SWISH-E. Un ejemplo de colaboración en el desarrollo de software libre

José Manuel Ruiz

1. Antecedentes

En 1999 al Boletín Oficial del Estado le surgió la necesidad de ofrecer a los usuarios que accedían a través de Internet a los contenidos de los servicios web de algún tipo de herramienta de búsqueda que permitiese de una manera sencilla localizar la información.

En aquel momento el BOE disponía de una solución de búsqueda e indexación de contenidos comercial que utilizaban sus usuarios de Bases de Datos. Sin embargo, tras realizar una serie de pruebas, esta solución se mostró claramente ineficiente para satisfacer el elevado volumen de usuarios simultáneos que requeriría la explotación de unos índices de bases de datos disponibles ante decenas de miles de potenciales usuarios. Básicamente, los recursos de proceso se perdían en las siguientes tareas:

- Búsqueda en índices ineficiente.
- Lenguaje de generación de páginas dinámicas propietario, interpretado, cerrado y lento en su ejecución.
- Ordenación de resultados excesivamente lenta cuando se realizaba por algún campo distinto de la relevancia del documento. Por ejemplo: fecha de publicación, departamento emisor, rango, etc.
- Excesiva utilización de memoria por parte de los procesos de búsqueda para los recursos disponibles en los servidores en aquel momento.
- Escasa documentación de la herramienta y soporte ineficaz por parte del proveedor (revendedor del producto).

Esta situación provocaba que las consultas se solapasen en el tiempo, incrementando progresivamente los retardos en las respuestas hasta colapsar los servidores al quedar estos sin recursos para atender a nuevas peticiones entrantes. Era, pues, extremadamente sencillo provocar, incluso de manera fortuita, un ataque de denegación de servicio en los servidores.

Estos problemas, unidos a otros detectados en los procesos de indexación, desaconsejaban la utilización de la herramienta para un previsible acceso de múltiples consultas simultáneas desde Internet.

2. Solución basada en software libre

Ante la situación presentada en el punto anterior se hacía necesario utilizar otro tipo de herramienta de indexación y búsqueda de contenidos aplicable a los contenidos a distribuir por el Boletín Oficial del Estado en Internet. Dicha solución no precisaba complejos elementos de gestión de contenidos ni una funcionalidad similar a las muy costosas soluciones existentes en el mercado. Su principal requerimiento era la velocidad de respuesta en las búsquedas.

Ante la premura de tiempo, se estudiaron las posibilidades que el software libre ofrecía. En aquel momento existían varios indexadores de contenidos susceptibles de ser utilizados (freeWAIS-sf, htdig, swish-e, etc) pero ninguno de ellos cumplía plenamente los requerimientos buscados. Básicamente, el aplicativo debía cumplir unos mínimos imprescindibles:

- Alto rendimiento en búsqueda ante el volumen esperado de consultas.
- Alto rendimiento en indexación ya que los contenidos se modificaban a diario.- Soporte de campos de búsqueda de diverso tipo.
- Disponibilidad de librería de programación (API)
- Ordenación de resultados por diversos criterios (campos)
- Indexación de caracteres nacionales (ISO-8859) y soporte de tablas de conversiones
- Utilización de filtros externos para poder indexar diferentes tipos de contenidos (texto, html, pdf, etc.)

Ninguna de las soluciones basadas en software libre cumplía todos los requisitos anteriores. Sin embargo, una de ellas, aunque prácticamente abandonada en su desarrollo por sus autores, era una implementación muy sencilla de un indexador, estaba bien documentada y su licencia, basada en GPL, permitía su modificación y adaptación. El paquete en cuestión era swish-e en su versión 1.3 y podía ser un buen punto de partida al implementar un índice invertido y los procesos básicos necesarios de indexación y búsqueda.

3. Desarrollo de swish-e versión 2.0

A partir de la versión de swish-e 1.3 se procede a desarrollar y adaptar el aplicativo a los requerimientos básicos del BOE. Para ello se le añadió la funcionalidad necesaria de la que carecía:

- Nueva gestión de índices. Se añade un índice hash al índice invertido para acelerar las búsquedas.
- Nuevo motor de indexación más rápido utilizando tablas hash en lugar de las originales listas enlazadas. De esta manera se podían indexar decenas de miles de documentos en un tiempo razonable.
- Almacenamiento en los índices de las posiciones de las palabras para permitir la búsqueda de frases.
- Soporte completo de caracteres nacionales y de tablas de conversión definibles por el usuario.
- Inclusión de una librería C y de su correspondiente API. De esta manera se hace posible escribir CGIs de búsqueda en lenguaje C.
- El formato de Base de Datos nativo se hace "portable" entre plataformas hardware y software diferentes. Se puede indexar en un sistema operativo diferente al que se vaya a utilizar como explotación para las búsquedas.

Todo el desarrollo se realizó en entornos de software libre (sistema operativo linux, compilador y herramientas de desarrollo GNU gcc) y se comprobó su correcto funcionamiento en varios sistemas UNÍX, tanto comerciales (SUN solaris e IBM AIX), como libres (Linux, FreeBSD).

Básicamente el aplicativo se compone de 2 partes, un indexador y un buscador. El indexador se encarga de analizar los documentos y extraer toda la información necesaria que permita crear la base de datos de índices. Una vez obtenida esta, se puede localizar la información a través del buscador. Además del buscador se dispone de una librería abierta que permite a los usuarios crear sus propios desarrollos.

Esta estructura se mantiene practicamente idéntica en la actualidad.

Una vez realizadas las correspondientes pruebas se pone el aplicativo en explotación en el BOE y se ofrece por Internet la Base de Datos IndiBOE (sumarios del BOE desde 1995 hasta la actualidad) que incluye cuatro índices:

- Sección I (Disposiciones Generales)
- Sección II (Autoridades y Personal)
- Sección III (Otras Disposiciones)
- Sección V (Anuncios)

Dado el origen del paquete, basado en licencia GPL, con el cual aún compartía gran parte del código, y acorde a sus condiciones, se procedió a ofrecer el nuevo paquete con la misma licencia, como versión 2.0. Durante el año 2000 se corrigieron algunos errores y se añadió alguna funcionalidad menor siendo la versión 2.0.5 la última de la rama 2.0.

4. Nuevas versiones. Un proyecto en colaboración

A partir de la versión 2.0 el desarrollo del paquete cobra más interés por parte de la comunidad de usuarios y se decide ampliar su funcionalidad. En ese momento se inicia un desarrollo nuevo desde diversos partes del mundo y se comienza a ampliar notoriamente la funcionalidad del aplicativo. Para coordinar todos estos esfuerzos, se decide alojar el proyecto en sourceforge.net y, mediante la herramienta libre CVS, se procede a coordinar todas las fases del desarrollo (control de versiones, corrección de errores, incorporación de mejoras, etc). Así, se crea la versión de desarrollo swish-e 2.1 a la que progresivamente y, desde diversas fuentes, se le ha ido ampliando su funcionalidad.

He aquí una lista de las opciones más significativas que se han incorporado y que se encuentran totalmente operativas en la actualidad:

Mejora introducida	Autor
Soporte de tipos de documentos básicos (texto, html, XML básico)	JMR
Mejora de los filtros externos (inclusión de ejemplos para distintas fuentes externas como PDF, RTF, etc.)	RS
Nueva documentación	WHM
La librería C de acceso se hace reentrante	JMR
Módulo perl para facilitar la escritura de CGIs	JMR
Formato de presentación de resultados configurable por el usuario	RS
Mejora del sistema de base de datos nativo (indexación más rápido)	JMR
Mejora de la gestión de memoria	BM
Mejora del proceso de indexación (dos modos de funcionamiento: En memoria y en disco)	JMR
Mejora de la gestión de los campos	WHM
Soporte completo de documentos HTML y XML con la utilizació de librerías estándar (expat en primer lugar y libxml2 posteriormente)	WHM
Nuevo modo de indexación “prog” para permitir indexar fuentes de datos externas de manera eficiente (Por ejemplo, indexar contenidos de Bases de Datos Relacionales como Oracle).	WHM
Incorporación de ordenación de resultados en tiempo de indexación en lugar de hacerlo en la búsqueda.	JMR
Se proporcionan scripts/CGI de búsqueda de ejemplo	WHM
Mejora del módulo perl de acceso. Incorporación de forma nativa a servidor Web apache	WHM
Porting a entornos Windows	DN

WHM: Bill Mosseley

RS: Rainer Scherg

BM: Bill Meier

DV: David Norris

JMR: José Manuel Ruiz

Esta tabla da una idea muy próxima a lo que puede ofrecer el software libre: Diversas personas que no se conocen combinando su esfuerzo y conocimientos para mejorar y ampliar un desarrollo.

5. Planes de futuro

Actualmente, a muy corto plazo, los planes incluyen la liberación de lo que será la versión 2.2 que incluye todas las funcionalidades anteriormente descritas.

Como planes de futuro se encuentran:

- Nuevo gestor de índices que permita una manera más flexible de añadir documentos dinámicamente a un índice ya creado. Ello conllevará un rediseño total del sistema de base de datos nativo de la aplicación.
- Creación de un Servidor de Base de Datos que permita, entre otras cosas, llevar una cache de las búsquedas.
- Soporte de otras Bases de Datos para mantener los índices, por ejemplo Berkeley DB
- Gestión transaccional de las actualizaciones

6. Algunas cifras de rendimiento

Las siguientes cifras muestran el comportamiento del indexador y del buscador sobre un servidor Dell PowerEdge (Intel Pentium III, 2 GB RAM, 4 discos 72 GB SCSI) con RedHat Linux 7.2 como sistema operativo.

Prueba de indexación

La prueba de indexación se compone de 100000 documentos XML (codificación ISO-8859), y un volumen total de 1.56 GB. Tiempo de indexación: 21 minutos.

Pruebas de búsqueda

Se realizan varias pruebas de búsqueda sobre el anterior índice en el mismo sistema.

En todos los casos, se muestran solamente los 20 primeros resultados (la ordenación se realiza sobre el número total de resultados).

Búsqueda	Núm. resultados	Criterio de ordenación	Tiempo ejecución	Tiempo busqueda
Búsqueda de una palabra muy común (“resolucion”)	17928	Por relevancia	0.056 seg.	0.033 seg.
		Por fecha de publicación	0.066 seg.	0.043 seg.
		Por departamento	0.057 seg.	0.034 seg.
Búsqueda de una palabra poco común (“informática”)	528	Por relevancia	0.024 seg.	0.001 seg.
		Por fecha de publicación	0.029 seg.	0.006 seg.
		Por departamento	0.028 seg.	0.005 seg.
Búsqueda de una frase (“ley 30/1984”)	45	Por relevancia	0.034 seg.	0.010 seg.
		Por fecha de publicación	0.038 seg.	0.015 seg.
		Por departamento	0.037 seg.	0.014 seg.

7. Referencias: Algunos sitios donde se usa swish-e

Al estar el código fuente disponible, swish-e se encuentra funcionando en la práctica totalidad de los sistemas UNIX (Linux, FreeBSD, Solaris, AIX, HPUX, etc), Mac OSX y plataformas windows, existiendo binarios para gran cantidad de dichos sistemas.

Adicionalmente, al haber sido construido de forma totalmente abierta, el tipo de contenidos que se están indexando varia desde los clásicos PDF o HTML hasta MP3.

Como ya se ha mencionado, el Boletín Oficial del Estado utiliza extensamente dicho buscador. Las Bases de Datos que ofrece con este sistema son:

- IndiBOE: Ya mencionada, incluye 4 índices. Cada índice incluye más de 100000 documentos en formato XML. Lenguaje de programación de búsqueda utilizado en el web: perl
- Web: Documentos fuente HTML. Lenguaje de programación de búsqueda utilizado en el web: perl
- Tienda del BOE: Documentos fuente en Base de Datos Relacional. Lenguaje de programación de búsqueda utilizado en el web: PHP
- Dictámenes del Consejo de Estado: Documentos fuente en Base de Datos Relacional. Lenguaje de programación de búsqueda utilizado en el web: perl
- Jurisprudencia del Tribunal Constitucional: Documentos fuente en formato XML. Lenguaje de programación de búsqueda utilizado en el web: perl

A continuación se incluyen otros lugares significativos que utilizan swish-e:

- Apache Web Server site (http://apache.org)
- Berkeley Digital Library Sunsite (http://sunsite.berkeley.edu/cgi-bin/search.pl)
- Librarians Index to Internet (http://lii.org)

Más sitios en: http://swish-e.org/sites.html

8. Algunas referencias de software utilizado

http://swish-e.orgWeb de swish-e

http://sourceforge.net Alojamiento del proyecto. Versiones de desarrollo.

http://www.jclark.com/xml/expat.html Librería XML expat

http://xmlsoft.orgLibrería XML libxml2

http://www.gzip.org/zlib/Librería compresión zlib

http://gcc.gnu.org/Compilador C gcc

http://www.cvshome.org/Software de control de versiones CVS

http://www.foolabs.com/xpdf/XPDF. Filtro de PDF a texto