

Las Noticias libres en castellano: BarraPunto y OpenResources

Vicente Matellán Olivera, Jesús M. González Barahona,
Pedro de las Heras Quirós, José Centeno González
Departamento de Ciencias Experimentales e Ingeniería
Universidad Rey Juan Carlos
vmo,jgb,pheras,jcenteno@gsync.escet.urjc.es
<http://www.gsync.escet.urjc.es>

Resumen

Los sitios Web BarraPunto.com¹ y OpenResourceS.com² son un centro de información sobre software libre. Desde sus comienzos se diseñaron para manejar gran cantidad de información que se recibe tanto automática, en el caso de OpenResources, como “manualmente” (con intervención humana directa), como en el caso de BarraPunto. Además, el sistema de información subyacente gestiona todos los asuntos relativos a la publicidad (*accounting*, *reporting*, etc.), que son su principal fuente de financiación. La temática cubierta por estos “sitios web” hizo que se considerase el uso de software libre como opción en su construcción, resultando además ser la opción más adecuada a las necesidades del mismo. Este artículo describe las herramientas utilizadas para la construcción de estos sitios.

Entre las herramientas utilizadas para implementar el sitio de OpenResources pueden destacarse el servidor Web Apache³, el lenguaje PHP⁴ utilizado para generar dinámicamente páginas HTML, el buscador htdig⁵, la herramienta GLOBAL⁶ de generación de código HTML a partir ficheros escritos en diferentes lenguajes de programación, el gestor de foros w-agera⁷, o el generador de estadísticas analog⁸.

¹<http://BarraPunto.com>

²<http://OpenResourceS.com>

³<http://www.apache.org>

⁴<http://www.php.net>

⁵<http://www.htdig.org>

⁶<http://wafu.netgate.net/tama/unix/global.html>

⁷<http://w-agera.araxe.fr>

⁸<http://www.statslab.cam.ac.uk/~sret1/analog/index.html>

Para el sitio BarraPunto⁹ el software empleado es también libre, basado en este caso casi en su totalidad en el paquete de Slashdot¹⁰. Además se utiliza como herramienta de coordinación Zope¹¹, construido a su vez con el lenguaje de programación Python¹².

1 Introducción

Los primeros sistemas distribuidos de publicación electrónica se basaban en la publicación de información estática (o al menos de costosa actualización). Por ejemplo, *WAIS (Wide Area Information Servers)* permitía, y permite mediante el renovado servicio Z39.50, realizar búsquedas en ficheros previamente indexados; o *gopher* permitía una navegación jerárquica basada en directorios y ficheros. El *WWW (World Wide Web)* (o simplemente Web) comenzó de forma similar, como un conjunto de páginas con un formato normalizado (HTML) en el que además de la propia información podían existir enlaces a otras páginas.

En el caso particular del Web, en los primeros momentos las páginas se creaban “ad-hoc”, es decir, sin reutilizar la información ya existente en otros formatos digitales. Las herramientas básicas para la creación de estos documentos eran editores de texto adaptados para generar HTML. Cuando la información requería modificaciones (por ejemplo, las actualizaciones), se reeditaban individualmente las páginas necesarias. Todo ello suponía grandes costes de actualización, gran número de errores, y en general una falta casi absoluta de dinamicidad en los servidores de información. Aún hoy día muchos centros de información basados en Web están contruidos siguiendo este modelo.

La demanda de servicios ágiles y dinámicos, donde la información pueda adaptarse a las necesidades de cada usuario particular, donde se permita por ejemplo que el usuario del servicio no sea un mero consumidor de información sino que pueda generarla como en el caso de BarraPunto¹³, demanda nuevas herramientas como Zope¹⁴, o Slashdot¹⁵.

De la misma forma que se adapta a los usuarios, un sitio web moderno debe adaptarse también a los formatos de la información, así han aparecido diversas herramientas que combinadas permiten producir páginas HTML a partir de distintos tipos de información. Hay programas que generan HTML a partir del código

⁹<http://www.BarraPunto.com>

¹⁰<http://www.slashdot.org>

¹¹<http://www.zope.org/>

¹²<http://www.python.org>

¹³<http://www.BarraPunto.com>

¹⁴<http://www.zope.org/>

¹⁵<http://www.slashdot.org/>

fuente de casi cualquier lenguaje de programación, a partir de formatos propietarios (como MS-Word), o de formatos públicos (ASCII, LaTeX, ...). También se puede generar HTML automáticamente a partir de servicios como el correo electrónico o los grupos de USENET; generarlas automáticamente a partir de plantillas en función del tipo de usuario o del momento de conexión; integrarlas con bases de datos o en general con el sistema de información de la empresa, etc.

Por otra parte está el tipo de software, el mundo de Internet ha utilizado generalmente software de libre distribución. En el caso particular del WWW, programas como Mosaic, el precursor de los navegadores Web actuales, o el servidor Web más usado¹⁶ actualmente Apache¹⁷, se han creado bajo el paraguas del software libre¹⁸, también conocido como software Open Source¹⁹(TM) y han colaborado de forma extraordinaria al desarrollo del software libre.

En el presente artículo se describen las estructuras utilizadas para crear Open Resources²⁰ y BarraPunto²¹ que reúnen una serie de características que lo hacen a nuestro juicio ²² muy interesantes:

- **Dinamicidad**, ya que ambos están diseñados para crecer rápida y autónoma. En el caso de OpenResources²³ incorporando automáticamente información recibida de listas de correo, grupos de noticias, etc. Y el caso de BarraPunto²⁴, aceptando los comentarios de los visitantes e incorporándoles a las páginas del sitio.
- **Interactividad**, proporcionando búsquedas, permitiendo y fomentando activamente la participación de los visitantes del sitio, etc.
- **Gestión comercial**: control de visitas a las distintas zonas del sitio web, analizando su procedencia geográfica y el navegador utilizado; gestión de banners, *accounting*, etc.
- **Gestión de la apariencia**, que ha permitido independizar el contenido (la información) de su apariencia, generando las páginas dinámicamente.

¹⁶<http://www.netcraft.com/survey/>

¹⁷<http://www.apache.org>

¹⁸<http://www.fsf.org>

¹⁹<http://www.opensource.org>

²⁰<http://www.OpenResourceS.com>

²¹<http://www.BarraPunto.com>

²²Los autores de este artículo han formado parte del núcleo que promovió y desarrolló estos sitios web

²³<http://www.OpenResources.com>

²⁴<http://www.BarraPunto.com>

- **Uso de software libre**, que se ha utilizado para la realización de todo el sistema. Esto le aporta cualidades como la fiabilidad, la rápida detección de erratas (*bugs*) y la facilidad para su corrección, su coste, etc.

En la siguiente sección de este artículo se analizan las ventajas que aporta la utilización de software libre en general y en la construcción de sitios Web en particular. A continuación se presenta la arquitectura global de Open Resources²⁵ y BarraPunto²⁶. Más adelante se describen cada una de las herramientas libres que se han utilizado. Por último, se extraen las primeras conclusiones sobre el comportamiento general observado del sitio Web en los últimos meses.

2 Software libre en la publicación electrónica

Cuando se habla de “free software”²⁷ hay una peligrosa ambigüedad, debido que “free” en inglés significa tanto “libre” como “gratis”. Afortunadamente, en castellano no tenemos esta ambigüedad y el significado de “software libre” está mucho más claro. De hecho, en la comunidad software de habla inglesa se está extendiendo el término “libre software”, usando el vocablo castellano, para evitar la ambigüedad de “free”. Otro término que se está empleando cada vez con más frecuencia para este tipo de programas es el de Open Source²⁸, marca registrada por Open Source Initiative, que tiene la idea de promoverla para su uso por programas que cumplan las condiciones que consideran ha de tener el software de este tipo.

De todas formas es bueno dejar claro que el software libre no tiene porqué ser gratis. Es más, no suele serlo, o al menos no completamente. Expuestas de forma sencilla, las condiciones que tiene que cumplir un determinado producto software para considerarse libre son:

- Posibilidad de que quien recibe el software pueda adaptarlo a sus necesidades. Naturalmente, esto incluye poder hacerle mejoras, corregir erratas, aumentar su funcionalidad, y por supuesto, utilizarlo como el usuario crea conveniente. Para poder satisfacer esta condición es preciso disponer de *código fuente* del software en cuestión.
- Permiso de redistribución del software a otros usuarios, que a su vez podrán seguir redistribuyéndolo y modificándolo. Esta redistribución puede hacerse gratuitamente o mediante pago, independientemente de como se haya obtenido el software en cuestión (gratis o pagando).

²⁵<http://www.OpenResourceS.com>

²⁶<http://www.BarraPunto.com>

²⁷<http://www.fsf.org>

²⁸<http://www.opensource.com>

De ahora en adelante llamaremos, por contraposición, software *propietario* al que no cumple estas condiciones, y por tanto no es libre. La mayor parte del software que se puede comprar hoy día es propietario, aunque al tendencia parece estar cambiando rápidamente.

¿Por qué es interesante utilizar software libre a la hora de construir un sitio Web? La respuesta se puede estructurar a partir de las siguientes ventajas, que son genéricas del software libre, pero especialmente interesantes cuando se aplican al desarrollo de un sitio Web: calidad y mejora rápida del software, y simplicidad de prueba.

2.1 Calidad del software

La disposición del código fuente de un programa permite verificar realmente la *calidad* del mismo. ¿Alguna vez no se tiene la sensación al usar un programa propietario de que algo no está bien programado? Cualquiera, con los conocimientos adecuados, puede comprobarlo en el caso de programas desarrollados según la filosofía del software libre. Es más, se pueden realizar sugerencias (o encargarlas y pagarlas) a los desarrolladores del producto sobre mejoras de diseño, de optimización, etc.

De igual modo, las erratas pueden detectarse mucho antes. Esto es especialmente importante para un sistema de publicación electrónica basado en el Web, donde el mal funcionamiento de programas básicos significa la imposibilidad de acceso a la información (con el consiguiente perjuicio para los usuarios).

2.2 Mejora rápida del software

Con el modelo del software libre los programas pueden desarrollarse *incrementalmente* realizando modificaciones a programas a su vez resultado de modificaciones anteriores. Esto, unido a la cantidad de programadores que pueden llegar a involucrarse en determinados proyectos de software libre, hace que la velocidad de desarrollo de las aplicaciones libres pueda ser mucho más alta que la de las propietarias.

El desarrollo incremental del software es una de las ventajas competitivas más importantes del software libre. Ninguna empresa, por grande que sea, puede llegar a competir con los grandes grupos de programadores voluntarios (de universidades, cedidos por empresas, profesionales en sus ratos libres, etc.) que se llegan a reunir alrededor de un proyecto común. Programas como GIMP o Apache son buena muestra de ello.

Para el caso de un sitio Web como Open Resources²⁹ el uso de software de

²⁹<http://www.OpenResourceS.com>

libre distribución supone una garantía de estar utilizando software muy probado (reutilizado de productos o versiones diferentes) y a la vez de fácil y rápida actualización (las mejoras incorporadas y las erratas detectadas están disponibles inmediatamente).

2.3 Simplicidad de prueba

Cuando se usa software libre, es posible realizar pruebas con muchos programas en un corto periodo de tiempo. Se buscan en el Web, se consulta sobre ellos en listas especializadas, y se bajan de la red los que se deseen probar. Las pruebas pueden ser muy completas, e incluir modificaciones si eso es necesario, para personalizar el programa.

Una vez elegido el software más adecuado para una tarea dada, el tiempo de puesta en producción es también muy corto, y no incluye fases imprescindibles en el caso de software propietario, como la adquisición o la negociación de una licencia especial.

Este proceso tan simple y flexible permite evaluar rápidamente herramientas que de otra forma ni siquiera se considerarían. El poder hacerlo permite estar en el frente de ola tecnológica, y proporcionar rápidamente nuevos servicios. Algo que en el campo de los servicios de información en Web es absolutamente imprescindible. Por ejemplo, en el caso de Open Resources³⁰ se han evaluado herramientas como crit³¹ para añadir comentarios a documentos HTML ya existentes (otros sitios web) visibles al navegar a través de una interfaz determinada.

3 El Web de Open Resources

El público objetivo de Open Resources³² está compuesto por los programadores involucrados en proyectos de software libre, los usuarios de este software, y en general cualquier usuario informático interesado en esta forma de entender el negocio de la producción y distribución de software. Por tanto el diseño de la arquitectura debe estar enfocado a obtener un sitio Web atractivo, fácil de mantener y de ampliar.

³⁰<http://www.OpenResourceS.com>

³¹<http://www.crit.org>

³²<http://www.OpenResourceS.com>

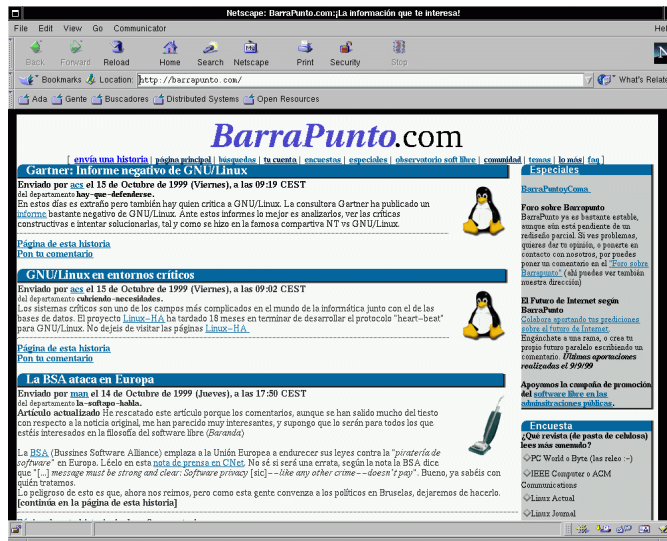


Figura 1: Página principal de BarraPunto

3.1 Diseño del sistema

El núcleo de cualquier sistema de noticias es la información que se va a publicar. En el caso de Open Resources³³ dicha información es de diferentes tipos (código de programas, artículos, mensajes de correo, etc.), está generada por diferentes tipos de autores (colaboradores diarios que escriben las noticias, regulares que escriben artículos o esporádicos que envían mensajes de correo), con diferente periodo de validez (las noticias tienen una vida más corta que los artículos, pro ejemplo). Por todo ello fue difícil elegir un formato común para toda la información.

En principio, el formato HTML podría considerarse como la opción “natural”, pues es en el que se distribuye la información. Sin embargo, elegir esta opción significaría ligar el contenido, la información en si misma, al formato. Para evitarlo se decidió usar otro formato para la información no generada de forma automática (los mensajes de correo, el código fuente, etc. utilizan sus formatos específicos y se han usado las herramientas descritas en la siguiente sección para convertirlos en HTML), que fuese estándar y pudiese utilizarse en cualquier plataforma. Estas restricciones descartaban cualquier formato binario, dejando solo los formatos basados en ASCII. Sin embargo, el ASCII no permitía dotar a los documentos de estructura (distinguir títulos, secciones, etc.) por lo que se barajaron diversas opciones: SGML, LaTeX, etc. Al final, la elección fue LaTeX por la existencia de

³³<http://www.OpenResourceS.com>

herramientas como GSyC-doc³⁴ que facilitaban su conversión en HTML de forma muy controlada y flexible.

De esta forma y como se muestra en la Figura 2 el centro del sistema es el servidor de páginas HTML, Apache³⁵ que se encarga de enviar las páginas a los navegadores de los visitantes.

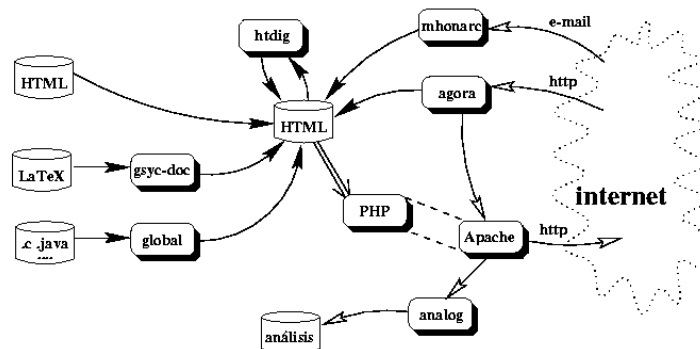


Figura 2: Relaciones entre las herramientas usadas

En la figura se pueden apreciar las diferentes fuentes de información del sistema que convergen en el HTML que el servidor enviará a Internet:

HTML: ficheros con código directamente escrito en HTML como por ejemplo la gestión de los banners de los mayoristas de publicidad.

LaTeX: los artículos, noticias y en general toda la información generada por los trabajadores o colaboradores de Open Resources. Usando la herramienta GSyC-doc³⁶ estos ficheros se convertirán en HTML.

Código fuente: en diferentes lenguajes (como C o Java), que serán convertidos en ficheros HTML por la herramienta GLOBAL³⁷.

E-mail: los mensajes de correo electrónico, recibidos de las lista de corre a las que está suscrito el sistema, se convierten también en páginas HTML usando la herramienta mhonarc³⁸.

³⁴<http://www.gsync.inf.uc3m.es/gsync-doc>

³⁵<http://www.apache.org>

³⁶<http://www.gsync.inf.uc3m.es/gsync-doc>

³⁷<http://wafu.netgate.net/tama/unix/global.html>

³⁸<http://www.oac.uci.edu/indiv/ehood/mhonarc.html>

http: es el nombre del protocolo que se usa para transferir las páginas HTML desde el servidor hasta el navegador del usuario. Dicho protocolo soporta también el traslado de información en sentido contrario, lo cual se usa en Open Resources³⁹ para realizar los foros de debate.

A partir de estas fuentes se genera la información que se almacenará en el sitio Web. Lo usual es alimentar con ella directamente al servidor, en este caso Apache⁴⁰. Es decir, el conjunto de información generado suele ser directamente el que se muestra en Internet.

Sin embargo, para hacer más modificable la estructura de las páginas, insistiendo en la idea de separar contenidos de presentación, se decidió generar las páginas finales dinámicamente. Para ello se parte de unas estructuras HTML ya construidas (cabeceras, menús, publicidad, etc.) en las que se inserta la información. Para hacerlo se empleó un lenguaje de programación PHP⁴¹ diseñado con ese objetivo.

También hay que realizar un seguimiento del sistema, para lo que se utilizará una herramienta de análisis de los ficheros históricos del servidor Web.

Por último, la gestión de publicidad se realiza con dos mecanismos básicos:

- Todas las páginas del sitio se sirven mediante una “plantilla” escrita en PHP, que incluye, además del código HTML correspondiente a cada página, llamadas a programas que insertan la publicidad dinámicamente. Estos programas están escritos también en PHP, y permiten funcionalidades típicas como rotación de anuncios o selección del anuncio a mostrar según la sección.
- Para llevar cuenta de las veces que se sirve cada anuncio se utilizan los ficheros históricos. Básicamente, se utiliza el mismo software para llevar la contabilidad de las páginas servidas y para llevar la contabilidad de los anuncios servidos. A partir de esta información se puede facturar a los anunciantes, y contrastar con sus propios ficheros históricos (cuando los tienen).

En la siguiente sección se describen en más detalle las herramientas citadas.

4 Software usado en OpenResourceS.com

Open Resources⁴² se ha construido con la restricción de ser un sitio de alta disponibilidad, con pocas caídas y capaz de servir a gran cantidad de usuarios, elevada.

³⁹<http://www.OpenResourceS.com>

⁴⁰<http://www.apache.org>

⁴¹<http://www.php.net>

⁴²<http://www.OpenResourceS.com>

Las herramientas informáticas utilizadas para conseguirlo se describen en los siguientes apartados.

4.1 GNU/Linux

GNU/Linux⁴³ es el sistema operativo libre de mayor difusión, siendo probablemente el segundo sistema operativo más utilizado en ordenadores personales tras MS-Windows. Es un sistema operativo tipo Unix, multi-tarea y multi-usuario. Además, dispone de una ingente cantidad de software disponible, desde el específicamente dedicado a los servicios de Internet hasta el software tradicional en ordenadores personales (hojas de cálculo, editores, etc.)

Entre las distribuciones de GNU/Linux existentes en la actualidad, se ha utilizado Debian⁴⁴, una de las más habituales, muy estable, y que es la que incluye el mayor número de paquetes software.

4.2 Apache

El proyecto Apache⁴⁵ es un esfuerzo para desarrollar y mantener un servidor HTTP para varios sistemas operativos modernos, como UNIX o Windows NT. Su objetivo es proporcionar un servidor seguro, eficiente y extensible.

La historia de Apache⁴⁶ comenzó en febrero de 1995. Entonces el servidor Web más popular en Internet era el demonio de HTTP desarrollado por Rob McCool en el *National Center for Supercomputing Applications (NCSA)*, en la Universidad de Illinois en Urbana-Champaign.

Sin embargo, su desarrollo se había paralizado por lo que un grupo de administradores (utilizando el correo electrónico) se pusieron de acuerdo para coordinar los “parches” que iban realizándose. Utilizando como base NCSA httpd 1.3 añadieron algunos de los parches más habituales, los probaron en sus servidores y realizaron la primera versión oficial de Apache⁴⁷ (la 0.6.2) en Abril de 1995. Después de un completo rediseño, la migración a numerosas plataformas y una serie de pruebas extensivas e intensivas, en Diciembre de 1995 vio la luz la versión 1.0 de Apache.

En menos de un año se convirtió en el servidor más utilizado en Internet. Apache⁴⁸ se ha convertido en el servidor Web más popular de Internet desde su

⁴³<http://www.linux.org>

⁴⁴<http://www.debian.org>

⁴⁵<http://www.apache.org>

⁴⁶<http://www.apache.org>

⁴⁷<http://www.apache.org>

⁴⁸<http://www.apache.org>

aparición en 1996. En Enero de 199 la estadística de Netcraft⁴⁹ afirma que el 53% de los servidores Web de Internet usan Apache⁵⁰ (el 58% si se añaden sus derivados) lo que le hace ser el servidor más usado (más que todos los demás juntos).

Open Resources⁵¹ utiliza Apache⁵² como servidor y por tanto constituye el núcleo central del software utilizado, el alimentan o con el que se enlazan el resto de los elementos descritos en esta sección.

4.3 PHP

PHP⁵³ es un lenguaje de tipo “script”, integrado con HTML, que se ejecuta en los servidores HTTP (por ejemplo en Apache) y del que existen intérpretes para múltiples plataformas.

Según la encuesta de Netcraft⁵⁴ en Febrero de 1999 se usaba en 345549 dominios y en 101140 direcciones IP. Tiene la particularidad de integrarse de forma muy eficiente con Apache⁵⁵ en forma de módulo.

El sitio Open Resources⁵⁶ genera las páginas HTML dinámicamente, utilizando PHP⁵⁷ para construirlas a partir de plantillas pre-configuradas. De esta forma la gestión de la apariencia, de la publicidad, etc. se puede realizar de forma independiente de los contenidos.

4.4 GSyC-doc

GSyC-doc⁵⁸ es el nombre que engloba un conjunto de herramientas escritas en Perl⁵⁹ para convertir diversos documentos, esta fundamentalmente pensada para documentos escritos en LaTeX⁶⁰, en documentos HTML. Estas herramientas proporcionan mecanismos para enlazar los documentos, configurar su apariencia, actualizarlos parcialmente manteniendo la coherencia de los enlaces, etc.

⁴⁹www.netcraft.com/survey/

⁵⁰<http://www.apache.org>

⁵¹<http://www.OpenResourceS.com>

⁵²<http://www.apache.org>

⁵³<http://www.php.net>

⁵⁴www.netcraft.com/survey/

⁵⁵<http://www.apache.org>

⁵⁶<http://www.OpenResourceS.com>

⁵⁷<http://www.php.net>

⁵⁸<http://www.gsysc.inf.uc3m.es/gsysc-doc>

⁵⁹<http://www.perl.org/>

⁶⁰<http://www.latex-project.org>

GSyC-doc⁶¹ se ha desarrollado y usado sobre la distribución Debian⁶² de GNU/Linux⁶³, aunque debería funcionar (probablemente sin modificaciones) sobre la mayoría de los sistemas operativos (especialmente los tipo-Unix).

GSyC-doc⁶⁴ se basa en las siguientes herramientas:

- GNU Make.
- LaTeX2HTML⁶⁵.
- Perl⁶⁶.
- TeX (y LaTeX). Usando la distribución teTeX, aunque cualquier distribución moderna de TeX y LaTeX debería servir.

El formato básico de la información generada en Open Resources⁶⁷ es LaTeX, por lo que herramienta GSyC-doc⁶⁸ constituye una parte fundamental a la hora de generar y configurar el HTML que se distribuye finalmente a través del servidor.

4.5 htdig

htdig⁶⁹ es un sistema completo para indexar y realizar búsquedas sobre un pequeño dominio o sobre una intranet. Este sistema no está pensado para reemplazar los sitios de búsqueda e indexación tradicionales de Internet como Lycos⁷⁰, Infoseek⁷¹, Webcrawler⁷² o AltaVista⁷³. Está pensado para cubrir las necesidades de búsquedas en una empresa, en el servidor de una universidad o incluso sobre solamente una determinada sección de un sitio Web.

htdig⁷⁴ se desarrolló en la universidad de San Diego (EEUU) (de ahí parte de su nombre) como mecanismo para buscar en los varios servidores del campus.

⁶¹<http://www.gsync.inf.uc3m.es/gsync-doc>

⁶²<http://www.debian.org>

⁶³<http://www.linux.org>

⁶⁴<http://www.gsync.inf.uc3m.es/gsync-doc>

⁶⁵<http://cbl.leeds.ac.uk/nikos/tex2html/doc/latex2html/latex2html.html>

⁶⁶<http://www.perl.org/>

⁶⁷<http://www.OpenResourceS.com>

⁶⁸<http://www.gsync.inf.uc3m.es/gsync-doc>

⁶⁹<http://www.htdig.org>

⁷⁰<http://www.lycos.com>

⁷¹<http://www.infoseek.com>

⁷²<http://www.webcrawler.com>

⁷³<http://www.altavista.com>

⁷⁴<http://www.htdig.org>

4.6 global

GLOBAL⁷⁵ es un sistema de etiquetado de código fuente que funciona de la misma forma sobre diversas plataformas.

Los lenguajes de programación que soporta son C, Yacc y Java. Una de sus funcionalidades más útiles es que permite localizar una función determinada entre distintos ficheros de código fuente, mostrando el código con una sola pulsación de ratón.

Además permite localizar elementos (tanto su definición como referencias a ellos) rápidamente, detectar repetidos, tratar un árbol de código fuente con sus subdirectorios como una única entidad, manejar expresiones regulares en las búsquedas (POSIX 1003.2), etiquetar el código, etc.

Uno de los fundamentos del software libre es precisamente poder acceder al código fuente. Open Resources⁷⁶ quiere facilitar el acceso al código fuente de cualquier programa libre, dándole formato con GLOBAL⁷⁷.

4.7 w-agera

w-agera⁷⁸ es un programa escrito en PHP⁷⁹ para instalar, gestionar y configurar foros de debate basados en Web. Permite instalar foros, BBS, libros de visitas y todos los elementos relacionados, permitiendo además adaptarlo con facilidad para distribuir informaciones, anuncios, difundir actualizaciones de software, publicar FAQs, permitir que los usuarios carguen información en el servidor, etc.

w-agera⁸⁰ constituye el mecanismo básico de interacción de los visitantes de Open Resources⁸¹, permitiéndoles añadir noticias, comentar los artículos, hacer sugerencias, etc.

4.8 mhonarc

mhonarc⁸² es un programa escrito en Perl⁸³, lo que hace que pueda usarse en multitud de plataformas, para convertir mensajes de correo electrónico (RFC 822 y MIME) en HTML. Además mhonarc⁸⁴ puede traducir carpetas o *mailboxes* tipo

⁷⁵<http://wafu.netgate.net/tama/unix/global.html>

⁷⁶<http://www.OpenResourceS.com>

⁷⁷<http://wafu.netgate.net/tama/unix/global.html>

⁷⁸<http://w-agera.araxe.fr>

⁷⁹<http://www.php.net>

⁸⁰<http://w-agera.araxe.fr>

⁸¹<http://www.OpenResourceS.com>

⁸²<http://www.oac.uci.edu/indiv/ehood/mhonarc.html>

⁸³<http://www.perl.org/>

⁸⁴<http://www.oac.uci.edu/indiv/ehood/mhonarc.html>

Unix en código HTML, eliminar o añadir mensajes a ficheros HTML creados por mhonarc⁸⁵, generar índices de los mensajes, utilizar filtros para los mensajes, etc. Todo ello además con la facilidad de adaptar su presentación.

Open Resources⁸⁶ ofrece como uno de sus múltiples servicios la posibilidad de consultar algunas de las listas de correo más útiles para la comunidad del software libre, permitiendo a sus visitantes navegar por ellas en formato HTML, para lo cual se emplea mhonarc⁸⁷.

4.9 analog

analog⁸⁸ es un programa para analizar los ficheros históricos de servidores Web. De esta forma se puede analizar que páginas son las más populares, de que países proceden los visitantes, quienes han tratado de seguir enlaces que no existen y desde dónde, etc. Además los resultados pueden visualizarse a través de un visor de HTML, como muestra la Figura 3.

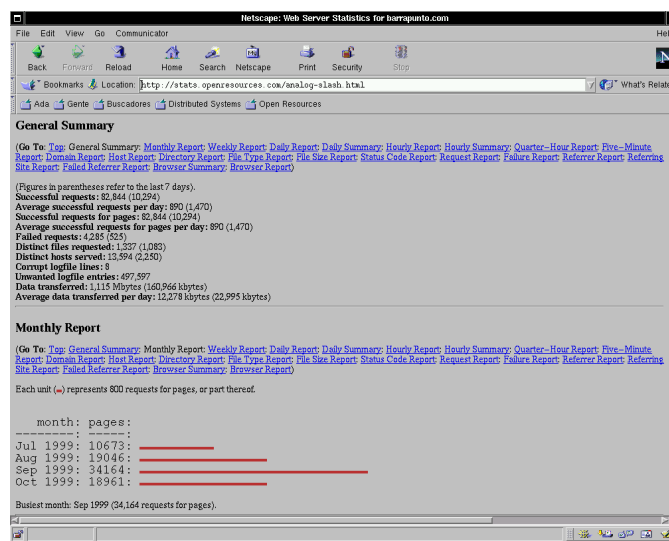


Figura 3: Estadísticas de visitas a BarraPunto

analog⁸⁹ es un software muy configurable (permite elegir entre más de 27 tipos de informes en 26 idiomas distintos) y se puede usar en multitud de plataformas

⁸⁵<http://www.oac.uci.edu/indiv/ehood/mhonarc.html>

⁸⁶<http://www.OpenResourceS.com>

⁸⁷<http://www.oac.uci.edu/indiv/ehood/mhonarc.html>

⁸⁸<http://www.statslab.cam.ac.uk/~sret1/analog/index.html>

⁸⁹<http://www.statslab.cam.ac.uk/~sret1/analog/index.html>

aparte de GNU/Linux (Windows (3, 95 y NT), DOS, Mac, diferentes tipos de Unix (TM), etc.) y entiende el formato de varios servidores Web (Microsoft IIS, Netscape, WebSTAR, Netpresenz) aparte de los formatos estándares (NCSA, W3, etc.).

5 El Web de BarraPunto

El objetivo de BarraPunto⁹⁰ es crear un sitio en español equivalente a Slashdot⁹¹, el cual constituye uno de los puntos de referencia en la información en tecnologías de la información en los Estados Unidos. La idea es sencilla: quién tenga algo que contar lo envía mediante un formulario al sitio web, que lo publica en forma de título y breve comentario en la página principal, generando además una página para el resto de la noticia y los posibles comentarios. Además se almacena en una base de datos. Si alguien quiere comentar alguna noticia su comentario se añade en la página de la noticia que se quiere comentar.

Actualmente existen diversas alternativas en el dominio del software libre para implementar un sitio de este tipo. Una de ellas es el software del propio Slashdot⁹², otro podría ser el de Squishdot⁹³, basado en Zope⁹⁴ un paquete de software realizado en Python⁹⁵ para desarrollar sitios webs de altas prestaciones.

Puesto que el objetivo era realizar un sitio gemelo al de Slashdot⁹⁶, se optó por utilizar su software.

5.1 El Software de BarraPunto

El secreto del éxito de Slashdot⁹⁷ es la participación de los visitantes que proporcionan la información, la comentan, la discuten, en resumen la convierten en algo vivo. Este formato tiene en su propio éxito el mayor de sus problemas: la cantidad de contribuciones puede llegar a hacer difícil discriminar cuales son más relevantes, para facilitar esta tarea el software del sitio aporta un sistema de “moderación” que permite a ciertos autores, aquellos que han demostrado una especial fidelidad con el sitio (aportaciones en forma de noticias, comentarios, etc.) o de especial relevancia, puntuar las noticias y de esta forma los usuarios puede elegir

⁹⁰<http://www.BarraPunto.com>

⁹¹<http://www.slashdot.org>

⁹²<http://www.slashdot.org>

⁹³<http://www.squishdot.org>

⁹⁴<http://www.zope.org>

⁹⁵<http://www.python.org>

⁹⁶<http://www.slashdot.org>

⁹⁷<http://www.slashdot.org>

ver sólo las noticias o comentarios que han alcanzado cierta puntuación, los más valorados por cierto comentarista, etc.

Por otra parte, el aspecto de la página que cada usuario ve al acceder al sitio puede personalizarse no sólo en el contenido, sino en el aspecto. Así cada usuario puede elegir que “cajas” se le van a mostrar en “su” página. Estas cajas contienen resúmenes de información obtenida de otros sitios web, buscadores, etc. Toda la personalización se realiza mediante “cookies” que permiten conservar el estado configurado por el usuario.

El software en si mismo está realizado en Perl⁹⁸ y puede obtenerse directamente del sitio de Slashdot⁹⁹. La base de datos que utiliza el sistema es MySQL.

6 Funcionamiento del sistema y conclusiones

La apariencia del sistema OpenResourceS¹⁰⁰ aparece en la Figura 1. Se puede apreciar que su aspecto no muestra en ningún caso la cantidad de software utilizado para su creación. Sin embargo, si lo muestra la facilidad con la que puede ser modificado, ampliado y mantenido.

En el caso de BarraPunto¹⁰¹ se puede apreciar el parecido actual con Slashdot¹⁰², que es intencional, pues el objetivo es conseguir que lectores hispanohablantes de Slashdot¹⁰³ identifiquen BarraPunto¹⁰⁴ como su “hermano” en español y colaboren en su despegue. El hecho de disponer del código fuente, facilita enormemente la previsión de su modificación a corto plazo, en cuanto se le considere suficientemente maduro.

El número de visitas ha este sitio Web está creciendo de forma apreciable desde su lanzamiento en Julio de 1999 (la tabla 1) resume dicho crecimiento (datos del 14 de Octubre). Ello demuestra que la aproximación elegida, software libre solamente, es *viable*, el sitio realmente se ha podido construir; *eficiente*, con un simple PC se espera atender cientos de peticiones concurrentes; *robusto*, como demuestra el tiempo de funcionamiento sin caídas; y desde luego *adaptable*, pues tanto la organización física, como el aspecto gráfico pueden ser modificadas con suma facilidad.

Aunque la decisión inicial de utilizar software libre fue motivada fundamentalmente por la temática de ambos sitios Web, la decisión ha demostrado ser la ade-

⁹⁸<http://www.perl.org>

⁹⁹<http://www.slashdot.org>

¹⁰⁰<http://www.OpenResourceS.com>

¹⁰¹<http://www.BarraPunto.com>

¹⁰²<http://www.slashdot.org>

¹⁰³<http://www.slashdot.org>

¹⁰⁴<http://www.BarraPunto.com>

Mes	Páginas
Julio 1999	10673
Agosto 1999	19046
Septiembre 1999	34164
Octubre 1999	18961

Tabla 1: Páginas servidas por el servidor de BarraPunto

cuada. Toda la funcionalidad requerida se ha podido realizar usando software libre. En muchos casos, de hecho, puede decirse que ha podido ser realizada *gracias* al software libre, ya que no existen herramientas propietarias adecuadas a muchas de las tareas. Por otra parte, el nivel de flexibilidad de uso de las herramientas libres ha permitido el adaptarlas completamente a las necesidades del sitio. En general, puede decirse que el sitio Web se ha beneficiado extraordinariamente del uso de software libre.