

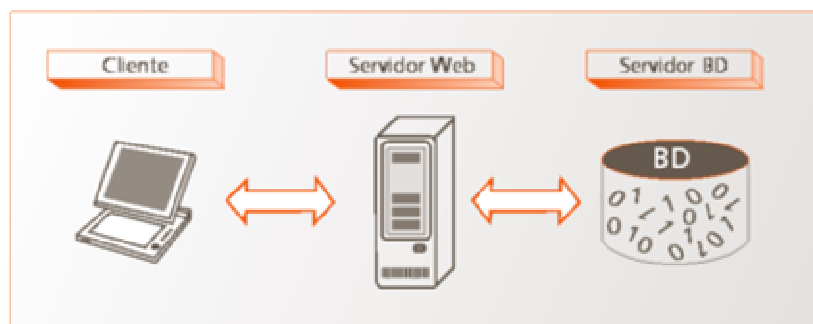
## ¿Portales Dinámicos o Estáticos?

### Introducción

En la actualidad, no se concibe un portal que no requiera de una gestión de contenidos dinámica, pero se hecha de menos esa rapidez de respuesta que ofrecen las webs estáticas. Como casi siempre ocurre, las mejores soluciones son las intermedias, las que extraen las mejores características de todas las posibilidades. Pero antes que nada vamos a intentar esgrimir los problemas con los que nos enfrentamos en ambos casos, y buscaremos a partir de este análisis cual es la solución más óptima.

### Portales dinámicos

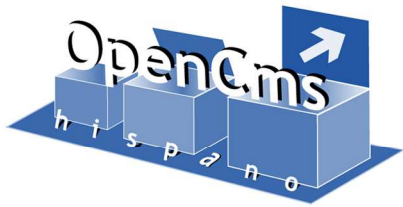
De todos es sabido que una arquitectura normal de un portal dinámico es la siguiente:



En ocasiones, y en portales con poca carga de trabajo, el servidor de Base de Datos puede compartir máquina con el servidor de aplicaciones, aunque esta arquitectura suele ofrecer problemas de rendimiento, sobre todo, cuando usamos gestores de Base de Datos que requieren una gran cantidad de recursos. Por esta razón, la solución adoptada en la mayoría de los casos es tener un servidor de aplicaciones en una máquina independiente de la máquina de la Base de Datos.

En esta arquitectura empiezan a surgir otros tipos de problemas, y es que todos los accesos a la Base de Datos se hace a través de la red, y pueden ocurrir que:

- La base de datos esté caída, y por tanto, no devuelva ninguna información.
- La red se encuentra saturada debido al elevado tráfico de información en un momento determinado, y que los acceso a la Base de Datos se retrasen por esta causa.
- El número de conexiones máximas se supere y no tengamos más hilos de comunicación que ofrecer, y que por tanto se rechace la petición.



Todos estos problemas entre otros menos recurrentes, pueden ocasionar que nuestro portal se caiga o sea lento a la hora de responder a las peticiones. Esto causará una total desconfianza de nuestro portal, ya que los visitantes verán que es una web poco profesional y no se pasarán más por ella, provocando un descenso del número de visitas.

### **Portales Estáticos**

En el caso de un portal estático, la información está escrita en disco, por ejemplo en HTML, por tanto, nuestro servidor web sólo tiene que acceder al fichero y devolverlo tal cual. No requiere por tanto ningún acceso a Base de Datos, por lo cual es el sistema más rápido. El problema radica en la actualización de información, que requiere de personas especializadas y de un coste en tiempo elevado.

### **Soluciones**

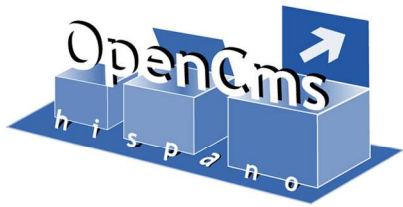
Una posible solución es hacer que nuestro portal esté completamente en el servidor de aplicaciones, de tal forma que toda la información sea accesible sin necesidad de acceder a la Base de Datos. Es decir, apostar por la filosofía de las webs estáticas. Pero debemos solucionar su principal problema, la actualización de la información, como comentamos antes.

Existen diversas opciones para solventar este problema, en este artículo os quiero comentar una posible solución válida para OpenCms.

En las últimas versiones de OpenCms, exactamente a partir de la 6.0, se integra una de las aplicaciones de indexación y búsqueda más extendida, Lucene. Aprovechando el potencial de esta herramienta, podemos mantener todo el contenido del portal indexado, de forma que, a la hora de acceder a la información en nuestros portales, no necesitemos ese pesado acceso a la Base de Datos, sino una rápida consulta a Lucene.

El sistema funcionaría de la siguiente forma:

- Edición de contenidos: Cuando editamos un nuevo contenido y lo publicamos, dicho contenido debe ser indexado en su totalidad con Lucene. En este punto es inevitable realizar accesos a la Base de Datos, pero este punto lo realizan los administradores, o los editores del portal, que tienen acceso a solucionar posibles problemas de comunicación con la Base de Datos.



- Acceso a la información: Cuando un visitante realiza una petición, se ejecutará la jsp correspondiente, y para acceder a la información realiza consultas a Lucene que contiene todo lo necesario. De forma que en ningún caso la JSP necesita acceder a la Base de Datos, pero en este diseño deberán estar exportadas en discos completamente y ser totalmente independientes de OpenCms, es decir, no podemos utilizar objetos de OpenCms que requieran tenerlo lanzado.

### **Conclusiones**

Desarrollar un portal siguiendo estas pautas puede ser costoso a la hora de desarrollarlo, ya que, hay que ser muy cuidadoso con la programación, pero consigue un portal bastante estable y sobre todo, mejoramos el tiempo de respuesta a la hora de resolver las peticiones de los usuarios.

De esta manera hemos analizado las ventajas de ambos sistemas, y hemos conseguido extraer las características principales de cada una para unir las en una solución intermedia.