

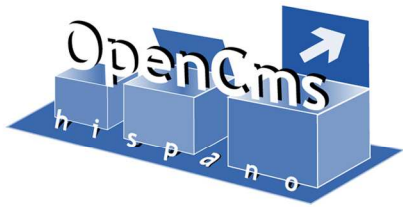
CMS, Repositorios y Gestores de Portales.

En el mundo de la programación estamos acostumbrados a que la mayoría de los avances que se realizan vayan orientados a simplificar el desarrollo de proyectos. Así, la aparición de nuevos frameworks diseñados para facilitarnos tareas es cada vez más frecuente, como por ejemplo, Spring, Hibernate, EJB3 (Enterprise JavaBeans) o JPA (Java Persistente Api) cada uno de estos especializado en una tarea, en la lógica de negocio o en la persistencia de datos entre otros.

Lo mismo ocurre con las entornos de desarrollo, productos que hacen que la tarea de programación sea más cómoda, como Eclipse y sus innumerables pluggins, u otros IDEs como JBuilder, JDeveloper, o el mismo NetBeans. Cada vez están más extendidos y prácticamente no se concibe la idea de programar sin usar alguna de estas herramientas.

Algo parecido ocurre con las aplicaciones para la realización de proyectos Web. El uso de Gestores de Contenido o CMS es cada vez mayor, pero también de otro tipo de herramientas como son los Gestores Documentales o Repositorios de Documentos y los Gestores de Portales. Podemos considerar estas tres “utilities” como el pilar en el que se sustentan este tipo de proyectos. Pero, ¿tenemos clara la funcionalidad básica que nos ofrece cada uno de ellos? Es complicado realizar una distinción entre estos tres tipos de herramientas o de clasificar los productos de los que disponemos en el mercado según el tipo de “utility.” Ya que en la mayoría de los casos, un producto suele contener funcionalidad básica de cada uno de ellos.

En este artículo, se pretende aclarar un poco la diferencia entre ellos, comentando que se espera y que tareas les corresponde a cada uno de ellos.

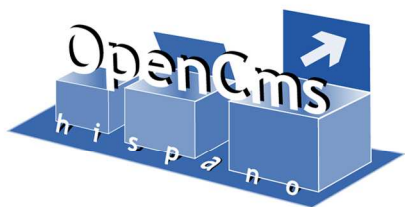


Gestores de Contenido o CMS

Cuando hablamos de Gestores de Contenido o CMS (Content Management System) hacemos referencia a aquellas herramientas que nos permiten gestionar todo el contenido de un portal, crear un contenido, editarlo, borrarlo o publicarlo. entre otras funciones.

Las características principales que deben tener son:

- Interfaz gráfica fácil de utilizar y lo más completa posible que permita realizar todas las operaciones sobre el contenido y con la ayuda necesaria para que cualquier persona, sin necesidad de amplios conocimientos de la herramienta, pueda usarla con el mínimo tiempo de aprendizaje posible.
Esta característica es quizás una de las más importantes o al menos, lo que es seguro, la más visible de todas ya que en la mayoría de los casos un simple vistazo de la interfaz gráfica puede deducir mucho del producto. Tecnologías como javascript o ajax, se han impuesto haciendo que las herramientas sean más intuitivas mejorando su claridad y facilidad de uso.
- Editores para introducir el contenido. Son muy utilizados los editores WYSIWYG (what you see is what you get) en los que vemos directamente el resultado de la edición. Por ejemplo, si marcamos una palabra en negrita, veremos dicha palabra en negrita en el momento de realizar dicha acción; y los formularios específicos para cada tipo de contenido, en los que se presentará un campo para cada elemento. Si disponemos de un contenido de tipo noticia introduciremos la información a través de un formulario que tendrá como campos el título, la introducción, el cuerpo, el autor, enlaces asociados, entre otros posibles campos útiles, de forma que el editor solo rellenando esos campos, genere el contenido.
- Motor de Workflow. La gestión del flujo de trabajo es una parte importante de la gestión del contenido. La posibilidad de separar las tareas a realizar entre distintos roles de personas es muy utilizada en estos casos ya que esta problemática se repite en gran parte de los proyectos web. Para poner un ejemplo de su utilidad, imagine un proyecto para una revista digital, en la que existirán redactores y responsables. El primer grupo de usuarios podrán editar un contenido, y el segundo publicarlo haciendo visible a todos los usuarios el contenido. Estas herramientas deben ofrecer la posibilidad de avisar a un



responsable de que tiene un contenido que revisar y de publicar, si lo estima oportuno, o de devolver el flujo al redactor para que modifique algún aspecto del contenido.

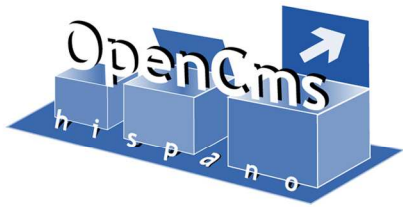
- Gestión de permisos de usuarios y grupos. Otro apartado importante de estas herramientas es la gestión de usuarios y grupos de usuarios, y los permisos de los que cada uno dispone. Es muy importante asegurar que un contenido es editado y publicado solo por aquellas personas que tienen permisos para ello. Usando el mismo ejemplo del punto anterior, en la revista digital es posible que existan distintas secciones, y que un redactor solo pueda editar en una o en varias de ellas. Igualmente tendremos responsables de secciones, que no deben poder publicar un contenido que no es de su sección. Para toda esta casuística, un CMS debe ofrecer mecanismo para realizar este tipo de control.
- Búsqueda del contenido. Ofrecer una herramienta para buscar información según una serie de palabras claves puede llegar a ser muy importante en portales, o en sitios donde la cantidad de información es tan elevada que ni siquiera una buena organización en secciones y categorías nos asegura el poder encontrar una información deseada en un momento dado.

Gestor Documental o Repositorio de Documentos

En lo referente a un Gestor Documental o Repositorio de Documentos, debemos tener en cuenta que su función principal es almacenar documentos, ya sean del tipo que sean, XML, texto, PDFs, Doc, imágenes, entre otros.

Las funciones principales que deben ofrecernos son:

- Garantizar la persistencia de la información, ya sea en Base de Datos, o en un sistema de ficheros.
- Permitir una estructura coherente según el modelo del problema que tengamos. Suele ser muy útil y flexible usar una estructura en árbol ya que este nos permite crear tantos niveles de información como necesitemos, sin estar atados a una estructura por defecto. Volviendo al ejemplo de la revista digital, supongamos que debemos organizar la información por secciones, y dentro por tipo de contenidos, problema al que una estructura en árbol da solución completa.
- Control de versiones. Una función igualmente importante que las anteriores es la necesidad de disponer de un control de las distintas versiones que se han producido en un contenido. Estas versiones e información adicional como quien



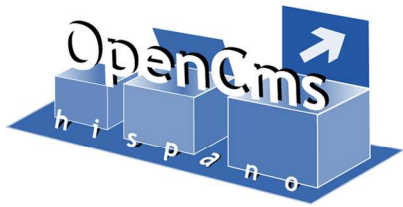
realizo la modificación (igualmente creación y borrado) y en que momento la realizó, nos puede ayudar a la hora de recuperar información que se ha modificado por error, o simplemente para temas estadísticos.

- Indexado de documentos. El mantener indexado el contenido que se tiene almacenado en el repositorio puede hacer que las búsquedas sobre el contenido sean bastante más rápidas que si las realizamos sobre toda la base de información. Suele ser buena costumbre tener estos índices en disco ya que la velocidad de acceso y de tratamiento es bastante mayor que si tuviéramos que acceder remotamente a otro servidor, o a una Base de Datos.

Gestor de Portales

El último tipo de herramientas orientada al mundo del desarrollo web, son los Gestores de Portales. Son aquellas que nos ofrecen utilidades para mostrar la información a los visitantes de la web. A continuación mostramos algunas de las funcionalidades que podemos encontrar en los gestores de portales:

- Creación de menús para organizar los contenidos. Utilidades con las que, un administrador pueda crear o borrar una sección nueva, en el portal, y que esta aparezca en el portal en el sitio y orden que le interese. Posibilidad de menús verticales, horizontales, desplegados, de dos niveles, ...
- Administrar la apariencia del portal, de forma que un administrador pueda cambiarla completamente: colores, logotipos, tipo de letras, ..., sin necesidad de tener ningún tipo de conocimiento de programación o de diseño.
- Procedimiento para que podamos crear paginas con listado de contenido sin mucho esfuerzo, y que se obtengan con una rapidez razonable que se exige en este tipo de proyectos.
- Componentes adicionales como encuestas, votación de recursos, comentarios del contenido, foro de discusión, listas de correos, calendario con información.
- Sistema de autenticación de usuarios para dar acceso restringido a ciertas zonas de nuestro sitio web.



La unión hace la fuerza

Estos tres tipos de herramientas están muy ligadas entre sí. ¿Para que nos serviría tener un potente sistema de gestión de contenidos, si no tenemos como guardarlo, o no tenemos como mostrarlo?. De aquí que entre ellos se necesiten mutuamente, ya que cada uno se utiliza en un nivel distinto.

Podríamos verlo fácilmente con la siguiente equivalencia:

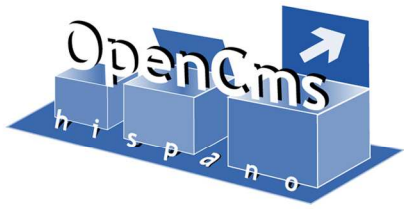
Capa de Presentación	-	Gestores de Portales
Capa de Lógica de Negocios	-	Gestores de Contenido
Capa de Persistencia	-	Gestores Documentales

Esta equivalencia no es del todo cierta, ya que, considerar que los gestores de portales solo se encargan de presentar los contenidos y que no poseen nada de la capa de negocios, no es cierto. No obstante sí puede que este esquema nos permita ver mejor la diferencia entre los tres tipos de herramientas.

Llegado a este punto, parece clara la fuerte dependencia que existe, aunque hay posibles excepciones según necesidades del proyecto. Por ejemplo, es posible que haya que instalar un repositorio de documentos, el cual se gestionará desde herramientas externas sin necesitar una herramienta para su edición, o para su presentación.

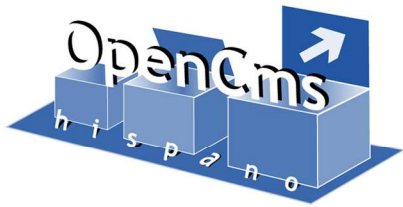
Problema existente

Uno de los problemas que nos podemos encontrar en esta rama del software es la gran cantidad de productos a medida existentes para implementar las distintas utilidades comentadas previamente. La ausencia de estándares que unificara criterios provocó que cada uno de estos productos implementara sus propias interfaces y vías de comunicación. Este camino convirtió la interconexión de distintas herramientas en una tarea prácticamente imposible, al igual que el mantenimiento por personas ajenas al desarrollo inicial.



OpenCmsHispano | La Comunidad de OpenCms en castellano





Actualmente disponemos de algunos productos que implementan la especificación JSR-168. Uno de los más usados, y de los más simples es Apache Pluto, pero existen muchos vendedores de gran caché que ofrecen implementaciones comerciales de un contenedor de portlets, como son IBM, Oracle, o BEA Systems.

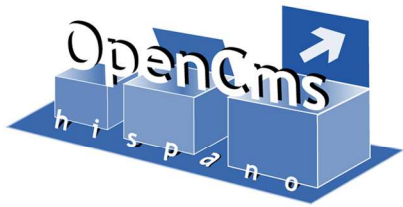
Igualmente, disponemos en el mercado gestores de portales que soportan este estándar como son Apache Jetspeed-2 Enterprise Portal, JBoss Portal, o Liferay Portal.

Para el desarrollo de estos componentes web o portlet existen algunos frameworks que nos facilitan la tarea de programarlos siguiendo el estándar. Un ejemplo es PORTIONS (portlet actions), desarrollado íntegramente en España lo que hace que tengamos toda la documentación en castellano, sin lugar a dudas, una ventaja para todos los hispano-parlantes. Para implementar PORTIONS se ha buscado la mayor similitud posible con Struts (framework de desarrollo para aplicaciones Web ampliamente extendido) con la idea de facilitar al conjunto de desarrolladores acostumbrados a trabajar con Struts el paso a este nuevo framework para el desarrollo de portlets.

Donde situamos OpenCms

OpenCms, en el marco donde mejor lo podemos encuadrar es en el de herramienta CMS. Su potencial radica en la amplia posibilidad de trabajar con el contenido, crearlo, estructurarlo, editarlo, manipularlo, etc. Pero también implementa una forma propia de almacenar la información, en este caso no se ha apostado por la integración del estándar JCR para el almacenamiento de información. Por lo tanto, en un momento dado, y llegado a un amplio desarrollo de éste, puede quedarse atrás, perdiendo así a hegemonía que actualmente tiene entre las herramientas gestoras de contenido basadas en código java.

Como es lógico, OpenCms no cierra las puertas a la integración con el estándar, sólo espera el momento oportuno para realizar dicha migración ya que un cambio como este afecta prácticamente a todo el core del producto. Por lo tanto, en la futura versión OpenCms 7, aún no podremos disfrutar de la posibilidad de integrar el entorno gráfico de OpenCms, con herramientas externas como JackRabbit, o Alfresco. Confiemos en que en el futuro Alkacon Software apueste por este desarrollo.



OpenCms nos ofrece también algunas herramientas para la gestión de la parte de presentación del portal que nos facilitan esas tareas, como la posibilidad de enlazar contenidos con sus plantillas. En cualquier caso no son del todo suficiente para considerar que OpenCms es también un gestor de Portales, ya que se necesita un importante tiempo de desarrollo al igual que suficientes conocimientos java para conseguir presentar el contenido al exterior. Esta limitación se puede ir solventando cada vez más con el desarrollo de módulos orientados a estas tareas. Es quizás una de las apuestas más interesantes para el futuro desarrollo de OpenCms.

En lo que respecta a la integración de OpenCms con un contenedor de portlets, tenemos que es una solución posible, aunque poco aconsejable, debido a la gran cantidad de modificaciones del core que hay que realizar, haciendo que las actualizaciones a posteriores versiones del producto no sean posible, y que el tiempo de desarrollo sea bastante alto, y por tanto, se disparen los costes de producción.

Autor: Sergio Raposo Vargas

serrapos@gmail.com; administrador@opencmshispano.com

Administrador de OpenCmsHispano (<http://www.opencmshispano.com>)