows a Google search for the URL `info:http://www.ub.edu.ar`. The search bar contains the text and a 'Buscar' button. Below the search bar, it indicates '1 resultado' and a link to 'Búsqueda avanzada'. The search results include a link to 'Universidad de Belgrano' with a brief description: 'Universidad de Belgrano. Carreras de grado y dos años. Posgrados. Cursos de Educación Ejecutiva y Continua. Educación a Distancia. [www.ub.edu.ar/](http://www.ub.edu.ar/)'. Below this, a message states: 'Google puede mostrar la siguiente información sobre esta URL:'. A list of suggestions follows:

- Mostrar la caché de [Google](#) de [www.ub.edu.ar](http://www.ub.edu.ar)
- Encontrar páginas Web [similares a](#) [www.ub.edu.ar](http://www.ub.edu.ar)
- Encontrar páginas Web que [tengan un enlace](#) con [www.ub.edu.ar](http://www.ub.edu.ar)
- Encontrar páginas [del sitio](#) [www.ub.edu.ar](http://www.ub.edu.ar)
- Encontrar páginas Web donde [aparezca el término](#) "[www.ub.edu.ar](http://www.ub.edu.ar)"

At the bottom of the search results, there is another search bar with the same URL and a 'Buscar' button, with links for 'Sugerencias de búsqueda' and 'Enviar comentarios'.

### Ejemplo negativo



The screenshot shows a Google search for the URL `info:http://www.urlInexistente.com`. The search bar contains the text and a 'Buscar' button. Below the search bar, it indicates 'Búsqueda avanzada'. The search results show a message: 'No hay información acerca de la dirección URL **www.urlInexistente.com**'. Below this, a list of suggestions follows:

- Si la dirección URL es válida, intente visitar dicha página haciendo clic en el siguiente enlace: [www.urlInexistente.com](http://www.urlInexistente.com)
- Encontrar páginas [del sitio](#) [www.urlInexistente.com](http://www.urlInexistente.com)
- Encontrar páginas Web donde [aparezca el término](#) "[www.urlInexistente.com](http://www.urlInexistente.com)"

Como se puede apreciar en las imágenes, si Google reconoce la URL nos brindará información básica de la misma. En caso contrario, nos indicará que no hay información sobre ella (`urlInexistente.com` no existe, por lo tanto no está indexada, por lo tanto Google no tiene información al respecto de la misma).

Ya logramos que Google nos reconociera, pero ahora debemos armar nuestra reputación... lo más difícil y lo que más trabajo necesita.

## Serving

### ***Analista: “¿Tengo contenido único y relevante a lo que busca el usuario?”***

En este momento Google ya sabe que existimos. Ahora, ¿para qué queremos que Google sepa que existimos? Obviamente para que, ante la búsqueda de un usuario, nuestros sitios aparezcan en los resultados, llevando a que los usuarios ingresen, aumente el tráfico de mi sitio, incremente la posibilidad de que esa visita convierta, etc.

Ninguno de los pasos anteriores es relevante si no sabemos para qué tipo de palabras claves (que de ahora en más llamaremos “keywords”) podemos intentar posicionarnos. A lo que me refiero es, por ejemplo, todos quisiéramos posicionarnos primeros en los resultados de la keyword “las vegas Hotels” ya que tiene un promedio de 2 millones de búsquedas mensuales pero, si nuestro sitio es un sitio de noticias locales, el contenido ofrecido no es relevante a la keyword. Entonces aquí el primer pilar de SEO: “el contenido debe ser relevante y único” [20]. “Relevante” porque el usuario quiere obtener lo que buscó, y sólo le interesa lo que buscó, y “único” por que el usuario no quiere ver contenidos duplicados. He aquí la importancia de conocer el mercado (qué busca la gente, relacionado con mi línea de negocios), de conocer el volumen global de una keyword (llamemos “volumen global” a la cantidad de veces que esa keyword es buscada en Google ya sea mensual o semanalmente) y de conocer nuestras posibilidades de posicionarnos en la misma. Como se verá más adelante en la práctica, la suite de herramientas que voy a desarrollar va a ser de ayuda en estas decisiones estratégicas, como también un punto clave para rastrear nuestra performance.

Es importante tener en claro la importancia del contenido. Google intenta darle la mejor y más eficiente experiencia al usuario. Esto se va a cumplir siempre y cuando el usuario haya encontrado lo más semejante a lo que buscó. Si los resultados obtenidos por el usuario no son los que él esperaba, el culpable es Google, por lo tanto Google debe asegurar que el contenido mostrado en los resultados sea lo que el usuario quiere. Esta es la única y sencilla explicación, que es la causante del secreto y complejo algoritmo de indexación y búsqueda de Google.

Es correcto resumir esta idea con la frase “content is king” (el contenido es el rey) [21]. He conocido innumerables casos en los cuales una empresa ha querido posicionarse alto en una keyword únicamente por que tiene alto volumen, pero sin tener contenido relevante a esa keyword... y todos esos casos han sido casos fallidos, sin excepciones. De nada sirve intentarlo si no se tiene contenido relevante.

A esta altura, se debe tener en claro la importancia del contenido. Ahora supongamos que una página de nuestro sitio tiene buen contenido (relevante y único). Esto de ninguna manera nos garantiza posicionarnos ni siquiera en las primeras páginas, debido a que existen los competidores. Imagínense alrededor de cien empresas diferentes focalizadas en la keyword “vuelos baratos”... un contenido óptimo no alcanza para posicionarnos alto. Aquí entran en juego otras iniciativas/estrategias SEO. Desde optimizaciones a realizar en nuestros sitios, hasta factores que indican “cuán popular es nuestro sitio en Internet”, o “cuán popular es nuestro sitio con respecto a los demás”. Haremos un repaso de los esfuerzos más pertinentes, conocidos y funcionales, pero es pertinente aclarar que llevar a cabo esto a rajatabla puede llevar a cambiar la arquitectura de nuestros sitios (con resultados positivos) por completo. En muchos casos se decide por crear páginas enteramente nuevas, que estén 100% optimizadas en SEO (optimizadas en SEO también significa optimizadas para el usuario).

Los esfuerzos/estrategias SEO más importantes son:

- Meta Tags y otros tags HTML
- URLs limpias
- Cross Linking
- Backlinking
- Redireccionamiento permanente (redireccionamiento 301)
- Contenido dinámico

## Meta Tags

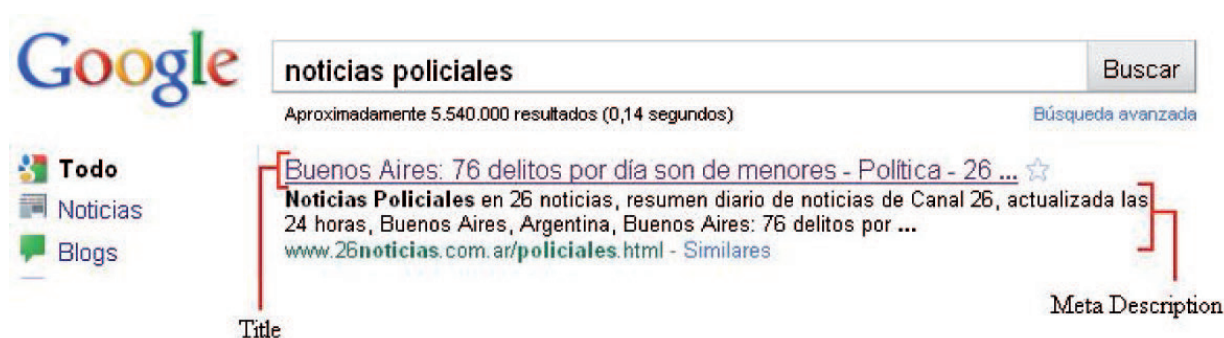
Los meta tags son caracteres pertenecientes al HTML que se deben escribir dentro del tag general <head> y que podemos definir como líneas de código que indican a los buscadores que le indexan **por qué términos debe ser encontrada la página**. Es la iniciativa más antigua y conocida de todas [22] [23].

Entre los meta tags se encuentra el “title” [24], cuyo objetivo es brindarle un título al buscador que resuma en 80 caracteres o menos lo que el usuario va a ver en la página. Entre los errores más comunes a cometer en este tipo de meta tags existe el sobrepasar los 80 caracteres (de esa forma, Google directamente obvia su indexado) o, el error más común, el poner títulos que no sean **descriptivos, relevantes** con lo que estoy ofreciendo ni ayuden al usuario. Seguiremos insistiendo con la **relevancia** de conceptos y el beneficio al **usuario**. Internet es una red inmensa, si no se es directo y relevante con lo que se quiere ofrecer, Google se da cuenta de esto y deja de lado el sitio. Es importante aclarar que cualquier meta tag debe ser sintácticamente correcto, ya que Google también reconoce errores sintácticos. Es recomendable incluir la/s keyword/s para la/s cual/es estoy optimizando el sitio en los meta tags.

Otro de los meta tags es el meta description, cuyo objetivo es brindarle una descripción breve al buscador que resuma las intenciones del sitio. También existen limitaciones de caracteres.

Por último nos encontramos con los meta keywords, o palabras clave, que en su momento eran muy importantes. Básicamente son un conjunto de palabras (o conjunto de frases) que le indican al buscador para qué parámetros de búsqueda es más adecuada mi página. Por ejemplo, si quiero que mi sitio sea buscado por flora y fauna, los meta keywords podrían ser: “flora, fauna, flora y fauna”. He aquí la fuente de errores más común en todo sitio Web mal encarado: llenar los meta keywords de muchas palabras clave pensando que nos hará bien, como por ejemplo: “flora, fauna, flora y fauna, mejor, top, barato, primero, mas flora, mas fauna” y etc. Un punto importante a destacar es que mientras MENOS keywords ingresemos, MÁS relevancia les dará el buscador [25]. Si yo ingreso 2 keywords, van a ser mucho más relevantes y mejor calificados que si ingreso 10 keywords (por más que la probabilidad de que alguno de ellos sea ingresado como parámetro de búsqueda aumente). Se puede comparar como “ingresar la respuesta concisa y directa en un examen” o “escribir todo lo posible, ya que en algún momento la respuesta va a aparecer”. Los buscadores detectan fácilmente esto. Buscan consistencia y relevancia; mientras más fácil sea diferenciar el objetivo de nuestro sitio, mejor para todos.

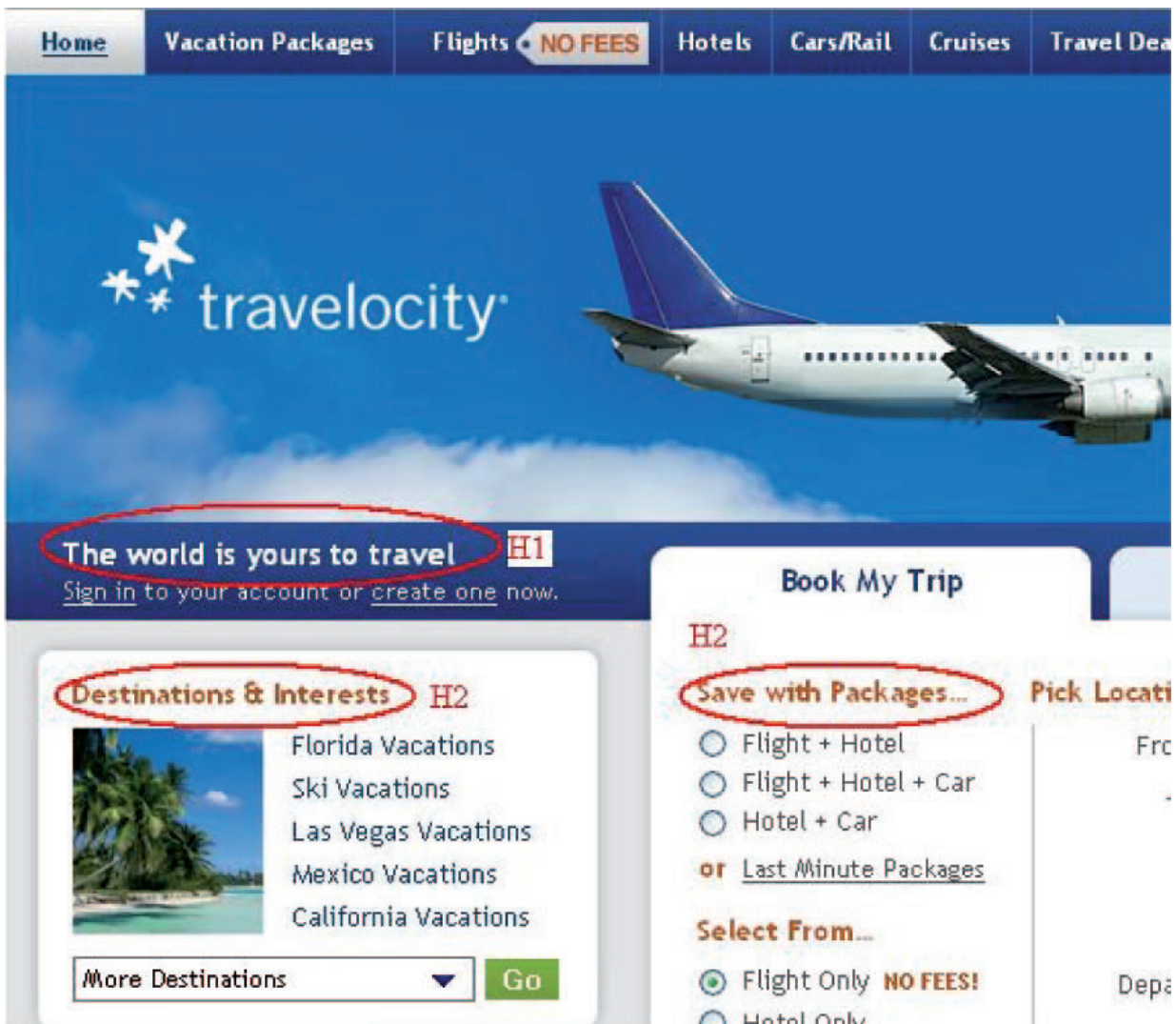
De todos modos, la importancia de estos meta tags (para los buscadores) disminuye con los años, ya que los sitios medianamente optimizados los tienen bien implementados. Como establecen [26] [27], concuerdo que son el punto de partida, pero no es un factor desequilibrante cuando nuestra competencia también tiene sus meta tags optimizados. Es un esfuerzo que vale la pena hacer debido a su facilidad, y a que además se mostrará en los resultados de Google, siendo entonces muy importantes para el usuario:



Podemos confirmar entonces que la descripción y el título no sólo son importantes para SEO, sino para atraer al usuario a acceder a nuestro sitio (tráfico). Más aún si estoy posicionado entre los primeros 5 resultados, en este caso un título y una descripción correctos son determinantes.

Otro tipo de tags HTML importantes son los tags h1, h2, h3, etc [28]. Los tags h1 y h2 se comparan con el título y el copete de una nota periodística. En otras palabras, son títulos de menor tamaño pero

igual con alta importancia. Mientras más factores tengamos a nuestro favor, habrá más probabilidades de posicionarnos mejor que nuestros competidores.



Un tag muy importante es la URL canónica. [29] La URL canónica es una etiqueta que se coloca en cualquier URL que pueda conllevar problemas de duplicidad de contenido con otra del mismo dominio. Utilizando esta etiqueta en todas y cada una de las páginas “parecidas”, le estaremos diciendo a Google que no se trata de una copia, sino de una variación debida a factores específicos, como en el caso de una versión para imprimir de la página, una consulta de búsqueda, una vista por categorías, etc. En muchos casos vemos a sitios duplicados indexados, debido a que en nuestro servidor de alguna manera se creó una página idéntica a otra, pero con algún parámetro de sesión extra en la URL. Si no se indica una URL canónica en este caso, el sitio puede llegar a ser penalizado por Google debido a contenido duplicado.

## URLs limpias

Un factor muy importante, y concuerdo con [30] que es más importante que los meta tags, es la URL del sitio en sí [31]. Primero y principal: incluir la keyword en la URL es, después del contenido, el paso principal para indicarle a Google que ese sitio en particular es relevante con respecto a esa keyword. Por ejemplo, podemos fácilmente entender que <http://www.loteria.gba.gov.ar> va a ser un competidor duro en la keyword “lotería”, ya que está en su URL.

Luego de entender este concepto, podemos empezar a hablar sobre “URLs extensas” y “URLs limpias”. Es adecuado explicar este concepto con ejemplos:

URL extensa: <http://www.algunsitio.com/index.php?variable1=valor1&variable2=valor2>

URL limpia: <http://www.otrositio.com/categoria-nombre-del-producto>

Veremos las URLs extensas principalmente en sitios dinámicos. Una URL extensa lleva a problemas con el indexado (si la URL es muy larga y tiene muchos parámetros, el robot de Google puede no indexarla), problemas en la confianza del usuario (y en el entendimiento) y menos relevancia de keywords en la URL, entre otros. Entonces, una URL limpia no sólo es amigable para el usuario (siendo descriptiva), sino para cualquier robot de cualquier buscador. Un ejemplo claro de un sitio cuya arquitectura de URLs es totalmente limpia es [www.mercadolibre.com.ar](http://www.mercadolibre.com.ar). Veamos unos ejemplos:

<http://listado.mercadolibre.com.ar/zapatillas-adidas>

<http://listado.mercadolibre.com.ar/zapatillas-nike>

<http://articulo.mercadolibre.com.ar/MLA-92002975-zapatillas-gola-2010-originales-nuevas-en-caja-entrego-ya- JM>

<http://articulo.mercadolibre.com.ar/MLA-94124943-capezio-freedom-free1oferta-liquidacion-zapatilla-danza- JM>

En los dos últimos ejemplos vemos que existen algunos parámetros como código de artículo (a veces es técnicamente imposible obviarlos), pero siempre manteniendo la limpieza y fluidez necesaria, con alta densidad de keywords pero sin perder el sentido. Vemos también como las palabras están divididas por “-“. Google reconoce las palabras separadas por guiones o puntos como “palabras de mismo peso, con la misma importancia”. Esto significa que si, por ejemplo, usáramos “/” en vez de guiones, Google le daría menos importancia a las palabras siguientes.

Ejemplo: <http://www.camperas.com/cuero/americanas/camperas-baratas>

Esta disposición todavía es correcta y amigable tanto al usuario como a SEO, pero el peso tanto de “americanas” como “camperas” y “baratas” va a ser menor. Insisto que esta igualmente es una buena manera de implementarlo, más aún si los sitios <http://www.camperas.com/cuero> y <http://www.camperas.com/cuero/americanas> existen. Ambas implementaciones son correctas, tanto las URLs 100% limpias, como las URLs en forma de directorios, pero digamos que es interesante que se pueda tener acceso a los contenidos en 2 o 3 clics. Una arquitectura con muchos más niveles es mala para el SEO. Y finalmente, si la keyword está en el nombre del dominio, la relevancia será mucho mayor para Google, teniendo una gran ventaja sobre los demás. Digamos que si una keyword está en el dominio, si se hace un buen trabajo se puede garantizar un posicionamiento entre los primeros 5.

Con respecto a las URLs dinámicas, hoy en día existen soluciones. Escribiendo un archivo llamado .htaccess se pueden crear direcciones amigables generadas automáticamente.

## Links

En esta sección del marco teórico damos paso a un factor clave en el mundo SEO: los links (enlaces) [32].

Antes de explicar sus beneficios, es pertinente dar una buena introducción a la importancia de los links en general. Para empezar desde lo global, la palabra “Internet” podría definirse como un conjunto descentralizado de redes de comunicación interconectadas. La palabra “interconectada” conlleva a la existencia de enlaces, o en inglés links. Google rige sus decisiones/algoritmos utilizando parte de la definición de Internet: si el sitio está “interconectado”, si el sitio tiene tanto links internos (links entre sus páginas, links entre sus dominios) como externos (links de otros dominios hacia páginas propias), entonces el sitio es popular, relevante e importante. Yendo al otro extremo, si mis páginas son “independientes”, nadie las enlaza ni ellas se enlazan entre sí... entonces su popularidad es baja. Veámoslo también desde el punto de vista del sentido común. Un sitio bien enlazado nos brindará información más relevante (ya que el sitio estará bien dividido en secciones/categorías, en diferentes páginas), cumplirá con nuestras necesidades de mejor manera, mantendrá al usuario más tiempo en el mismo, será más fácil de navegar. Además, si el sitio es enlazado por otros, esto significa que el sitio es popular... que alguien vio cierto potencial en el sitio.

Dejando esto en claro, ahora podemos definir que es un “cross link”. Un cross link es básicamente un link entre páginas propias, del mismo dominio [33]. En otras palabras un link interno. Sin abusar de ellos como dije previamente es importante entender que además de dar beneficios SEO, también mantendrán por mayor tiempo a los usuarios en el sitio. Con respecto a la arquitectura del sitio, la base arquitectónica de un sitio bien optimizado con relación a SEO son los links internos. De todos modos, se estima que no es del agrado de los robots de Google que una página tenga más de 100 links. Debido a esto, entre otras cosas, se creo el atributo “nofollow”.

Básicamente, “nofollow” es un atributo en código HTML que se le pone a un link cuando queremos que éste no sea tenido en cuenta por los buscadores [34]. De esta manera, Google y otros buscadores no dan valor a ese enlace. Entonces, de esta manera se puede mantener una buena experiencia para el usuario sin dejar de lado SEO. Un buen ejemplo podría ser el de un link que me direcciona a una página de autenticación para los miembros registrados. Este link no tiene ningún peso SEO, ya que está dispuesto en la página solo para beneficio del usuario. Un ejemplo de un link que no debería tener “nofollow” es un link a la página de “espectáculos” en el sitio de un diario online. Es un link importante, ya que le estoy dando popularidad de links a ese sitio. Esto aplica también a los links que salen de nuestra página hacia páginas de otros dominios. Tranquilamente puede agregar un “nofollow”.

Otro atributo importante de los links tanto internos como externos es el “anchor text”, o en español “texto ancla” [35]. El anchor text es el texto en el cual se clickea para acceder al link, el texto que se muestra en el enlace. Hay muchos casos en los cuales este texto es simplemente un “click aquí”, algo que es erróneo no sólo para SEO sino para la experiencia del usuario. El punto más importante es que si hay una keyword en el anchor text, entonces ese link le dará más popularidad al sitio destino con respecto a esa keyword. Volvamos al ejemplo de un diario online. Si tengo un link en la sección de “policiales” que direcciona a la sección de “espectáculos” cuyo anchor text es “click aquí”, no se está aprovechando al máximo el beneficio del link (o como se conoce popularmente, “link juice” [36]). En cambio, si el anchor text es “espectáculos”, entonces Google le dará más relevancia a esa página con respecto a la keyword “espectáculos”. Esto aplica tanto a links internos como externos.

Ahora, con la cantidad de sitios que hay en Internet Google tuvo que buscar una manera de poder ordenar los links por importancia. Tuvo que buscar una manera de que un link que está en una página importante tenga más valor que un link de una página menos popular. Si un link proveniente de yahoo.com tiene el mismo valor que un link proveniente de un blog con un día de antigüedad, sería muy fácil obtener una ventaja a través del fraude. Para esto, Google inventó el “PageRank” [37].

PageRank es una marca registrada y patentada por Google el 9 de enero de 1999 que ampara una familia de algoritmos utilizados para asignar de forma numérica (valores del 1 al 10) la relevancia de los documentos (o páginas web) indexados por un motor de búsqueda. Dicho valor es calculado por Google empleando algoritmos especiales basados en la valoración cualitativa y cuantitativa de los enlaces externos de cada página. Entonces, de esta manera puedo confirmar que se pueden ordenar la importancia de los links por “PageRank del sitio origen”. Páginas como Facebook y Twitter tienen PageRank de 7 y el único sitio con PageRank de 10 es, obviamente, Google.

Este aspecto es muy importante tanto en links internos como externos. Con respecto a links internos, lo más probable es que la página con mayor PageRank es mi sitio principal, o homepage. Los links que tenga mi homepage hacia otros sitios internos propios van a ser de alta importancia (mucho más si tienen anchor texts optimizados para la keyword deseada). Con respecto a links externos, puedo confirmar que un link externo con anchor text optimizado y de PageRank 7 tiene más peso que 10 links de un sitio con PageRank 3 y anchor text erróneos.

Lo último a tener en cuenta (pero muy importante) es la cantidad de dominios diferentes que poseen enlaces hacia mi página. Es preferible tener links de dominios variados que muchos links de un mismo dominio. Mientras más variedad de dominios haya, la lógica nos dice que seremos más populares alrededor de Internet. Una excelente estrategia de ejemplo es la de sitios “socios”. Un sitio socio puede ser un sitio de la misma empresa, pero con diferente dominio (por ejemplo, miSitio.com con miSitio.com.ar y miSitio.com.uk). Si se logra armar una buena arquitectura de links entre estos sitios, los beneficios entre ellos se multiplicarán. Hay muchas URLs posicionadas #1 en keywords importantes que aumentaron su posición y su cantidad de links solo gracias a esta estrategia.

**En conclusión, si mi página tiene contenido relevante a la keyword ‘x’, su URL está optimizada y limpia, sus meta tags, H tags optimizadas para esa keyword, tiene un alto número de links con anchor text ‘x’, tanto internos como externos... entonces estar entre las primeras posiciones es una garantía.**

Todas las iniciativas que vimos son posibles hacerlas “onsite” (dentro de mi sitio), menos la iniciativa de links externos. Este es un tema muy complejo que no es pertinente para la tesina, pero es correcto nombrar sus categorías:

- “Social media” (medios de comunicación social): ser populares en las redes sociales [38]
- “Article marketing” (marketing de artículos): tener una buena fuente de artículos con links optimizados y direccionando a nuestro sitio [39]
- Blogs
- Foros
- Otros

Esto involucra una inversión de capital en contratar escritores, hacer campañas de marketing online, hacer campañas de compra de links en directorios de PageRank alto, hacer sociedades con bloggers populares y hacer campañas de marketing en redes sociales, entre otros. Es tan complejo el tema, que hay especialistas SEO que solo se dedican al “link building” (construcción de links).

### **Redireccionamiento permanente**

Cuando cambiamos de ubicación alguna página en nuestro sitio los accesos a ella desde otras páginas que la enlazan suelen perderse, además de que quienes nos visiten desde un buscador como Google no van a encontrar la página. Lo que tenemos que hacer es tratar de configurar la página para que cuando llegue el robot del buscador sepa que la página se ha cambiado de ubicación de forma definitiva, para ello vamos a utilizar la redirección 301 [40].

Cabe resaltar que este estado traspasa los valores de PageRank de Google y del número de páginas que la enlazan. La redirección 301 consiste en un estado de los estándares del protocolo HTTP.

### **Contenido dinámico**

Por último, veamos un concepto importante con respecto al contenido. Habíamos confirmado que el contenido debe ser relevante y único. Con el pasar de los años la información viaja más rápido, la vida entera se vuelve más dinámica. Por lo tanto, el algoritmo de Google no es la excepción. Si nuestro contenido es dinámico (cambiante a través del tiempo), los robots de Google se darán cuenta de esto y re-indexarán nuestra página con más frecuencia (resultando en una mayor probabilidad de que Google encuentre más páginas de mi dominio) [41].

Una buena fuente de contenido dinámico pueden ser los usuarios finales y sus comentarios, como opiniones sobre productos, artículos, etc. Si el sitio es un sitio que se mantiene activo durante un largo período y encima con contenido dinámico, entonces la lógica nos dice que estamos haciendo bien el trabajo.

Para finalizar el marco teórico, quiero resaltar lo difícil que es llevar a cabo todas estas iniciativas/estratégicas sin un plan estratégico. Para garantizar un buen desempeño en SEO, es necesario tener un buen conjunto de normas y buenas prácticas que toda la empresa y todos los proyectos de la empresa deben respetar. Desde un formato estándar para las URLs limpias, hasta un formato estándar y a su vez dinámico para módulos de links internos.

La experiencia personal me enseñó que SEO es algo que necesita de la cooperación de todos los equipos de una empresa, tanto técnicos como de negocios. Puedo optimizar mi sector/página al máximo, pero necesito de la relación con otras páginas de mi sitio y otras páginas externas. Después de todo, esto es lo que Google quiere: que sigamos conectados, que nos sigamos enlazando, que sigamos brindándole contenido relevante y único al usuario.



La única manera de que el usuario siga utilizando Google, es que los resultados de sus búsquedas cumplan con sus expectativas. Hasta el día de hoy lo siguen logrando, y hasta el día de hoy esas expectativas siguen cambiando, dando lugar a la última premisa relacionada con SEO: SEO está siempre en constante cambio, por lo tanto no es algo en lo que se puede trabajar y dejar de un día para el otro [42]. Hoy podemos posicionarnos primeros pero mañana en la tercera página. Con el auge de las redes sociales, los parámetros cada vez van cambiando más y más, obligando a los analistas a estar atentos a cualquier patrón que movilice los posicionamientos. Nuestro trabajo como analistas no es sólo el de optimizar, sino el de investigar y razonar sobre lo que Google está ideando y analizando hoy en día, y lo que puede llegar a estar analizando en el futuro. Lograr esto puede significar ganancias o pérdidas millonarias de tráfico.

Obviamente es imposible predecir este tipo de cosas, pero si hay algo que Google siempre cumple es que se rige por el sentido común de las cosas, y siempre favoreciendo al que sigue las buenas prácticas y al que se preocupa por crear un buen contenido para el usuario final.

## V. Propuesta de desarrollo de la tesina

Con el fin de cumplir con los objetivos planteados, mi propuesta para el desarrollo de la tesina consiste en crear una serie de herramientas libres que sirvan de ayuda en los 3 pilares de Google (Crawling, Indexing, Serving). Hoy en día no existen herramientas de código abierto que ayuden en este sentido, ni tampoco herramientas fáciles de modelar y moldear a las diferentes necesidades de los diferentes usuarios/emprendimientos. He tenido la posibilidad de trabajar en una empresa grande, donde he utilizado muchas aplicaciones y he estado presente en muchas demostraciones de diferentes compañías ofreciendo sus herramientas de SEO... pero ninguna me pareció moldeable ni escalable (sin mencionar el elevado precio). Otro punto en contra de estas herramientas es que toda la información sensible está almacenada en servidores FUERA de la empresa, algo peligroso en caso de querer terminar con el contrato.

Como ejemplo de este tipo de herramientas, puedo nombrar a la más popular: Searchlight. Hoy en día empresas muy prestigiosas (como por ejemplo, la norteamericana “Toy’s R Us”) trabajan con ellos. Basan su negocio en tres pilares principales: seguimiento del posicionamiento, reporte y datos históricos. Al iniciar contrato, nosotros le entregamos una lista de pares keyword\_URL para las cuales deseamos saber nuestro posicionamiento, y una lista de nuestros competidores más cercanos. Aquí el primer riesgo: estamos entregando información sensible de nuestro plan estratégico a terceros. No solo eso, sino que los datos históricos radicarán en sus servidores... significando que, en caso de que quisiéramos romper contrato con ellos, toda esa información no estará a nuestro alcance. Realizan su trabajo de forma correcta, pero el precio a pagar no es justificable ya que la información que ellos recopilan es pública de Google... sólo hay que tener las herramientas para conseguirla. Con respecto a reporte, ofrecen los reportes básicos (como, por ejemplo, keywords posicionadas en cada SERP y las oportunidades de crecimiento). Como lo establecí anteriormente, es imposible conseguir un reporte detallado de lo que uno verdaderamente necesita para su negocio. Sólo teniendo una aplicación propia, de manejo propio, es posible detallar y analizar la información que uno verdaderamente necesita.

Teniendo en cuenta lo estipulado, me plantearé la siguiente hipótesis para orientar mi desarrollo:

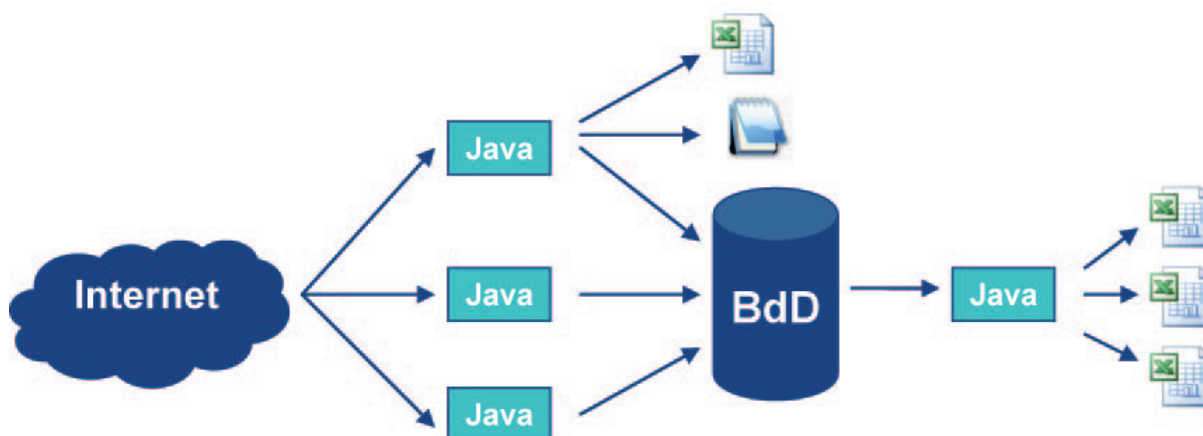
*“Es posible desarrollar un conjunto de herramientas de software libre de gran personalización y con alto grado de entrega de información para toma de decisiones en el área SEO”.*

La idea del desarrollo será la de crear aplicaciones Java que obtengan información, analicen, ordenen, procesen y finalmente se comuniquen con una base de datos MySQL (de esta manera, guardaremos datos históricos en servidores locales). Luego, facilitaremos la generación de reportes a través de otra aplicación Java. También, algunos de estos procesos deberán poder ser ejecutados sin la necesidad de persistir la información que obtengan en una base de datos (o sea, que automáticamente genere un reporte de lo analizado/buscado/obtenido), dando lugar también a reportes rápidos. Como tercera opción de output también es conveniente implementar la salida a un archivo de texto plano en forma de sentencias “Insert” de MySQL, en caso de que no queramos que se ingrese la información en la base de datos directamente.

Decidí inclinarme por MySQL ya que, además de ser libre, es un lenguaje que pude conocer en la facultad. Las bases las obtuve de diferentes materias que, a pesar de que tuve que investigar e informarme sobre MySQL avanzado, me brindaron los cimientos suficientes para tener confianza y animarme a trabajar con este lenguaje.

Un caso similar ocurre con Java, y principalmente con el paradigma orientado a objetos. Poder modelar teniendo este paradigma presente me permitió que el nivel de abstracción al codificar fuera lo suficientemente cercano al modelo real, dando lugar a un mejor modelado general y a un mejor manejo de errores. Además, Java tiene la ventaja de poseer diferentes APIs que facilitaron mi trabajo y evitaron que tenga que realizar arduos métodos (de conexión de sockets, por ejemplo) que me hubieran llevado más tiempo de lo estipulado.

A continuación, el modelo básico a muy alto nivel de todo el sistema:



### Aplicaciones a desarrollar

Utilizando el modelo Crawling – Indexing – Serving de guía, en esta sección del marco teórico pasaré a detallar los procesos a desarrollar.

Antes que nada es importante crear una base sólida no sólo para nuestra información sensible sino para los datos históricos que nos van a servir, como mencionamos previamente, a analizar los resultados de los esfuerzos realizados como también a nuestros competidores. Para aquello, modelaré e implementaré una base de datos MySQL. Para esto utilizaré herramientas libres nativas de MySQL. Como servidor utilizaré MySQL Server 5.1, para el modelado utilizaré MySQL Workbench 5.1 y para administrador de consultas y backups utilizaré una suite de herramientas MySQL GUI Tools 5.0.

Esta base de datos será la madre de la información que será obtenida/persistida con nuestra suite de herramientas.

Recordemos que el punto de partida para posicionarnos es que Google sepa que existimos, que Google pueda ver nuestros sitios. Recordemos también que la forma correcta y precisa de realizar eso es a través de una consulta avanzada a Google (“info:<URL>”). Ahora, un sitio puede llegar a tener millones de URLs, lo que hace imposible la ejecución manual de esta consulta. Por lo tanto, automatizaremos la ejecución de la misma con un proceso que llamaremos “Indexed Or Not (ION)”.

### Proceso “Indexed Or Not (ION)”

Para realizar estas consultas, implementaremos un sistema que se comunique y actúe de mediador con la API de búsqueda de Google. Una interfaz de programación de aplicaciones o API (del inglés application programming interface) es el conjunto de funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de

abstracción, generalmente (aunque no necesariamente) entre los niveles o capas inferiores y los superiores del software [46]. Su nombre oficial es “Google AJAX Search API”. Su objetivo es, básicamente, el de permitir a sitios terceros crear una interfaz en AJAX que consulte a Google. En nuestro caso, no nos interesa la interfaz gráfica sino los resultados. Por lo tanto, utilizaremos el poder de la API para eso. Ahora, hay que tener en cuenta que la API está preparada para que un usuario humano la utilice. Si realizamos consultas automatizadas de manera obsesiva, Google nos excluirá de este servicio gratuito. Debido a esto, deberemos implementar cierta lógica para intentar emular a un usuario humano lo más cercano posible siendo de esta manera, “Search Engine friendly” (como lo define la documentación de la API, “amigable hacia el buscador”) [43].

Esta es una herramienta que jamás encontré, y es algo crucial para cualquier campaña SEO. Hice un relevamiento sobre cuan interesados estarían los expertos SEO en adquirir esta herramienta, y los resultados me sorprendieron. Recibí muchas respuestas positivas, con mucho interés por ver un modelo terminado. También relevé a las diferentes empresas que ofrecen aplicaciones SEO. Ninguna lo implementó, y todas tomaron notas sobre el asunto estando más que interesadas en investigar al respecto (resultando, por ejemplo, en ofertas monetarias por el futuro software).

Otro aspecto importante es el de saber cómo estamos posicionados en Google con respecto a nuestras keywords. No sólo eso, sino también saber quiénes son y cómo están posicionados nuestros competidores más cercanos. Para satisfacer esta necesidad, será necesario implementar una herramienta que valide los posicionamientos en Google.

### **Proceso “Rank Checker (RC)”**

Utilizando la API de Google, esta vez haciendo búsquedas normales, haré una herramienta que permita verificar si algún sitio de mi dominio está posicionándose en Google con respecto a una serie de keywords. También será interesante obtener los primeros 5 competidores para cada keyword, como también algún otro competidor que yo quiera rastrear (por ejemplo, competidores conocidos).

Esta aplicación también me permitirá ir definiendo las diferentes entidades de mi base de datos. Necesitaremos, por ejemplo, persistir las entidades keyword, URL, la relación keyword\_URL, dominio, ranking, entre otras. Será importante diagramar bien la base de datos, ya que en este proceso se debería incluir validaciones en cada entidad (ejemplo: si hay una URL que posiciona en cierta keyword y esa URL no está en mi base de datos, deberé agregarla... no sin antes agregar el dominio de esa URL en caso de que no esté).

Después de obtener los posicionamientos tanto propios como de mis competidores, es importante empezar a analizar las URLs (tanto propias como las de los mejores posicionados), para saber el nivel de optimización que cada una tiene y, de esta manera, ver cuáles son mis oportunidades de mejora.

### **Proceso “Meta Tag Puller (MTP)”**

Utilizando librerías nativas de Java (java.net) y librerías de apache (commons), ambas libres, buscaré una manera de descargar el código fuente de cada URL y obtener el título de la página, meta keywords, meta description, los tags H1, H2, H3, H4, la cantidad de links que tiene la página y la URL canónica de la misma.

El proceso también deberá validar si el sitio está en funcionamiento (ya que puede ocurrir que el sitio este fuera de servicio). Deberá, de cierta forma, soportar todos los errores de protocolo http para que no falle en medio de su ejecución.

Siguiendo el análisis SEO, el siguiente paso sería el de analizar la cantidad de links que van hacia las URLs, no sólo las propias sino las de los principales competidores en cada keyword. De esta manera, sabremos cuántos links necesitamos para alcanzarlos.

### **Proceso “Yahoo Site Explorer Links (YSE)”**

En este caso, utilizaré la API de Yahoo para obtener los links que van hacia cada URL. Como vimos anteriormente, es vital para nuestra estrategia SEO saber quienes nos están enlazando, como también la cantidad de links que nos enlazan (y la comparación con nuestros competidores). Yahoo posee una API llamada Yahoo Site Explorer (gratuita) que permite obtener la cantidad de links que van hacia una URL. Vale la pena aclarar que la manera más exacta de obtener los links hacia un sitio es a través de la API de Yahoo Site Explorer, no a través de Google. Esto puede sonar ilógico ya que trabajamos para posicionarnos en Google, pero esto está comprobado y todos los expertos en SEO lo hacen [44]. En nuestro caso entonces, no tendremos que utilizar directamente YSE, sino que automatizaremos las consultas a la herramienta con el proceso.

El proceso también tendrá su versión para obtener estos links, de manera de poder analizarlos. De esta forma, podremos saber exactamente qué links tienen nuestros competidores, para poder intentar igualarlos y superarlos.

La limitación de esta segunda parte de la herramienta es que solo me dará los links, no sus anchor texts. Como vimos en el marco teórico, de nada sirve un link sin un anchor text acorde a la keyword. Será pertinente entonces crear un proceso para obtener estos anchor texts.

### **Proceso “Link Checker (LC)”**

Este proceso utilizará el mismo diseño que el MTP, pero con la diferencia que obtendrá los tags <a> (links en HTML) que obtengan el nombre de mi dominio. De esta manera, podré analizar URL a URL si existe alguna llamada a mis sitios. En caso de que esto ocurra, obtendré ese link junto con su anchor text.

También vale la pena destacar la aplicación de este proceso en el monitoreo de campañas de link building. Si realizamos alguna campaña de forma terciarizada, o si adquirimos algún link por intercambio o compra, será importante validar que nuestros links existan verdaderamente en estas páginas. En caso contrario, los proveedores podrían estar estafándonos realmente. Con este proceso podremos validar que los artículos comprados, links comprados posean verdaderamente referencias a nuestra página.

El último paso será el de obtener reportes. La manera más fácil y exacta de obtener reportes es la de consultar directamente a la base de datos. Pero también va a ser eficiente el crear un proceso que genere reportes más avanzados de forma automática.

### **Proceso “Report Generator (RG)”**

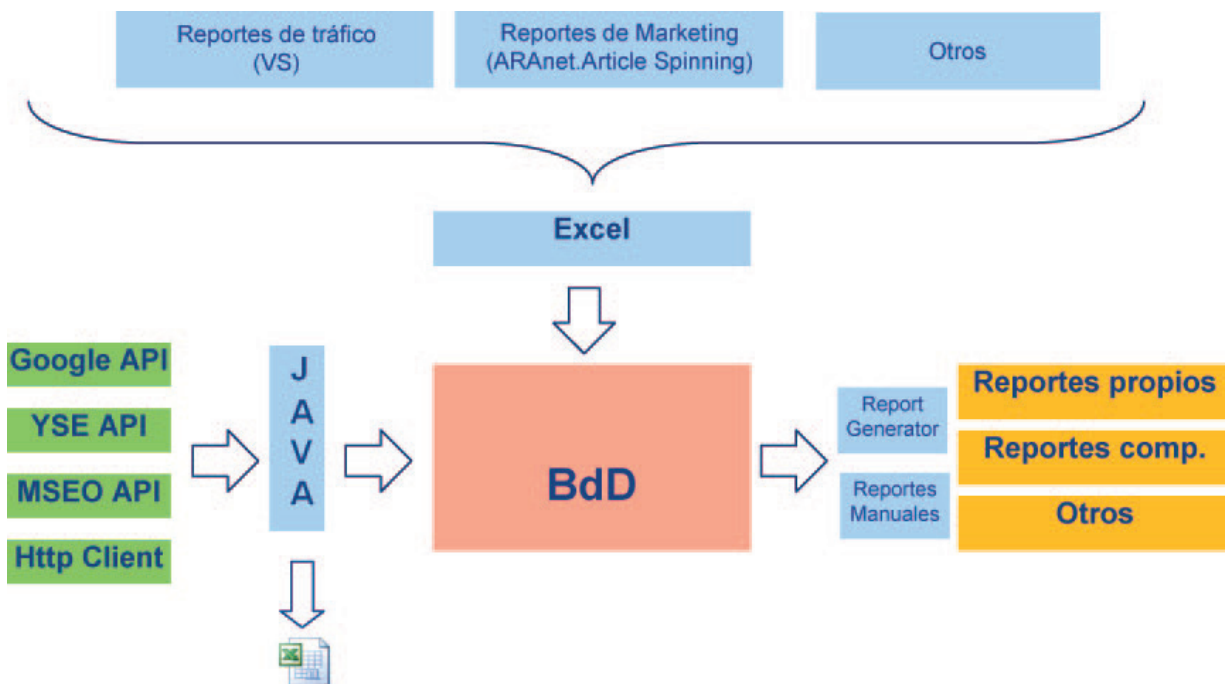
El objetivo de este proceso es que cualquier programador pueda codificar la generación del reporte que quiera dependiendo de sus necesidades de negocio, de manera de que este/estos se genere/n de forma automática sin necesidad de consultar manualmente a la base de datos. El fuerte de este proceso es que será modular, y que irá creciendo reporte a reporte.

Utilizaré dos librerías que se comuniquen con Excel, ambas libres y totalmente gratuitas. Utilizaré “JExcelAPI” para escribir en un archivo Excel, y “JFreeChart” para generar gráficos. He realizado una extensa investigación sobre qué librerías utilizar, y estas son las que satisfacen mis dos necesidades básicas para una librería: precio (nulo, ya que es gratis) y dificultad (baja).

Al finalizar con el modelado de la base de datos y la suite de herramientas, ejecutaré varias pruebas con una serie de keywords de ejemplo de manera de poder ver el output final: reportes (tanto generados por los procesos como obtenidos a través de consultas directas a la base de datos).

## VI. Práctica

### Sistema final



El sistema final está compuesto por la base de datos, como entidad central, dos tipos de sistemas de entrada (semi-automático, a través de Excel y automatizado a través de los procesos creados que se comunican con las diferentes fuentes de información y APIs correspondientes), y dos tipos de sistemas de salida (el proceso Report Generator para salidas automatizadas, y reportes manuales realizados a través de consultas a la base de datos por medio de la aplicación MySQL Query Browser).

### Base de datos

A partir de mi experiencia como analista SEO y a partir de las necesidades del negocio, pude modelar una base de datos relacional completa y modular (fácil de modificar). El modelo es el siguiente (también adjuntado en el anexo, para una mejor visualización):

(Para una mejor visualización, el diagrama será incluido en el anexo bajo “Diagrama de base de datos”)

Antes de analizar cada tabla es importante aclarar que todas las entidades poseen un ID numérico, ya que al utilizar un valor numérico como identificador los algoritmos de búsqueda, matching y ordenamiento funcionan más rápido [45].

También, al modelar la base de datos, me pareció una buena idea agregar tablas que puedan contener información extra que no necesariamente deba ser ingresada a través de los procesos. De esta manera, se alienta a programadores a seguir aumentando el tamaño del sistema como también de la suite de herramientas. En la primera implementación de este sistema, estoy ingresando información a este tipo de tablas a través de una forma semi-automatizada, utilizando Excel.

Vale la pena destacar también que tanto las tablas como los atributos están en inglés, para darle mejor inserción en mi día a día laboral y en la informática en general. Describiré a continuación las tablas (las más pertinentes y/o más difíciles de entender serán descritas con mayor profundidad). Todos los gráficos (tanto los de la base de datos, los del sistema, los de cada proceso y los ejemplos de reportes) estarán incluidos al final de la tesina en los anexos.

## Entidades de la base de datos

### Keyword

La tabla 'keyword' es, obviamente, una de las más importantes y centrales de toda la base de datos. En ella radican las keywords, junto con ciertos atributos que pasaré a detallar:

- Id\_keyword: primary key de la tabla
- Value: valor literal de la keyword
- LOB: línea de negocio de la keyword. Básicamente, es una manera de agrupar las keywords. Entenderemos mejor este concepto cuando veamos los reportes
- Category: categoría de la keyword (utilizada para luego filtrar más fácilmente)
- Category2: categoría secundaria de la keyword
- Type: tipo (de interés alto, medio o bajo)
- Importance: en qué tipo de campaña se utilizará la keyword

### URL

Algo que caracteriza a esta base de datos es que le da mucha importancia a las URLs, a diferencia de otros sistemas que sólo se interesan en las keywords. La principal razón es porque la URL alberga todos los esfuerzos relacionados a SEO dentro de mi página, como también mi contenido. Algo en ese nivel de importancia merece un análisis más centrado. Atributos:

- Id\_URL: primary key de la tabla
- Value: valor literal de la URL
- url\_category: categoría de la URL
- url\_type: tipo de URL
- fk\_URL\_domain: foreign key que relaciona la tabla con la tabla 'dominio'. Cada URL debe tener obligatoriamente un dominio relacionado, y un dominio puede tener varias URLs (relación n:1)

### Keyword\_URL

La relación keyword\_URL. Decidí realizar una tabla para esto ya que hay varios atributos importantes y determinantes que necesitan guardarse, y que veremos a continuación:

- id\_keyword\_URL: primary key de la tabla
- fk\_keyword\_URL\_keyword y fk\_keyword\_URL\_URL: foreign keys que relacionan la tabla con las entidades 'keyword' y 'URL'
- order: la importancia de la keyword con respecto a la URL. Si es de orden 1, significa que la URL va a estar optimizada para esa keyword. Si es de orden 2, significa que los contenidos de la URL deberían relacionarse de cierta manera con la keyword, pero que la misma no es su keyword principal
- density: la densidad de la keyword en la URL. Cuál es el porcentaje de ocurrencias de la keyword en el contenido de la URL
- category: categoría de la relación
- locale: país a la que está orientada esta relación. Esto sirve para empresas grandes donde se utilizan varios sitios destinados a varios países
- hasAra, hasSpinning: ambos atributos están relacionados con lógica de negocios interna. "Ara" y "Spinning" son campañas de link building. Creando estos dos atributos puedo indicar qué pares keyword\_URL fue atacados con estas campañas. Sirven para filtrar más fácilmente estos pares, junto con las tablas 'Ara' y 'Article\_Spinning', que tendrán más información sobre esos esfuerzos, como por ejemplo fecha y cantidad de links obtenidos

### Dominio

Cada URL tendrá un dominio asociado, de manera de que, al momento de realizar reportes, pueda filtrar y ordenar por dominios para tener una vista general propia y de mis competidores. Tiene dos atributos, la primary key y el valor literal del dominio, o nombre en otras palabras.

## Traffic, Booking y Booking Conversion

Estas tres tablas tienen una arquitectura similar. Las tres están relacionadas con la entidad ‘keyword’, y poseen datos sensibles sobre el comportamiento del usuario con nuestras páginas. También poseen un atributo “date”, que permite guardar los datos históricamente indicando una fecha.

‘Traffic’ persiste la cantidad de tráfico que ingresa (a través de Google) a través de cierta keyword. ‘Booking’ la cantidad de compras que ese tráfico llevó, y ‘Booking Conversion’ la cantidad de dinero ingresado como resultado de esas compras. La información para las tres tablas es ingresada manualmente. Esta información puede ser obtenida en la herramienta de Webmasters de Google. También posee atributos de “date” para guardar información histórica e ir comparando semana a semana/mes a mes. Se verán ejemplos más claros.

## Keyword\_Volume, local\_volume y keyword\_trend

Estas entidades persisten información relacionada con el volumen de búsqueda de cada keyword en Google. Esto significa “la cantidad de usuarios que buscan esa keyword en Google por mes”. Esta información se puede obtener en la Keyword Research Tool de Google.

La tabla ‘Keyword\_Volume’ persiste el volumen global de la keyword (un estimativo del número de búsquedas anual, importante para hacer el plan SEO anual). ‘Local\_volume’ persiste el volumen local, que está relacionado con el volumen mensual de la keyword (este puede y va a variar dependiendo de la temporada alta/baja de esa keyword). Por último, la tabla keyword\_trend permite predecir tendencias futuras de la búsqueda de una keyword (ejemplo, es fácil saber que la keyword “mundial de fútbol” tendrá mucho volumen en los meses del mundial, o la palabra “navidad” en diciembre).

## Article\_Marketing

Esta tabla se usará en cualquier tipo de empresa, aunque sus atributos cambiarán dependiendo de los proveedores de artículos que se utilicen en campañas de marketing de artículos. Esto va a ser moldeable a las necesidades del usuario/empresa.

## PR

Servirá para indicar las variaciones de PageRank de las URLs a través del tiempo.

## Ranking

Como su nombre lo indica, esta tabla guardará los datos de posicionamiento de un par keyword\_URL. Atributos:

- id\_ranking: primary key de la tabla
- date: importante para guardar la fecha del posicionamiento
- locale: para qué país esta posicionándose esa keyword
- Google\_ranking: posición del par keyword\_URL
- Google\_SERP: página de los resultados de búsqueda donde está posicionado ese par keyword\_URL
- Bing y Yahoo: la tabla soporta los mismos datos para estos buscadores
- Fk\_ranking\_keyword\_URL: foreign key que relaciona la tabla con el par keyword\_URL correspondiente

## Rank\_Categories

Esta tabla está relacionada con la tabla ‘Ranking’. Básicamente persiste el tipo de ranking que es, como por ejemplo “Top 5”, “Competidor top 5”, “Página propia secundaria”, etc. Será más fácil de entender en los ejemplos.

### Indexed\_pages

Me indicará si cada URL estuvo indexada en determinada fecha. Veremos reportes enteros que utilizan información de esta tabla, información muy importante.

### Inbound\_links y link\_popularity

Estas tablas persistirán la información de backlinks proveniente del proceso YSE. Obviamente están relacionadas con la entidad 'URL' y poseen atributo 'date' para persistir datos históricos.

### Majestic\_backlink\_numbers y Majestic\_links

Esta tabla persistirá la misma información que las tablas previamente mencionadas, pero la misma vendrá de otro proveedor (un sitio llamado MajesticSEO, que posee un índice propio más grande que el de Yahoo con mucha información sobre links).

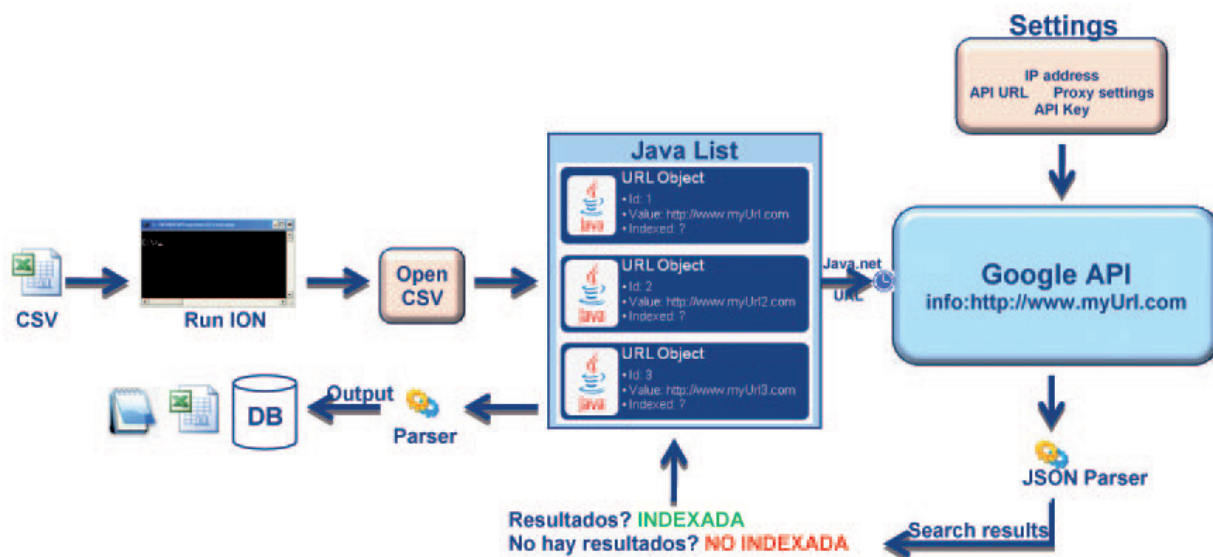
### Meta\_tags

Esta tabla persistirá toda la información obtenida por el proceso MTP, a través del tiempo. Del mismo modo que las tablas anteriores, esta tabla está relacionada con la entidad 'URL'.

### Link\_referrals

Persistirá la información obtenida por el proceso LC, y también estará relacionada con la entidad 'URL'.

### Proceso ION





Podemos ver una simplificación de la lógica del código a alto nivel. Pasaré a detallar los puntos más importantes:

1. La entrada del proceso es un archivo CSV (comma separated values, o valores separados por comas en inglés) que contiene las URLs deseadas (y sus IDs de la base de datos, en caso de que queramos que la salida del proceso vaya directamente a la misma). Un archivo de este estilo se genera con Excel sin ningún tipo de complicación.
2. Una vez que corremos el proceso, el módulo de lectura de archivos CSV comienza. El archivo es leído con “Open CSV”, una API libre que básicamente puede iterar y leer un archivo CSV, para luego generar objetos String (nativos de Java, básicamente son cadenas de caracteres) para su manejo.
3. El tercer paso es el de tomar estos String de información y generar objetos URL (creados por mí). Cada objeto va a tener un ID, un valor y un flag verdadero o falso que servirá para saber si la URL en cuestión está indexada o no.
4. A través de los paquetes apache.commons y java.net (ambos libres), creamos un cliente http que va a ser el encargado de enviar un método GET hacia la API de Google. El método GET (utilizado por los webservices del 100% de las grandes aplicaciones a nivel mundial), obtiene cualquier información identificada por el request (solicitud, pedido) enviado. El cliente va a mandar a la URL de la API ciertos parámetros de búsqueda, como también la IP a donde el resultado tiene que ser enviado. Como vimos anteriormente, estos parámetros van a ser “info:<URL>”. Entonces, junto con otros datos de configuración como IP y Proxy, el cliente http se va a encargar de este contacto. Nótese de un reloj en este paso. Eso significa que, después de ‘n’ cantidad de solicitudes a la API, debemos esperar unos segundos, o “dejar descansar a la API” en otras palabras. De esta manera, como nombramos anteriormente, seremos “amigables a la API”, previniendo así penalizaciones o bloqueos constantes. Esta implementación tuvo varias pruebas con fallos (he sido bloqueado varias veces, y he pedido perdón varias veces también), pero finalmente encontré una configuración de tiempo adecuada.
5. Una vez obtenida la réplica de la API de Google, debemos traducir esta respuesta en un lenguaje que podamos entender. Para aquello utilizamos un parser llamado JSON. Un parser es, básicamente, un analizador. Este analizador va a encontrar patrones en la respuesta e irá creando objetos que yo le solicite siguiendo esos patrones. De esa manera, puedo traducir esos resultados en resultados entendibles.
6. Con los resultados ahora entendibles, puedo llevar a cabo la lógica de negocios. Si existe algún resultado en la respuesta, significa que mi URL está indexada. En caso contrario, significa que no lo está. De esta manera, iré actualizado la lista de URLs que estoy analizando.
7. El paso final es traducir estos objetos Java en lenguajes adecuados para los diferentes tipos de salida. En caso de la base de datos y el archivo de texto plano, esta salida estará compuesta por sentencias de INSERTS en MySQL. En caso de la salida en Excel, utilicé una API libre llamada JExcelAPI que me permite escribir en un archivo fácilmente.
8. Se da por finalizado el proceso.

**Ejemplo de reporte directo de Excel (opción de salida del proceso)**

	A	B
1	URL	Indexed Today (10-21)?
2	http://www.travelocity.com/Flights	Yes
3	http://www.cheapflights.com	Yes
4	http://www.cheapair.com	Yes
5	http://www.expedia.com/Flights	Yes
6	http://www.priceline.com	Yes
7	http://www.tripadvisor.com/Flights	Yes
8	http://www.globester.com	Yes
9	http://www.expedia.com/daily/flights/default.asp	Yes
10	http://www.orbitz.com	Yes
11	http://www.orbitz.com/App/PrepareFlightsTab	Yes
12	http://travala.priceline.com/flights/?plf=pcln	Yes
13	http://www.flights.com	Yes
14	http://www.travelocity.com/Hotels	Yes
15	http://www.hotels.com	Yes
16	http://www.hotwire.com	Yes
17	http://www.expedia.com/Hotels	Yes
18	http://www.tripadvisor.com	Yes
19	http://www.orbitz.com/hotels	No
20	http://www.hotels.com/last-minute-hotel-deals	No
21	http://travala.priceline.com/hotels/?plf=pcln	Yes
22	http://www.expedia.com/Gillette-Hotels_d1289.Travel-Guide-Hotels	Yes
23	http://www.expedia.com/Riccione-Rimini-Hotels_0-n6051798-0.Travel-Guide-Filter-Hotels	Yes
24	http://www.travelocity.com/hotel-d4980-nevada-las-vegas-hotels	Yes
25	http://www.bellagio.com	Yes
26	http://www.vegas.com/resorts	Yes
27	http://www.venetian.com	Yes
28	http://www.wynnlasvegas.com	Yes
29	http://www.expedia.com/Las-Vegas-Hotels_d178276.Travel-Guide-Hotels	Yes
30	http://www.tripadvisor.com/Hotels-g45963-Las_Vegas_Nevada-Hotels.html	Yes
31	http://www.mgmgrand.com	Yes
32	http://www.orbitz.com/App/PerformMDLPDealsContent?deal_id=vegasvacation&cnt=PKH	Yes
33	http://www.expedia.com/Fremont-Street-Downtown-Las-Vegas-Las-Vegas-Hotels_0-n800047-0.Travel-Guide-Filter-Hotels	Yes
34	http://www.hotels.com/de1504033/hotels-las-vegas-nevada	No
35	http://www.vegas.com	Yes

Cómo podemos apreciar en la imagen, la salida del proceso en su modo Excel, nos va a mostrar cada URL solicitada junto con la fecha cuando fue ejecutado el proceso y el resultado del mismo. Como mencioné anteriormente, utilicé “JExcelAPI” para poder comunicarme con Excel y crear y escribir en un archivo .xls.

**Ejemplo de reporte obtenido desde la Bdd**

A	B	C	D	E	F	G	H
Id	URL	Type	Was Indexed On 7/23?	Was Indexed On 7/30?	Was Indexed On 8/06?	Was Indexed On 8/13?	Is Indexed Today 8/20?
152542	http://www.travelocity.com/flight-d155-kuwait-kuwait-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152543	http://www.travelocity.com/flight-d166-nigeria-lagos-cheap-flights	FLP	No	Yes	Yes	No	Yes
152544	http://www.travelocity.com/flight-d269-morocco-casablanca-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152545	http://www.travelocity.com/flight-d301-senegal-dakar-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152546	http://www.travelocity.com/flight-d337-uganda-entebbe-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152547	http://www.travelocity.com/flight-d377-nepal-kathmandu-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152548	http://www.travelocity.com/flight-d401-malaysia-penang-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152549	http://www.travelocity.com/flight-d430-malaysia-kuala-lumpur-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152550	http://www.travelocity.com/flight-d441-go-go-cheap-flights	FLP	No	Yes	No	No	Yes
152551	http://www.travelocity.com/flight-d452-south-korea-cheju-city-cheap-flights	FLP	No	Yes	No	No	No
152552	http://www.travelocity.com/flight-d455-singapore-singapore-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152553	http://www.travelocity.com/flight-d473-philippines-cebu-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152554	http://www.travelocity.com/flight-d503-philippines-manila-cheap-flights	FLP	No	Yes	No	Yes	Yes
152555	http://www.travelocity.com/flight-d541-china-hong-kong-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152556	http://www.travelocity.com/flight-d544-cambodia-phnom-penh-cheap-flights	FLP	No	Yes	No	Yes	No
152557	http://www.travelocity.com/flight-d553-bangladesh-dhaka-cheap-flights	FLP	No	Yes	No	Yes	Yes
152558	http://www.travelocity.com/flight-d631-egypt-cairo-cheap-flights	FLP	No	Yes	No	Yes	Yes
152559	http://www.travelocity.com/flight-d653-south-korea-incheon-cheap-flights	FLP	No	Yes	No	Yes	Yes
152560	http://www.travelocity.com/flight-d662-south-korea-pusan-cheap-flights	FLP	No	Yes	Yes	No	Yes
152561	http://www.travelocity.com/flight-d734-no-st-lucia-cheap-flights	FLP	No	Yes	No	No	No
152562	http://www.travelocity.com/flight-d741-grenada-grenada-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152563	http://www.travelocity.com/flight-d747-bermuda-bermuda-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152564	http://www.travelocity.com/flight-d748-barbados-barbados-cheap-flights	FLP	No	Yes	Yes	Yes	No
152565	http://www.travelocity.com/flight-d750-aruba-aruba-cheap-flights	FLP	No	Yes	Yes	Yes	Yes
152566	http://www.travelocity.com/flight-d756-bahrain-bahrain-cheap-flights	FLP	No	Yes	Yes	Yes	Yes

Este reporte fue generado a través de una consulta a la base de datos. Como podemos ver, en esta oportunidad es fácil encontrar URLs problemáticas a través del tiempo, como también es fácil analizar

si los esfuerzos realizados dieron resultado o no. Vamos viendo como unos de los objetivos del sistema (facilitar las decisiones SEO) se va cumpliendo.

Cada columna nos indica una fecha y si la URL fue indexada o no. De esta manera, podemos generar gráficos de este estilo:

	23-Jul	30-Jul	6-Aug	13-Aug	20-Aug
<b>Indexed</b>	46	614	439	505	536
<b>Not Indexed</b>	582	14	189	123	93
<b>Index rate</b>	7%	98%	70%	80%	85%

Podemos entonces analizar nuestra performance con respecto a las campañas de indexación realizadas. Como verán también en la tabla, utilizamos URL categorizadas como “FLP”. Esta categorización tiene que ver con el negocio, y nos permite filtrar de forma sencilla en la base de datos para, de este modo, generar reportes de este estilo (no solo propios, sino con las URLs de nuestros competidores).

La consulta en lenguaje SQL utilizada fue la siguiente:

```

SELECT
u.id_URL as 'Id',
u.value as 'URL',
u.url_type as 'Type',

(SELECT i.isIndexed
FROM indexed_pages i left join url u1
on i.fk_indexed_pages_URL = u1.id_URL
WHERE u1.url_type = "FLP" AND i.date = "2010-07-23" AND u1.id_URL = u.id_URL) as 'Was Indexed On 7/23?',

(SELECT i.isIndexed
FROM indexed_pages i left join url u2
on i.fk_indexed_pages_URL = u2.id_URL
WHERE u2.url_type = "FLP" AND i.date = "2010-07-30" AND u2.id_URL = u.id_URL) as 'Was Indexed On 7/30?',

(SELECT i.isIndexed
FROM indexed_pages i left join url u3
on i.fk_indexed_pages_URL = u3.id_URL
WHERE u3.url_type = "FLP" AND i.date = "2010-08-06" AND u3.id_URL = u.id_URL) as 'Was Indexed On 8/06?',

(SELECT i.isIndexed
FROM indexed_pages i left join url u4
on i.fk_indexed_pages_URL = u4.id_URL
WHERE u4.url_type = "FLP" AND i.date = "2010-08-13" AND u4.id_URL = u.id_URL) as 'Was Indexed On 8/13?',

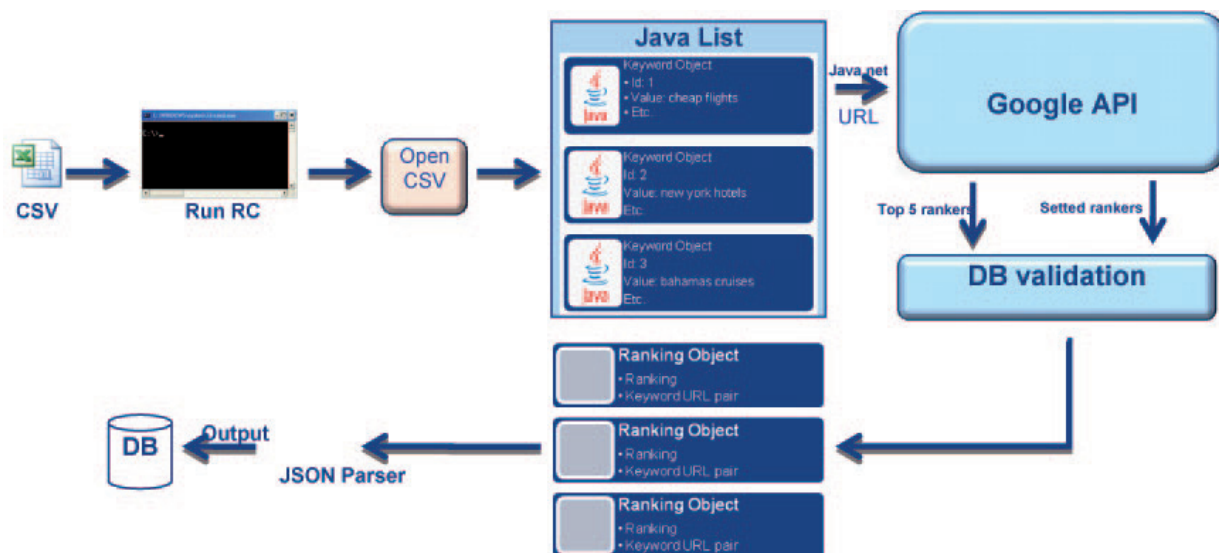
(SELECT i.isIndexed
FROM indexed_pages i left join url u5
on i.fk_indexed_pages_URL = u5.id_URL
WHERE u5.url_type = "FLP" AND i.date = "2010-08-20" AND u5.id_URL = u.id_URL) as 'Is Indexed Today 8/20?'

FROM
url u

WHERE
u.url_type = "FLP"
    
```

En un principio, al realizar las consultas, creaba tablas temporales por cada fecha deseada. Pero luego de investigar encontré una manera de eliminar estas tablas temporales y filtrar directamente en la consulta SELECT (como se puede apreciar en el ejemplo) resultando en consultas más fáciles de entender y, sobre todo, mucho más rápidas y eficientes.

### Proceso RC



Recordemos que este proceso utiliza, de la misma manera que el proceso anterior, la API de Google. Por lo tanto la comunicación con la misma será idéntica, con la diferencia de que en este caso los resultados serán diferentes, como también la lógica de negocio utilizada.

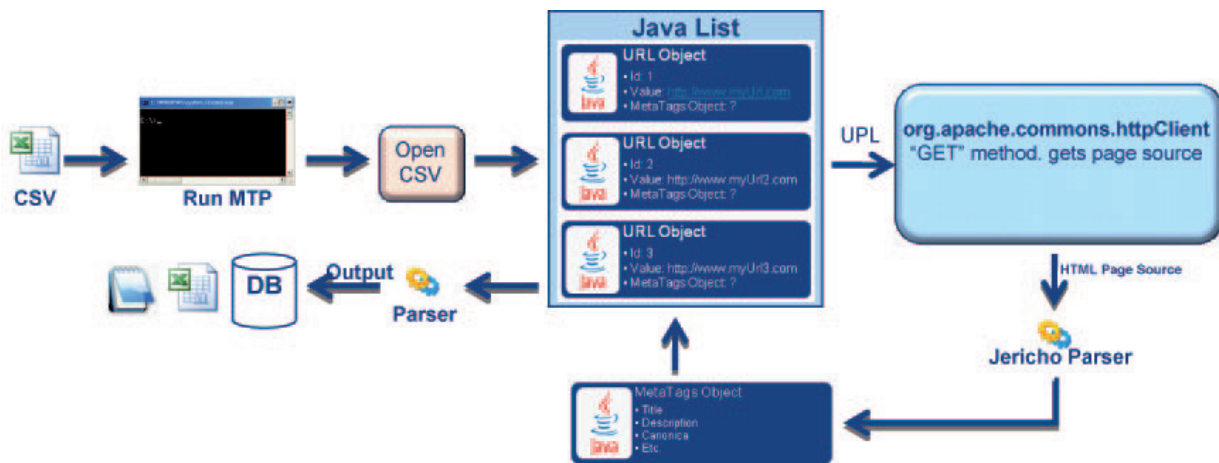
1. La entrada de este proceso es una lista de keywords de las cuales deseamos saber su posicionamiento. Al igual que en el proceso anterior, esta entrada es leída por un módulo de lectura implementado con “Open CSV” con la diferencia que, en este caso, la salida de ese módulo serán objetos Keyword.
2. De la misma manera que en el proceso anterior, los paquetes apache.commons y java.net enviarán una solicitud a la API de Google, pero esta consultando por la cada keyword.
3. De la salida de este proceso se tomarán las URLs posicionadas en los primeros 5 lugares (importante información de la competencia). A estas URLs se les harán tres tipos de validaciones:
  - a. Se validará que el dominio de la URL exista en la base de datos. En caso de que no exista, se agregará.
  - b. Se validará que la URL exista en la base de datos. En caso de que no exista, se agregará.
  - c. Se validará que el par keyword\_URL exista en la base de datos. En caso de que no exista, se agregará. Las tres validaciones incluyen un algoritmo que cargará todas las URLs, dominios y pares keyword\_URL en memoria para mayor velocidad y para facilitar el matching de ids.
4. Luego se ejecutará el mismo módulo, pero con la diferencia de que se buscarán URLs posicionadas que contengan el nombre de nuestro sitio y el nombre de una lista de competidores que previamente le debemos especificar al proceso. De esta manera, no sólo tendremos información de los primeras 5 URLs posicionadas para cada keyword, sino también de la mejor URL propia posicionada para cada keyword, y la mejor URL de cada competidor nuestro para cada keyword (lo veremos en el ejemplo a continuación).
5. El siguiente paso es utilizar toda esta información ordenada para crear objetos Ranking, que poseen el id del par keyword\_URL, la posición y la página de resultados en la que se encuentra.
6. Por último se utiliza un parser para volcar la información de estos objetos en sentencias INSERT, y agregarlos a la base de datos.

### Ejemplo de reporte obtenido desde la BdD

A	B	C	D	E	F	G
Domain	Keyword	URL	2010-10-31	2010-10-24	SERP 2010-10-31	SERP 2010-10-24
www.travelocity.com	cheap flights	http://www.travelocity.com/Flights	3	4	1	1
www.cheapflights.com	cheap flights	http://www.cheapflights.com	1	1	1	1
www.cheapair.com	cheap flights	http://www.cheapair.com	2	2	1	1
www.fly.com	cheap flights	http://www.fly.com	4	3	1	1
www.expedia.com	cheap flights	http://www.expedia.com/Flights	15	14	2	2
www.priceline.com	cheap flights	http://www.priceline.com	5	5	1	1
www.tripadvisor.com	cheap flights	http://www.tripadvisor.com/Flights	11	12	2	2
www.orbitz.com	cheap flights	http://www.orbitz.com	82	75	9	8
www.orbitz.com	cheap flights	http://www.orbitz.com/App/PrepareFlightsTab	18	18	2	2
www.priceline.com	cheap flights	http://travala.priceline.com/flights?plf=pcln	38	36	4	4

Como se puede apreciar en el ejemplo, podemos obtener una tabla con nuestra performance y la de nuestros competidores en cada keyword. Más adelante veremos cómo podemos utilizar esta información para un análisis más abarcativo.

### Proceso MTP



1. Al igual que en los procesos anteriores, utilizamos el módulo de lectura de archivos CSV a través de Open CSV. La salida de este módulo serán objetos URL con: ID, la URL literal, y un objeto MetaTags asociado vacío.
2. A través de apache.commons volvemos a utilizar el método GET para salir a Internet, pero en este caso nos comunicaremos directamente con la URL deseada. A través del manejo de inputs, podemos obtener el código fuente en HTML de la URL, que bajaremos localmente.
3. El código fuente contiene toda la información que deseamos, pero en un lenguaje que no entendemos. Para traducirlo utilizamos en Jericho Parser, un parser libre que es utilizado para traducir HTML a objetos deseados por el programador.
4. En este caso, programamos al parser para que obtenga el título, meta keywords, meta description, H1, H2; H3; H4, URL canónica y cantidad de links en el sitio para luego crear un objeto MetaTags completo que incluiremos en el objeto URL.
5. El último paso es el de utilizar un parser para traducir estos objetos Java en la salida deseada (INSERTS de MySQL o escribir en un archivo de Excel).

Ejemplo de reporte directo de Excel

#	A	B	C	D	E	F	G	H	I	J
1	URL	Title	Keywords	Description	H1 (separated by ';')	H2 (separated by ';')	H3 (separated by ';')	H4 (separated by ';')	Canonical URL	Links
2	<a href="http://www.travelocity.com/Flights">http://www.travelocity.com/Flights</a>	Cheap Flights - Find Cheap Tic	cheap flights, cheap fl	Cheap flights and	<span>Find Cheap	Need Ideas? - Flig	Advanced Search	None	<a href="http://www.travelo">http://www.travelo</a>	151
3	<a href="http://www.cheapflights.com">http://www.cheapflights.com</a>	Cheap Flights - Airline Tickets	cheap flights, airline ti	Compare cheap flig	Find Cheap Airline	<span>Price com	Search preference	Beach Escapes =	<a href="http://www.cheapfl">http://www.cheapfl</a>	156
4	<a href="http://www.cheapair.com">http://www.cheapair.com</a>	Cheap Flights - Airline Tickets	cheap flights, airline ti	Find cheap airline	Search Cheap Air	Search Best Car R	Find Cheap Flights	<a href="http://www	None	191
5	<a href="http://www.expedia.com/Flights">http://www.expedia.com/Flights</a>	Flights: Find the Cheapest Flig	Flights, Cheap Flights	Cheap flights are a	Find Cheap Flights	Search for a flight	Don't see the city	<a id="cx_why_bo	<a href="http://www.expedi">http://www.expedi</a>	766
6	<a href="http://www.priceline.com">http://www.priceline.com</a>	Priceline.com   Cheap Flights	Priceline, travel, cheap	Get deep discount	shop for discount	<a onclick="gfCall	No One Out-Negot	<span class="hdr	<a href="http://www.pricelin">http://www.pricelin</a>	424
7	<a href="http://www.tripadvisor.com/Flights">http://www.tripadvisor.com/Flights</a>	Cheap Flights and Airline Ticke	airline tickets, flights,	Find the best deal	Cheap Flights - All	Popular airfares; f	None	Child & Senior Far	None	108
8	<a href="http://www.globester.com">http://www.globester.com</a>	Airline Tickets, Cheap Flights,	airline tickets, cheap	Find cheap flights	<a href="http://ww	Airline Tickets, Ch	October-2010 Che	None	None	220
9	<a href="http://www.expedia.com/daily/flights">http://www.expedia.com/daily/flights</a>	Flights: Find the Cheapest Flig	Flights, Cheap Flights	Cheap flights are a	Find Cheap Flights	Search for a flight	Don't see the city	<a id="cx_why_bo	<a href="http://www.expedi">http://www.expedi</a>	766
10	<a href="http://www.orbitz.com">http://www.orbitz.com</a>	Orbitz Travel - Airline Tickets	cheap airline tickets,	Book cheap airline	None	None	Need help booking	Manage Corporate	<a href="http://www.orbitz.c">http://www.orbitz.c</a>	88
11	<a href="http://www.orbitz.com/App/Prepare">http://www.orbitz.com/App/Prepare</a>	Cheap Flights, Cheap Tickets,	discount airline tickets	Save on cheap flig	None	None	Top US Flight Des	None	<a href="http://www.orbitz.c">http://www.orbitz.c</a>	77
12	<a href="http://travels.priceline.com/flights/7/">http://travels.priceline.com/flights/7/</a>	Find Cheap Flight Tickets & Se	Discount airline ticket	Search for cheap f	search flights ;	<span>flight deals	flights up to 50%	<span class="hdr	<a href="http://www.pricelin">http://www.pricelin</a>	271
13	<a href="http://www.flights.com">http://www.flights.com</a>	Flights.com - Cheap Flights, Ai	Flights, Discount Airfa	Travel and save wi	None	None	Shop for discount	None	None	93
14	<a href="http://www.travelocity.com/Hotels">http://www.travelocity.com/Hotels</a>	Travelocity: Hotels, Motels and	Hotels, Hotel, Motels	New! Save up to 4	<span>Find Your	Cheap Hotel Deals	Travelocity Top Se	None	<a href="http://www.travelo">http://www.travelo</a>	139
15	<a href="http://www.hotwire.com">http://www.hotwire.com</a>	Hotels.com - Hotel rooms with	business hotels, corp	Choose from luxur	Our Promise to Yo	Stay 10 nights. Ge	Where are you goi	None	None	122
16	<a href="http://www.hotwire.com">http://www.hotwire.com</a>	Cheap & Discount Travel Deals	None	Find cheap and dea	None	None	Know when prices	Hotwire Hot Rate	<a href="http://www.hotwire">http://www.hotwire</a>	78
17	<a href="http://www.expedia.com/Hotels">http://www.expedia.com/Hotels</a>	Hotels: Find cheap hotel deals	resort, cheap hotels,	Cheap hotel deals	Hotel Deals & mp	<span id="hot-title	Call Center Hours	<a id="cx_why_bo	<a href="http://www.expedi">http://www.expedi</a>	291
18	<a href="http://www.tripadvisor.com">http://www.tripadvisor.com</a>	Reviews of Hotels, Flights and	vacation, vacations, va	TripAdvisor - Unbia	Plan the Perfect T	<div class="Label	None	None	None	145
19	<a href="http://www.orbitz.com/hotels">http://www.orbitz.com/hotels</a>	Hotels, Hotel Rooms, Discount	<p>last minute hotel	<p>Book hotels, c	None	None	Top US hotel dest	None	<a href="http://www.orbitz.c">http://www.orbitz.c</a>	149
20	<a href="http://www.hotels.com/last-minute/">http://www.hotels.com/last-minute/</a>	URL not working	URL not working	URL not working	URL not working	URL not working	URL not working	URL not working	URL not working	0
21	<a href="http://travels.priceline.com/hotels/7/">http://travels.priceline.com/hotels/7/</a>	Find Cheap Hotels & Discount	Discount hotel rooms	Search for cheap f	search hotels ;	name your own pr	featured partners	None	<a href="http://www.pricelin">http://www.pricelin</a>	288
22	<a href="http://www.expedia.com/Gillette-Ho">http://www.expedia.com/Gillette-Ho</a>	Gillette Hotels - Compare Hotel	Gillette Hotels, Gillette	Need a hotel in Gi	Gillette Hotels ;	Other hotel option	Narrow results by:	<a href="#" class=	<a href="http://www.expedi">http://www.expedi</a>	188
23	<a href="http://www.expedia.com/Riccione-Ho">http://www.expedia.com/Riccione-Ho</a>	Riccione Hotels near Rimini - B	Riccione Rimini Hotel	Need a hotel in Ri	Riccione Hotels n	Other hotel option	Your selection: N	<a href="#" class=	<a href="http://www.expedi">http://www.expedi</a>	263
24	<a href="http://www.travelocity.com/Hotels-d4">http://www.travelocity.com/Hotels-d4</a>	Las Vegas Hotels - Find Your H	Hotels, Las Vegas Ho	Las Vegas Hotels	Las Vegas Hotels	Know When You'r	Star Rating - Star	<a href="#" id="tra	<a href="http://www.travelo">http://www.travelo</a>	217
25	<a href="http://www.bellagio.com">http://www.bellagio.com</a>	Las Vegas Hotels - Bellagio	None	Bellagio is ultim	None	None	None	None	None	117
26	<a href="http://www.vegas.com/resorts">http://www.vegas.com/resorts</a>	Las Vegas Hotels, Hotels in La	las vegas hotels, hotel	Save money on La	Las Vegas Hotels	Featured Las Veg	<a href="#">Vegas	None	None	226
27	<a href="http://www.venetian.com">http://www.venetian.com</a>	The Venetian - Resort, Hotel, Cl	las vegas hotel, las ve	Of the top hotels L	<a href="#">LangTy	None	None	None	None	118
28	<a href="http://www.wynnlasvegas.com">http://www.wynnlasvegas.com</a>	Las Vegas Hotels - Wynn Las	Vegas Hotels, Las Ve	Wynn Las Vegas	None	None	None	None	None	2
29	<a href="http://www.expedia.com/Las-Vegas">http://www.expedia.com/Las-Vegas</a>	Las Vegas Hotels - Compare H	Las Vegas Hotels, La	Need a hotel in La	Las Vegas Hotels	Other hotel option	Narrow results by:	<a href="#" class=	<a href="http://www.expedi">http://www.expedi</a>	329
30	<a href="http://www.tripadvisor.com/Hotels-g">http://www.tripadvisor.com/Hotels-g</a>	Las Vegas Hotels - Read Las Vi	Las Vegas, Nevada, L	October 25, 2010	Las Vegas, NV Ho	Las Vegas; Refin	Neighborhoods; R	None	None	167
31	<a href="http://www.mgmgrand.com">http://www.mgmgrand.com</a>	MGM Grand Hotel & Casino	None	None	None	None	None	None	None	124
32	<a href="http://www.orbitz.com/App/Perform">http://www.orbitz.com/App/Perform</a>	Las Vegas Vacations, Las Veg	Las Vegas Vacation	Find great rates of	Markel	None	What is a promoti	I have a promotion	<a href="http://www.orbitz.c">http://www.orbitz.c</a>	463
33	<a href="http://www.expedia.com/Fremont-S">http://www.expedia.com/Fremont-S</a>	Fremont Street - Downtown Las	Fremont Street - Down	Need a hotel in Fr	Fremont Street - D	Other hotel option	Your selection: N	<a href="#" class=	<a href="http://www.expedi">http://www.expedi</a>	273
34	<a href="http://www.hotels.com/de1504033/H">http://www.hotels.com/de1504033/H</a>	Hotels in Las Vegas, Nevada, L	hotels, vegas, las veg	Las Vegas Hotel	<strong>Las Vega	Search for hotels i	Email " <strong>L	Thanks! " <a id="H	None	185
35	<a href="http://www.vegas.com">http://www.vegas.com</a>	Las Vegas Hotels - Las Vegas	las vegas, vegas, las	Planning a trip to U	None	Las Vegas Guide	<a href="#">resort	None	None	139

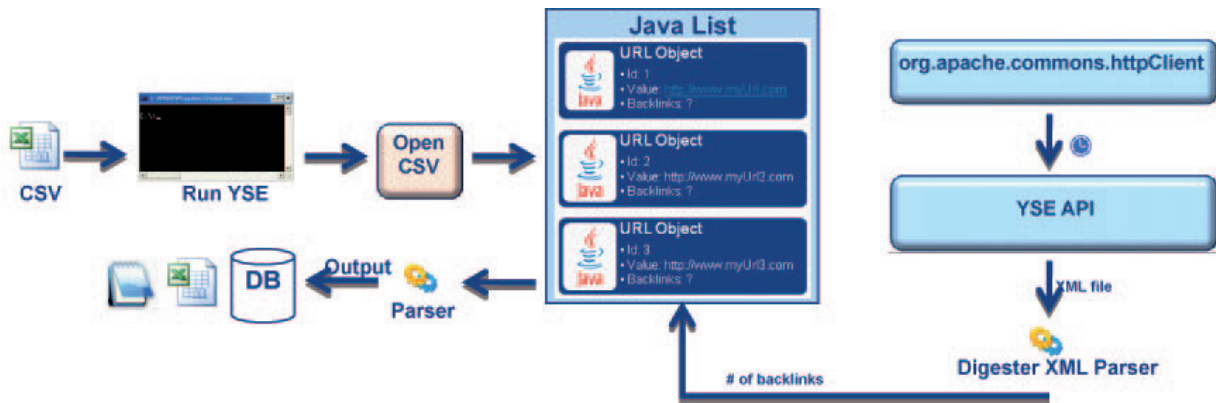
Podemos ver en este ejemplo la salida del proceso en su modo Excel. Cada atributo está dispuesto en una columna diferente. En caso de que los tags H1, H2, H3, H4 sean mayor a uno, los incluirá separándolos con el carácter ‘;’.

Ejemplo de reporte obtenido desde la BdD

<a href="http://aransaspassprogress.com/ara/travel/11312.txt">http://aransaspassprogress.com/ara/travel/11312.txt</a>	Aransas Pass Progress - Online Edition
<a href="http://aransaspassprogress.com/ara/travel/11309.txt">http://aransaspassprogress.com/ara/travel/11309.txt</a>	Aransas Pass Progress - Online Edition
<a href="http://aransaspassprogress.com/ara/travel/10957.txt">http://aransaspassprogress.com/ara/travel/10957.txt</a>	Aransas Pass Progress - Online Edition
<a href="http://aransaspassprogress.com/ara/travel/10960.txt">http://aransaspassprogress.com/ara/travel/10960.txt</a>	Aransas Pass Progress - Online Edition
<a href="http://aransaspassprogress.com/ara/travel/10941.txt">http://aransaspassprogress.com/ara/travel/10941.txt</a>	Aransas Pass Progress - Online Edition
<a href="http://aransaspassprogress.com/ara/travel/10933.txt">http://aransaspassprogress.com/ara/travel/10933.txt</a>	Aransas Pass Progress - Online Edition
<a href="http://aransaspassprogress.com/ara/travel/10945.txt">http://aransaspassprogress.com/ara/travel/10945.txt</a>	Aransas Pass Progress - Online Edition
<a href="http://aransaspassprogress.com/ara/travel/11306.txt">http://aransaspassprogress.com/ara/travel/11306.txt</a>	Aransas Pass Progress - Online Edition

Aquí podemos ver una aplicación rápida de este proceso. Supongamos que invertimos en la creación de artículos, y tercerizamos la instauración de los mismos. Luego, nuestro proveedor nos enviará las URLs de los artículos en cuestión para comprobar que realizó su trabajo. Luego de correr nuestro proceso con estas URLs como entrada, podemos observar como su título es literalmente igual. Entonces, podemos concluir que pueden ser tratadas en un futuro cercano como “contenido duplicado”, teniendo poco valor SEO.

**Proceso YSE**



1. El primer paso es igual al de los procesos anteriores: entrada a través de un módulo de lectura implementado con “Open CSV”. La salida de este módulo se utilizará para crear objetos URL, con ID, la URL literal y un atributo numérico backlinks.
2. Luego, se utilizará el paquete apache.commons para enviar una request a la API de Yahoo. Vale la pena aclarar que los tiempos de descanso de esta API son diferentes a la de Google, por lo tanto la implementación cambia.
3. La respuesta de la API de Yahoo no es igual a la de Google. Utiliza un lenguaje totalmente diferente, el XML.
4. Para entender XML debemos utilizar un parser XML, en este caso utilizamos Digester. XML es un lenguaje con arquitectura de tags (similar a HTML). Este parser permite establecer qué tags estamos esperando, para que después el parser mismo se dedique a traducirlos.
5. El siguiente paso es utilizar esta información traducida para completar los objetos URL.
6. Por último, se utiliza otro parser para volcar esta información en la base de datos o en un archivo Excel.

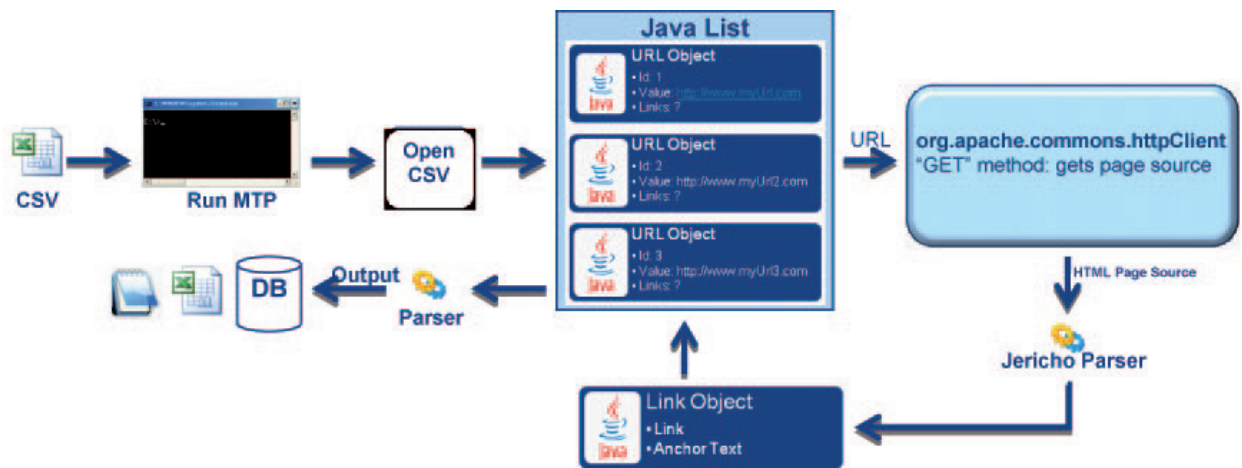
**Ejemplo de reporte directo de Excel**

URL	Backlinks Except For Domain
<a href="http://travala.priceline.com/insideTrack/flights/Charlotte-CLT-Dallas-DFW.html">http://travala.priceline.com/insideTrack/flights/Charlotte-CLT-Dallas-DFW.html</a>	0
<a href="http://travala.priceline.com/insideTrack/flights/from_Charlotte-CLT.html">http://travala.priceline.com/insideTrack/flights/from_Charlotte-CLT.html</a>	0
<a href="http://www.expedia.com/Charlotte-Hotels.d178247.Travel-Guide-Hotels">http://www.expedia.com/Charlotte-Hotels.d178247.Travel-Guide-Hotels</a>	142
<a href="http://www.expedia.com/Cheap-Flights-To-Charlotte.d178247.Travel-Guide-Flights">http://www.expedia.com/Cheap-Flights-To-Charlotte.d178247.Travel-Guide-Flights</a>	255
<a href="http://www.priceline.com/sale/sale.asp?CSSP=FlightsUnder150_09142009">http://www.priceline.com/sale/sale.asp?CSSP=FlightsUnder150_09142009</a>	0

En este ejemplo podemos ver como la salida del proceso es la lista de URLs junto con la cantidad de backlinks que poseen al día de la fecha. De esta manera podemos compararnos con nuestros competidores, como también así ver nuestro crecimiento/decrecimiento en el tiempo y los esfuerzos necesarios para sobrepasar a nuestra competencia.

Más adelante, en el reporte de performance, veremos más ejemplos que utilicen esta información.

**Proceso LC**



1. Este proceso imita al proceso MTP en todos sus aspectos, con excepción de la acción del Jericho Parser. En este caso, en vez de tomar cada meta tag, el parser está programado para obtener todos los “tags <a>” o, en otras palabras, todos los links que residen en la página.
2. Luego de obtener todos los links, se analiza qué link tiene como destino una página de mi dominio. En caso de que esto sea positivo, se analizará luego si ese link es normal o “nofollow”.
3. Luego de tener esta lista limpia de links, se crean objetos Link para luego ingresar en cada objeto URL y seguir su camino hacia el output final.

**Ejemplo de reporte directo de Excel**

	A	B	C
1	URI	TVL Links (separated by :)	TVL Nofollow Links (separated by :)
2	http://www.wisebread.com/topic/vacation	Cruise Deals: http://www.travelocity.com/RealDeals/0,,TRAVELOCITY_CR	None
3	http://www.wisebread.com/topic/airfare-0	Airline Tickets: http://www.travelocity.com	None
4	http://www.wisebread.com/3-step-strategy-to-airfare-deals	Cheap Flights: http://www.travelocity.com/Flights ;	None
5	http://www.wisebread.com/topic/booking-flights	Cheap Flights: http://www.travelocity.com/Flights ;	None
6	http://www.washingtonian.com/sections/visitorsguide/itineraries/index.html	Cheap Flights: http://www.travelocity.com/Flights ;	None
7	http://www.washingtonian.com/sections/travel/us/index.html	Cheap Travel: http://www.travelocity.com/	None
8	http://www.washingtonian.com/sections/travel/newyork/index.html	Cruises: http://www.travelocity.com/Cruises ;	None
9	http://www.wisebread.com/topic/flights-0	Flight: http://www.travelocity.com/flights ;	None
10	http://www.washingtonian.com/sections/travel/index.html	Travel: http://www.travelocity.com ;	None

En este reporte generado por el proceso LC, podemos ver si cada URL posee links referidos a nuestro dominio (como también su anchor text en caso positivo), tanto para links normales como para links “nofollow”.

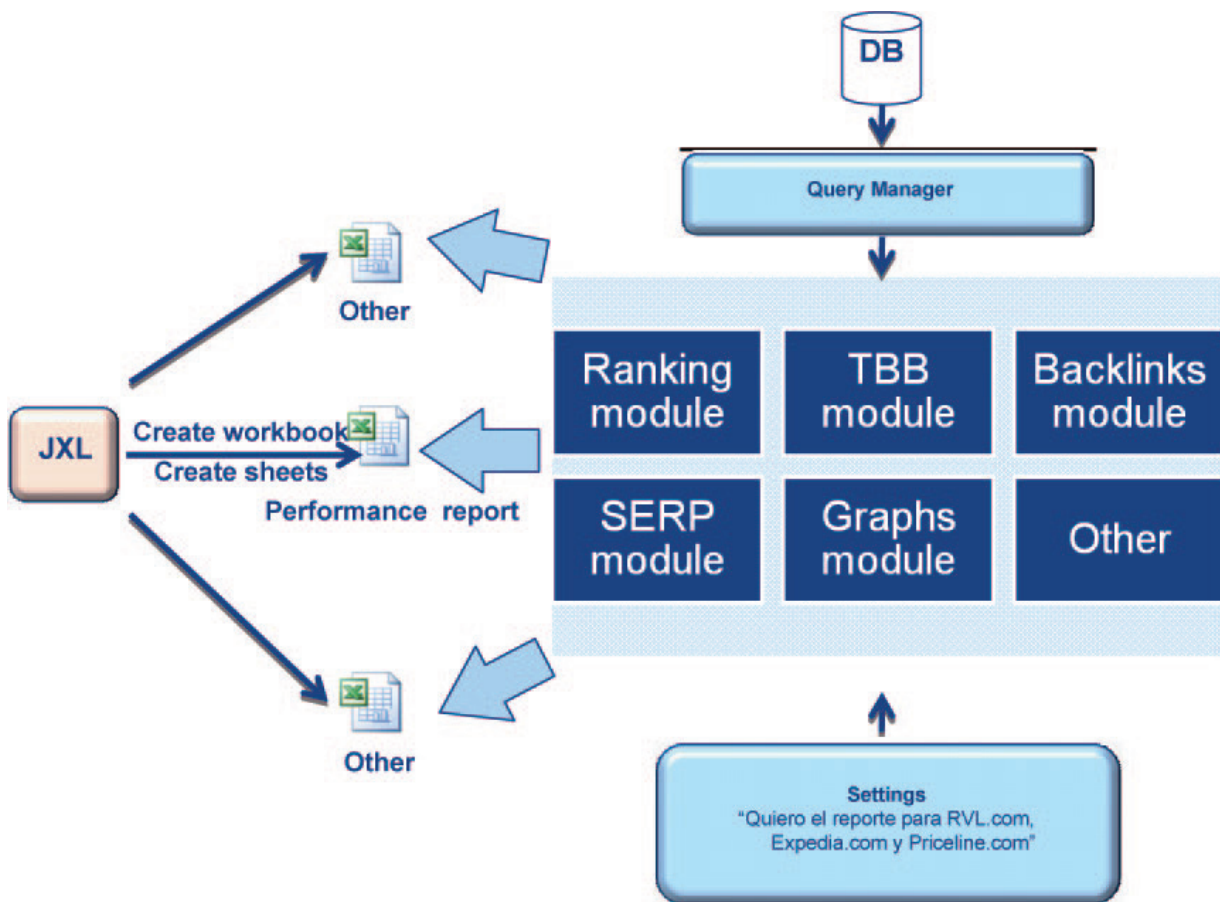
**Ejemplo de reporte obtenido desde la Bdd**

	A	B	C
1	URI	Links On 9/27	Links On 10/28
3038	http://www.iclassifiedsnetwork.com/v2/cc	Atlantic City hotels: http://www.travelocity.com/hotel-d5681-new-jersey-atlantic-city-hotels?configid=S21949556	URL not working
3039	http://www.sjnewsline.com/brandedcon	Houston cheap flights: http://leisure.travelocity.com/Promotions/0,,TRAVELOCITY 1931 airfare_houston,00.html	None
3040	http://www.waynepost.com/brandedcont	Vacation packages: http://www.travelocity.com/Vacations	Vacation packages: http://www.travelocity.com/Vacations

Aquí queda demostrado cuán importante es la salida de este proceso a través del tiempo. Supongamos que hemos adquirido links por medio de intercambio con sitios amigos. Podemos ver como una URL que tenía links nuestros en su contenido en Septiembre, hoy ya no los tiene (por ejemplo), o que otra URL antes los tenía y hoy en día ni siquiera funciona. De esta manera, es fácil y concreto verificar nuestros esfuerzos a través del tiempo, previniendo así la “fuga de links”.



Proceso RG



El último proceso no necesita comunicación con Internet ni con ninguna API externa, ya que utiliza lógica de negocios y la información ya persistida en nuestra base de datos. Vimos ejemplos de reportes generados automáticamente por cada proceso, como también reportes generados a partir de consultas directamente a la base de datos. Este proceso generará reportes automatizados previamente programados. Veremos cómo su arquitectura permite un crecimiento incremental y modular, adecuado a la necesidad de cada programador/línea de negocio.

1. Para la creación de el/los archivo/s de Excel, utilizamos la JExcelAPI y una API más potente y completa llamada JXL. Para la utilización de gráficos, utilizaremos la API JFreeChart. En el principio del proceso, la clase controladora verificará qué tipo de reportes será generado y creará los archivos y hojas de cálculo necesarias para cada uno de ellos (en este momento, hay un reporte codificado, el “Reporte de Performance”, que posee información completa en diferentes hojas de cálculo).
2. La clase controladora luego utilizará cada módulo necesario para generar estos reportes. Por ejemplo, el módulo de Rankings nos dará información sobre Rankings, y el módulo de Backlinks nos dará información de backlinks. Estos módulos son programados aparte, dándole un valor incremental a toda la aplicación. También es pertinente aclarar que se pueden reutilizar los módulos en diferentes archivos, a criterio propio, como también cambiar la lógica de cada módulo.
3. Cada módulo se comunicará con la base de datos a través de un “Administrador de consultas”, que básicamente es una clase encargada de consultar a la base de datos utilizando parámetros que cada módulo le envía.
4. Utilizando este administrador y otras configuraciones previas del usuario (como, por ejemplo, nombre de la empresa de la cual quiero información), el módulo ira persistiendo la información en el archivo correspondiente.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
<b>All Keywords</b>																					
SERP/date	2010-03-07	2010-03-14	2010-04-11	2010-05-02	2010-05-09	2010-05-16	2010-05-23	2010-06-06	2010-06-13	2010-06-20	2010-06-27	2010-07-11	2010-07-18	2010-07-25	2010-08-01	2010-08-08	2010-08-15	2010-08-22	2010-08-29	2010-09-05	
First Page	72	86	94	96	98	97	92	88	89	91	92	953	956	963	967	976	988	994	992	992	921
Second Page	30	37	43	46	44	35	41	39	43	46	44	322	344	356	359	316	299	312	312	339	342
Third Page	45	46	48	33	47	44	45	47	42	43	49	234	230	211	196	206	197	202	141	77	
Fourth Page	32	42	35	39	39	35	37	35	38	39	37	186	172	200	185	185	172	153	97	77	
Fifth Page	20	21	24	20	17	25	22	25	24	23	32	129	142	156	166	165	137	146	104	60	
Other	21	41	32	50	56	52	50	53	50	54	55	365	369	351	422	473	417	445	248	207	
<b>Cars</b>																					
SERP/date	2010-03-07	2010-03-14	2010-04-11	2010-05-02	2010-05-09	2010-05-16	2010-05-23	2010-06-06	2010-06-13	2010-06-20	2010-06-27	2010-07-11	2010-07-18	2010-07-25	2010-08-01	2010-08-08	2010-08-15	2010-08-22	2010-08-29	2010-09-05	
First Page	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1
Second Page	5	5	4	4	5	4	4	5	5	5	5	5	5	5	4	4	4	4	4	5	4
Third Page	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fourth Page	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Fifth Page	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Other	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
<b>Hotels</b>																					
SERP/date	2010-03-07	2010-03-14	2010-04-11	2010-05-02	2010-05-09	2010-05-16	2010-05-23	2010-06-06	2010-06-13	2010-06-20	2010-06-27	2010-07-11	2010-07-18	2010-07-25	2010-08-01	2010-08-08	2010-08-15	2010-08-22	2010-08-29	2010-09-05	
First Page	15	14	18	11	12	12	15	12	13	14	16	227	234	240	200	197	195	195	151	167	
Second Page	19	27	26	19	16	13	13	12	17	20	15	227	254	262	195	199	187	196	198	205	
Third Page	39	31	43	23	36	32	33	35	32	32	37	209	179	187	146	153	140	152	111	63	
Fourth Page	28	37	27	36	34	35	34	33	32	33	31	170	157	181	138	136	127	127	83	64	
Fifth Page	20	14	19	18	16	23	20	20	20	20	29	121	131	144	155	115	90	100	77	42	
Other	21	32	39	44	49	49	55	45	49	46	44	338	285	327	249	263	234	259	166	132	
<b>Cruises</b>																					
SERP/date	2010-03-07	2010-03-14	2010-04-11	2010-05-02	2010-05-09	2010-05-16	2010-05-23	2010-06-06	2010-06-13	2010-06-20	2010-06-27	2010-07-11	2010-07-18	2010-07-25	2010-08-01	2010-08-08	2010-08-15	2010-08-22	2010-08-29	2010-09-05	
First Page	5	6	5	2	1	2	3	2	4	4	4	9	9	8	5	3	3	5	8	5	
Second Page	0	0	2	2	1	2	1	2	2	2	2	3	3	3	2	2	2	2	4	4	
Third Page	2	2	1	0	1	1	1	0	0	0	0	3	2	2	2	2	3	4	4	2	
Fourth Page	1	0	1	1	0	0	0	0	0	0	0	1	1	2	1	2	2	1	1	1	
Fifth Page	0	1	0	0	0	0	0	0	0	1	1	2	2	1	2	0	1	1	1	0	
Other	0	0	0	1	1	1	2	2	3	1	1	2	2	2	0	1	3	1	1	1	

En este ejemplo, vemos una manera aun más concreta de ver nuestra performance semanalmente. Podemos ver la cantidad de keywords que se posicionan en cada grupo: primera página, segunda página, tercera página, cuarta página, etc., como también sus coloraciones respectivas. De esta manera, podemos analizar globalmente nuestros esfuerzos. Por ejemplo, vemos que en la LOB "Autos", no estamos trabajando directamente relacionado con SEO. Utilizamos el módulo de SERP para esto (SERP significa "Search Engine Result Pages", o en español "Páginas de resultados de búsqueda").

	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS
<b>All Keywords</b>																					
Ranking/date	2010-03-07	2010-03-14	2010-04-11	2010-05-02	2010-05-09	2010-05-16	2010-05-23	2010-06-06	2010-06-13	2010-06-20	2010-06-27	2010-07-11	2010-07-18	2010-07-25	2010-08-01	2010-08-08	2010-08-15	2010-08-22	2010-08-29	2010-09-05	
#1	0	0	2	0	2	2	2	2	2	1	1	30	32	27	31	31	52	32	29	13	29
#2	4	3	5	6	6	6	6	3	3	3	3	57	57	58	63	53	63	53	28	34	
#3	6	8	7	6	8	9	7	12	11	10	11	66	69	62	75	79	81	77	64	66	
#4	12	11	10	11	19	22	22	19	19	21	21	83	77	86	95	82	88	89	74	84	
#5	10	10	10	9	13	20	9	13	12	11	12	104	99	94	84	90	92	98	85	79	
<b>Cars</b>																					
Ranking/date	2010-03-07	2010-03-14	2010-04-11	2010-05-02	2010-05-09	2010-05-16	2010-05-23	2010-06-06	2010-06-13	2010-06-20	2010-06-27	2010-07-11	2010-07-18	2010-07-25	2010-08-01	2010-08-08	2010-08-15	2010-08-22	2010-08-29	2010-09-05	
#1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
#2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
#3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
#4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
#5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
<b>Hotels</b>																					
Ranking/date	2010-03-07	2010-03-14	2010-04-11	2010-05-02	2010-05-09	2010-05-16	2010-05-23	2010-06-06	2010-06-13	2010-06-20	2010-06-27	2010-07-11	2010-07-18	2010-07-25	2010-08-01	2010-08-08	2010-08-15	2010-08-22	2010-08-29	2010-09-05	
#1	0	0	1	1	1	1	1	1	1	1	0	12	12	16	16	16	14	14	7	13	
#2	2	2	3	1	3	3	3	1	1	1	1	27	29	21	15	16	14	14	7	13	
#3	1	1	1	3	1	1	1	2	2	2	2	13	18	15	19	18	21	19	12	18	
#4	3	4	2	2	2	2	2	1	1	1	1	26	19	22	24	24	24	19	14	16	
#5	1	2	0	0	1	0	0	4	2	1	1	30	33	34	19	24	24	24	19	15	
<b>Cruises</b>																					
Ranking/date	2010-03-07	2010-03-14	2010-04-11	2010-05-02	2010-05-09	2010-05-16	2010-05-23	2010-06-06	2010-06-13	2010-06-20	2010-06-27	2010-07-11	2010-07-18	2010-07-25	2010-08-01	2010-08-08	2010-08-15	2010-08-22	2010-08-29	2010-09-05	
#1	0	0	0	0	0	0	0	0	0	0	0	4	4	5	5	5	5	5	1	5	
#2	0	0	0	1	1	1	1	1	1	1	1	6	8	8	8	5	5	4	4	3	
#3	0	1	1	2	0	0	0	0	0	0	5	3	4	5	5	6	6	6	5	6	
#4	1	1	1	2	1	2	1	1	1	1	3	3	3	3	3	3	3	3	5	4	
#5	1	3	1	5	1	4	1	6	0	0	0	7	6	6	5	4	5	5	4	4	

Estas tablas son parte de la misma hoja que las tablas anteriores, pero con diferencia de que los grupos en este caso son primera posición, segunda posición, tercera posición, cuarta posición y quinta posición. De esta manera podemos analizar cómo estamos creciendo en las keywords que nos darán el mayor tráfico en nuestro sitio, las top 5.

Tesinas “3S: SEO System Suites” Suite de herramientas de SW libre para facilitar decisiones de SEO

	A	B	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR
	Keyword	LOB	2010 05-24	2010 05-31	2010 06-07	2010 06-14	2010 06-21	2010 06-28	2010 07-05	2010 07-12	2010 07-19	2010 07-26	2010 08-02	2010 08-09
2	cheap flights	Flights	3,445	3,464	8,506	7,195	3,276	2,637	2,864	3,196	2,934	4,017	3,573	2,466
3	hotels	Hotels	10,592	9,590	8,102	7,439	9,168	8,705	8,294	11,720	11,706	9,478	11,996	13,946
4	hotel	Hotels	827	710	690	708	1,058	1,157	1,096	1,177	1,930	1,692	2,307	1,953
5	las vegas hotels	Hotels	17	7	15	7	9	15	8	15	31	39	77	44
6	car rental	Cars	149	151	198	206	166	148	150	157	160	187	203	167
7	travel	Travel	2,563	2,584	2,052	2,380	2,468	2,225	2,452	2,331	2,148	2,195	2,246	2,145
8	motels	Hotels	138	158	190	189	207	294	188	257	247	225	288	331
9	airline tickets	Flights	7,152	7,838	8,312	5,676	6,366	6,022	7,411	8,197	8,142	7,764	7,664	6,877
10	new york city hotels	Hotels	1	8	0	3	7	24	14	21	21	5	6	18
11	paris hotels	Hotels	3	2	1	2	1	8	2	0	0	0	0	0
12	myrtle beach hotels	Hotels	0	0	0	0	0	0	0	0	0	0	0	0
13	orlando hotels	Hotels	3	4	2	4	1	3	2	0	0	0	0	0
14	london hotels	Hotels	2	1	2	3	0	4	0	0	0	1	0	0
15	chicago hotels	Hotels	0	0	0	0	0	0	0	0	0	0	0	0
16	cancun hotels	Hotels	0	0	0	0	0	0	0	1	1	1	2	3
17	san diego hotels	Hotels	2	4	1	1	2	4	1	0	0	0	0	0
18	san francisco hotels	Hotels	2	5	1	1	3	6	4	0	0	0	0	0
19	rental cars	Cars	90	79	58	95	92	76	78	77	65	90	86	62
20	virginia beach hotels	Hotels	3	2	2	4	1	6	6	0	0	0	0	0
21	cruise deals	Cruises	143	108	113	115	58	64	45	42	55	81	95	63
22	new york hotels	Hotels	3	2	0	2	3	1	3	0	0	2	0	0
23	cruises	Cruises	1,254	1,741	1,764	2,049	1,908	1,369	1,672	1,716	1,590	1,601	1,893	2,339
24	los angeles hotels	Hotels	0	0	0	0	0	0	0	0	0	0	0	0
25	boston hotels	Hotels	0	0	0	0	0	0	0	0	0	0	0	0
26	seattle hotels	Hotels	0	0	0	0	0	0	0	0	0	0	0	0
27	vacation packages	Vacations	578	1,039	1,157	1,161	1,101	955	1,674	1,254	1,247	1,854	965	741
28	cruise lines	Cruises	95	139	201	272	302	251	235	301	267	317	279	254
29	las vegas vacation packages	Vacations	240	265	237	250	278	250	275	254	216	209	172	230
30	las vegas vacations	Vacations	368	375	327	230	154	152	317	315	239	199	169	146
31	hawaii vacations	Vacations	3,622	4,120	3,408	2,340	2,710	3,475	4,325	4,035	4,213	2,692	3,369	2,749
32	Disney Vacations	Vacations	261	284	297	235	255	222	224	315	359	313	354	326
33	las vegas deals	Vacations	1,036	863	792	557	681	599	716	660	686	700	584	635
34	florida vacations	Vacations	93	194	200	323	371	264	143	149	152	44	39	23
35	cancun vacations	Vacations	305	358	505	253	209	177	172	162	101	168	127	131
36	mediterranean cruises	Cruises	758	847	892	799	912	782	858	846	844	828	811	1,016
37	key west vacations	Vacations	483	550	560	239	303	159	161	236	367	231	199	213
38	mexico vacations	Vacations	280	291	225	192	269	355	140	35	32	30	35	18
39	Cheap Tickets	Flights	656	653	1,071	1,510	1,342	1,136	813	917	763	733	599	685
40	las vegas cheap flights	Flights	0	0	0	0	0	0	0	0	0	0	0	0
41	los angeles cheap flights	Flights	0	0	0	0	0	0	0	0	0	0	0	0
42	flights	Flights	8,447	5,580	5,507	5,003	5,097	3,912	3,690	4,208	4,073	4,461	3,573	3,678
43	seattle cheap flights	Flights	0	0	0	0	0	0	0	0	0	0	0	0
44	travel deals	Travel	124	156	145	129	129	140	167	143	145	123	107	79
45	boston cheap flights	Flights	0	0	0	0	0	0	0	0	0	0	0	0

En este tráfico podemos apreciar el tráfico que ingresa a nuestro sitio semanalmente, a través de cada keyword. De esta manera podemos analizar cuáles son nuestras keywords que mejor rinden, o cuan relacionado está el posicionamiento con el tráfico (lo mismo para conversión de visitantes a compradores, todos ellos con el módulo de TBB).

A	B	C	D	E	F	G	H	I	J	K	L	M	N
URL	YSE Backlinks 2010.06.04	YSE Backlinks 2010.06.07	Link Gain	YSE Backlinks 2010.06.13	Link Gain	YSE Backlinks 2010.06.20	Link Gain	YSE Backlinks 2010.06.24	Link Gain	YSE Backlinks 2010.06.27	Link Gain	YSE Backlinks 2010.07.04	Link Gain
http://www.travelocity.com/Hotels	40,415	41,016	601	41,831	815	42,204	373	48,553	6,349	49,369	816	49,353	-16
http://www.travelocity.com/la	0	0	0	0	0	0	0	285,641	285,641	201,909	83,732	106,855	96,254
http://www.travelocity.com/LA/actividades	0	0	0	0	0	0	0	1	1	1	0	1	0
http://www.travelocity.com/LA/autos	0	0	0	0	0	0	0	0	0	0	0	0	0
http://www.travelocity.com/LA/hoteles	0	0	0	0	0	0	0	0	0	0	0	0	0
http://www.travelocity.com/LA/ofertas	0	0	0	0	0	0	0	1	1	1	0	1	0
http://www.travelocity.com/LA/paquetes	0	0	0	0	0	0	0	0	0	0	0	0	0
http://www.travelocity.com/LA/vuelos	0	0	0	0	0	0	0	1	1	1	0	1	0
http://www.travelocity.com/Travel-Deals	8,330	8,295	-35	8,162	-113	8,022	-140	7,265	-757	7,186	-79	6,988	-198
http://www.travelocity.com/TravelForGood/gr-directory.html	0	0	0	0	0	0	0	0	0	0	0	0	0
http://www.travelocity.com/usvi	0	0	0	0	0	0	0	0	0	0	0	0	0
http://www.travelocity.com/Vacations	6,859	6,854	-5	6,985	131	6,980	-5	7,806	826	8,223	417	8,248	25

En esta última hoja del reporte de performance generada por el módulo de backlinks, podemos no sólo ver la cantidad de links yendo a cada URL a través del tiempo, sino la pérdida/ganancia total de los mismos. De esta manera podemos ver si nuestros esfuerzos fuera del sitio están cumpliendo nuestros objetivos o no.

La idea general está expresada y cumplida: tener la posibilidad de generar reportes de forma automatizada, y que los mismos puedan ser programados y codificados a gusto de cada necesidad de negocio. En este caso, vimos ejemplos generales que pueden servirle a cualquier empresa, pero las limitaciones están en la imaginación del programador. La información existe (la BdD), sólo hace falta aplicar la lógica de negocios o las ideas necesarias para armar reportes que “piensen” y analicen por nosotros.

## VII. Conclusiones

El objetivo principal de esta tesina era crear una suite de herramientas libres que sirvan para facilitar decisiones estratégicas relacionadas a SEO, focalizándose en una base fuerte de información para trabajar de forma automatizada. Puedo confirmar que logré cumplir y sobrepasar estos objetivos. Una aplicación paga no siempre es la mejor solución, básicamente por dos razones. Primero por que hay que invertir mucho dinero (son muy costosas), y segundo por que ningún proveedor conoce los requerimientos internos de una empresa. Por lo tanto, al tener un sistema de código libre, el mismo puede ser moldeado en semanas, alineándose a lo que necesite.

El sistema general sirve como base SEO de cualquier empresa (pequeña, mediana o grande), ya que permite obtener información, relacionarla y crear reportes acordes a lo necesitado. Es destacable que en toda parte del sistema la mano del programador puede participar, dando lugar a uno de los pilares más fuertes de las aplicaciones libres que es la participación de cualquier programador con las habilidades técnicas requeridas para incrementar el valor general. Hoy en día, con un solo programador (yo), el sistema entero creció mucho y sigue creciendo, superando expectativas de diferentes managers y equipos alrededor de la empresa en la cual trabajo. Puedo confirmar también que unos de los fuertes del sistema es que no sólo se focaliza en “keywords” y en pares “keyword\_URL”, sino que también la entidad “URL” tiene un peso muy importante, premisa que nunca vi en ningún sistema pago o comparable con el mío, ni en ninguna demostración.

La base de datos resultó ser una solución perfecta y centralizada para persistir toda la información relativa a SEO. Al ser una base de datos relacional, podemos crear reportes y obtener información que filtre de manera horizontal a través de toda la base de datos, dando lugar a números/resultados 100% acordes a lo necesitado por cada negocio. Una de las premisas que mayor satisfacción me da es que, hoy en día con el sistema, la limitación no es más técnica ni funcional, sino “de la imaginación del usuario”. Tener datos históricos a través del tiempo permite hacer reportes mes a mes o año a año con valores claves para planificar campañas a futuro y realizar ROIs (análisis de retorno de inversión).

Con respecto al proceso ION, cumple altamente con sus objetivos ya que permite saber si el primer paso de todo proceso SEO (el elemental, el “existir”) se cumple o no. Como defendí anteriormente, no conozco de ninguna aplicación que lo haga.

El proceso LC es uno de los procesos más ventajosos de todos, ya que permite verdaderamente identificar qué técnicas de link building funcionan y qué técnicas son un desperdicio de inversión. He logrado muchos resultados aplicándolo en el día a día, y es otra aplicación que no he visto.

El proceso MTP es un proceso muy veloz para la cantidad de información que obtiene, y tiene como fuerte que puede obtener cualquier información siempre y cuando la misma esté en el código fuente del sitio. Dando lugar así a la participación de programadores para idear nuevos filtros, configuraciones y maneras de obtener esta información.

Los resultados que da el RC son resultados que todo analista SEO debe tener. La ventaja de este proceso es que me permite correrlo y modificarlo a gusto. Su contacto directo con la base de datos nos hace también ahorrar el paso de la validación, algo muy importante para ahorrar tiempo en “obtener y hacer reportes”, y ganándolo en “realizar el trabajo de analista”.

Los procesos para obtener links son complementos importantísimos a la hora de posicionarnos en Google. Internet crece día a día, y Google cambia junto con la red. Cada día se le da más importancia a los links y queda demostrado gracias al análisis de los resultados de este proceso.

Por último, el proceso generador de reportes es una base importante para generar cualquier tipo de análisis relacionado con la información en la base de datos. Hoy en día puedo crear cualquier módulo, analizar cualquier requerimiento necesitado por los diferentes analistas SEO y codificar la generación automatizada de la solución.

çLa práctica de la tesina me permitió no sólo aprender sobre estudio de requerimientos, estudio de factibilidad, análisis, diseño y testing, sino también que me dio confianza para prepararme para encarar

nuevos proyectos. Hoy en día estoy convencido que, por más que no se tengan los conocimientos avanzados sobre una tecnología, el paso más difícil en cualquier proyecto es empezar. Mientras se trabaja, se va aprendiendo y rehaciendo uno mismo.

La facultad me dio los conocimientos básicos en Java y en base de datos para poder comenzar con el proyecto, pero tuve que ser autodidacta en el aprendizaje de MySQL avanzado y de diseño de patrones en programación, especialmente todo lo relacionado con la Web. La facultad da una buena base en programación, siempre y cuando el alumno esté dispuesto a aprender obviamente. Tuve la ventaja también de tener materias donde nos enseñaron en profundidad el paradigma orientado a objetos, punto más que importante en la planificación de todo el sistema en general. Estos aspectos, sumado a mi experiencia, fueron la combinación perfecta de herramientas que facilitaron que lograra mi objetivo. Pude volcar estos conocimientos y utilizar el sentido común y mi creatividad para encontrar la solución a ciertos problemas que no han sido encarados en el mercado.

Puedo concluir que, a diferencia de un sistema pago y no-libre, un conjunto de herramientas de software libre siempre estará más cerca de las necesidades del usuario final. Tener la ventaja de compartir el código, compartir los conocimientos y personalizar todo aspecto necesario, brinda una solución mayor más cercana a lo que el usuario tiene en mente. Y quizás algo más importante aun, con la posibilidad de escalar y modificar el sistema a gusto. Algo que caracteriza a las compañías americanas es la ambición. Cuando se logran objetivos, siempre se está buscando el siguiente paso para la mejoría o para un mayor ingreso. El paradigma y el sentido del software libre van de la mano con esta “filosofía de trabajo”, ya que un software de código abierto está en continuo crecimiento y cambio, y permite amoldarse a las necesidades cambiantes y a los nuevos objetivos de forma eficaz.

Mi mayor satisfacción final radica en que hoy en día, en mi empresa, se aprobó un proyecto para crear un sistema global de administración de keywords (una aplicación Web) cuyo subsistema principal será este sistema. Por lo tanto, hoy en día dispongo de capital y de un equipo calificado para seguir trabajando sobre el mismo y lograr una herramienta compleja y completa, sin tener que depender de costosas aplicaciones pagas. La empresa en donde trabajo me ha dado un voto de confianza, y estaré viajando a la sede central en EEUU para exponer mis ideas y brindar mi apoyo técnico y consultoría al equipo encargado del nuevo proyecto.

## VIII. Bibliografía

1. <http://google.dirson.com/post/3020-porcentaje-uso-google-paises/>, Segundo párrafo
2. Graphone Jennifer, Couzin Gradiva, “*Search Engine Optimization An Hour A Day*”, segunda edición, Wiley Publishing, 2008
3. Kent Peter, “*Search Engine Optimization For Dummies*”, Wiley Publishing, 2004
4. Ledford Jerri, “*Search Engine Optimization Bible*”, Wiley Publishing, 2008
5. <http://searchenginewatch.com/2167931>, 02/10/09
6. Google, “*Google’s Search Engine Optimization Starter Guide*”, <http://www.google.com/Webmasters/docs/search-engine-optimization-starter-guide.pdf>, 15/02/08
7. Rank Above, [www.rankabove.com](http://www.rankabove.com)
8. Searchlight Conductor, [www.conductor.com/searchlight](http://www.conductor.com/searchlight)
9. <http://es.wikipedia.org/wiki/Metatag>, 12/07/09
10. [www.bing.com](http://www.bing.com)
11. [www.yahoo.com](http://www.yahoo.com)
12. <http://www.webexperto.com/articulos/art/289/algunas-bases-de-link-building/>, 14/12/09
13. <http://www.trueba.com.mx/blog/2010/02/%C2%BFcomo-google-encuentra-mi-sitio-o-robots-que-indexan-resultados/>, 09/09/10
14. <http://www.google.com/support/webmasters/bin/answer.py?hl=en&answer=70897#1>
15. <http://www.google.com/support/webmasters/bin/answer.py?hl=en&answer=70897#2>
16. <http://www.google.com/support/webmasters/bin/answer.py?hl=en&answer=70897#3>
17. <http://www.entrepreneurs-journey.com/296/do-meta-tag-keywords-matter-anymore/>, 29/09/09
18. [http://www.googleguide.com/advanced\\_operators.html](http://www.googleguide.com/advanced_operators.html), 17/11/10
19. <http://www.google.com/support/webmasters/bin/answer.py?hl=es&answer=66359>, 17/11/10
20. <http://www.interacciones.com.ar/url-semantic-clean-url-friendly-url/>, 17/11/10
21. <http://akamarketing.com/content-is-king.html>, 17/11/10
22. [www.Webdeveloper.com/html/html\\_metatags.html](http://www.Webdeveloper.com/html/html_metatags.html), 04/10/09
23. <http://google.dirson.com/googlebot.php>, 10/09/10
24. <http://www.webmarketingnow.com/tips/meta-title.html>, 10/09/10
25. <http://www.forosdelweb.com/f64/hasta-cuantas-keywords-title-315486/>, 10/09/10
26. Kent Peter, “*Search Engine Optimization For Dummies*”, Wiley Publishing, 2004
27. Ledford Jerri, “*Search Engine Optimization Bible*”, Wiley Publishing, 2008
28. [http://www.w3schools.com/TAGS/tag\\_hn.asp](http://www.w3schools.com/TAGS/tag_hn.asp)
29. <http://techtastico.com/post/%C2%BFque-es-la-redireccion-301-y-como-hacerla/>, 15/09/10
30. <http://www.miempresaenlinea.com/blog/post/keywords-seo.aspx>, 15/09/10
31. <http://www.google.com/support/webmasters/bin/answer.py?hl=es&answer=66359>, 17/11/10
32. <http://www.webdesignerforum.co.uk/topic/8771-301-perm-redirect-how/>, 17/11/10
33. <http://www.seoprofesional.com/crosslink/>, 14/11/10
34. <http://es.wikipedia.org/wiki/Nofollow>, 15/11/10
35. <http://www.newbriefing.es/glosario-de-terminos-usados-en-el-marketing-en-buscadores/>, 17/11/10
36. <http://blog.linklift.es/%C2%BFque-es-el-link-juice/>, 17/11/10
37. <http://es.wikipedia.org/wiki/PageRank>, 17/11/10
38. [http://en.wikipedia.org/wiki/Social\\_media](http://en.wikipedia.org/wiki/Social_media), 17/11/10
39. [http://en.wikipedia.org/wiki/Article\\_marketing](http://en.wikipedia.org/wiki/Article_marketing)
40. <http://www.webdesignerforum.co.uk/topic/8771-301-perm-redirect-how/>, 17/11/10
41. <http://estrategias-marketing-online.com/seo-para-novatos-las-claves-para-lograr-un-buen-posicionamiento-google/>, 17/11/10
42. <http://www.aleydasolis.com/seo/arquitectura-web-seo-buscadores/>, 17/11/10
43. <http://code.google.com/intl/es-AR/apis/websearch/docs/>. 17/11/10
44. <http://www.materiablogger.com/%C2%BFquienes-te-enlazan-yahoo-site-explorer-la-mejor-herramienta-seo-gratis-del-planeta/>. 04/12/10
45. <http://articulos.nolimites.net/consultar/optimizacion-de-consultas-en-mysql-t790.0.html>, 17/11/10
46. [http://es.wikipedia.org/wiki/Interfaz\\_de\\_programaci%C3%B3n\\_de\\_aplicaciones](http://es.wikipedia.org/wiki/Interfaz_de_programaci%C3%B3n_de_aplicaciones), 17/11/10

## IX. Anexos

### Manual de usuario

#### Requerimientos previos:

Para los procesos:

- o Tener instalado cualquier “Java Runtime Environment”, de 1.6 en adelante
- o Tener configurada la variable de entorno JAVA\_HOME apuntando al archivo .bin de Java, creado en la instalación del “Java Runtime Environment” correspondiente
- o Crear los siguientes directorios:
  - o “C:/Reports”: alojará los reportes generados automáticamente por los procesos (en su modo de salida a Excel)
  - o “C:/DB Inserts”: alojará los archivos de texto con las sentencias SQL generados automáticamente por los procesos (en su modo de salida Log)
  - o “C:/Inputs”: directorio donde residirán los archivos de entrada para cada proceso. Dentro de este directorio deben residir también:
    - IONAIURLs.csv
    - MTPAIURLs.csv
    - YSEAIURLs.csv
    - LinkCheckURLs.csv
    - RankCheckerInput.csv

Para la base de datos:

- o Tener instalado cualquier servidor MySQL
- o Recomendación: instalar alguna interfaz de consultas SQL para facilitar las consultas directas a la base de datos

#### Ejecución de los procesos

Utilizando la línea de comandos todos los procesos (en Windows: CMD), con excepción de los procesos RC y RG, pueden ser ejecutados de tres maneras. Ubicándonos en la carpeta donde reside el archivo.jar:

- o “java -jar nombreDelProceso.jar DB” : ejecuta el proceso y persiste su salida en la base de datos (obviamente, la base debe estar antes instalada)
- o “java -jar nombreDelProceso.jar log” : ejecuta el proceso y persiste las sentencias SQL “Inserts” en un archivo .txt que se guardará bajo “C:/DB Inserts”
- o “java -jar nombreDelProceso.jar” : ejecuta el proceso y persiste la salida en un archivo Excel que se guardará bajo “C:/Reports”





## X. Glosario

*Anchor Text:* Es el texto que se muestra en el enlace

*Cross Linking:* Varias webs se enlazan entre ellas, todas con todas. Técnica que puede ser penalizada por los buscadores.

*Dominio:* Localización de una entidad en Internet.

*Header/Headings:* Son unos tags de HTML como por ejemplo H1> y el H2. Dan más importancia a los textos de cara a los buscadores.

*HTML (HyperText Markup Language):* Es un lenguaje de creación de webs. Actualmente el mejor visto por los buscadores.

*Inbound Link:* Es un enlace interno en nuestro website, un buen uso de ellos beneficiará a nuestra web en el posicionamiento en buscadores e incrementará nuestro link popularity.

*Keyword Density (Densidad de palabra clave):* La medición de la densidad de palabras clave es usado por los motores de búsqueda para determinar el contenido relevante dentro de cada página. Una densidad correcta de las palabras clave nos ayudará al posicionamiento de esa página por ese criterio.

*Link Building:* contruir la popularidad web (a través de links tanto internos como de otros sitios/dominios) necesaria para posicionar nuestro sitio en buscadores, ya sea la homepage del dominio, o sus documentos internos.

*Link Reciproco:* Cuando dos webs se enlazan mutuamente.

*Link Popularity:* Popularidad de un sitio web. Número de enlaces que hacen referencia a un sitio web. Cuanto mayor sea nuestro link popularity mejores resultados se conseguirán en los buscadores.

*Marketing en Buscadores:* Estrategia de marketing destinada a promocionar una Web a través de los buscadores.

*Meta Tags:* etiquetas html que se incorporan en el encabezado de una página web y que resultan invisibles para un visitante normal, pero de gran utilidad para navegadores u otros programas que puedan valerse de esta información.

*Motor de búsqueda:* sistema informático que busca archivos almacenados en servidores web gracias a su «spider» (o Web crawler o robot de búsqueda).

*Navegabilidad:* Características que hacen que sea fácil navegar una Web, pasando de una página a otra.

*Page Rank (PR):* Valoración algorítmica usada Google para clasificar las webs bajo un baremo de 0 a 10. Este valor mide la importancia que tiene una página para un buscador.

*Palabra clave:* Son las palabras para las cuales se estima aparecer en la primera posición de los resultados de Google cuando se busca por ellas.

*Ranking:* Organización de las webs por parte de los buscadores. La optimización de la web ayuda a salir en las posiciones más altas.

*Redirección 301:* El número '301 Moved Permanently' es un estado del servidor web, al igual que nos podemos encontrar con los estados '404 Not Found' o '500 Internal Server Error'. Estos estados están perfectamente definidos en los Estándares del protocolo HTTP.

**ROI:** es un porcentaje que se calcula en función de la inversión y los beneficios obtenidos para cuantificar la viabilidad de un proyecto. Se utiliza junto al VAN (valor actual neto ) y a la TIR ( tasa interna de retorno o rentabilidad ).

**Robots.txt:** Es un archivo de texto que usan los spider de los buscadores cuando visitan las webs. En el se puede indicar que partes de tu sitio son indexables y que robots pueden acceder a estas.

**Robot de búsqueda Googlebot:** Googlebot es el robot de búsqueda usado por Google. Colecciona documentos desde la web para construir una base de datos para el motor de búsqueda Google. Si un webmaster no quiere que su página sea descargada por un Googlebot, puede insertar un texto llamado robots.txt, el cual puede hacer que Googlebot (y otros bots) no investiguen una o varias páginas o incluso todo el sitio web. Googlebot tiene dos versiones, deepbot y freshbot. Deepbot investiga profundamente, tratando de seguir cualquier enlace en esa página, además de poner esta página en el caché, y dejarla disponible para Google. En Marzo del 2006, completaba este proceso en casi un mes. Freshbot investiga la web buscando contenido nuevo. Visita sitios que cambian frecuentemente. Idealmente, el freshbot visitará la página de un periódico todos los días, mientras que la de una revista cada semana, o cada 15 días. Googlebot descubre enlaces a otras páginas, y se dirige hacia ellos también, así puede abarcar toda la web fácilmente.

**SEO (Search Engine Optimization):** Optimización para los buscadores o Posicionamiento en buscadores. Proceso de modificación y optimización de un website para mejorar las posiciones en los buscadores.

**SERP:** Son las paginas de resultados que ofrece los buscadores cuando se realiza una búsqueda.

**Sitemap:** Es un fichero normalmente con extensión xml o gz (comprimido) que se aloja en el directorio raíz de nuestra web y contiene información para google de las diferentes paginas que componen todo nuestro sitio web. A diferencia de un Mapa web, este no es visible para nuestros visitantes tan solo para los buscadores.

**Spamming:** Utilización de técnicas no aceptadas por los buscadores con el fin de engañarlos y así mejorar las posiciones en el mismo. Si los buscadores lo perciben banean esa web.

**Spider:** Un robot enviado por los motores de búsqueda para catalogar websites en Internet. Cuando un robot nos visita se dice que hemos sido indexados.

**Title:** El titulo de la pagina el texto que va dentro de los tags del código fuente <title></title> justo en el Head del documento HTML. Actualmente es la etiqueta más importante de cara al posicionamiento en buscadores. Este texto se muestra en la zona alta del navegador cuando visualizamos la pagina.

**Tráfico:** El numero de visitas que recibe una web.

**URL:** Forma corta para (Universal Resource Locator). Es una dirección única que identifica un website.

**Volumen global:** cantidad de veces que esa keyword es buscada en Google ya sea mensual o semanalmente

**XML (Extensible Markup Language):** XML no es más que un conjunto de reglas para definir etiquetas semánticas que nos organizan un documento en diferentes partes.

**.htaccess:** es un simple archivo de texto que Apache usa para tener algunas reglas sobre los directorios y los archivos. Resumiendo, este contiene determinadas directivas que obligan al servidor de web a actuar según los usuarios, así como a definir otras reglas para los documentos.

