

Capítulo I: Estado del arte

1 – Introducción:

Desde el nacimiento de Internet, la red de redes se ha ido expandiendo hasta abarcar prácticamente todos los aspectos de nuestra vida cotidiana. En la actualidad, buscamos información de cualquier tipo, nos comunicamos a través del correo electrónico o la mensajería instantánea, mantenemos el contacto con amigos y conocidos gracias a las numerosas redes sociales, realizamos transferencias y trámites burocráticos, jugamos, trabajamos, escuchamos música, vemos vídeos e incluso películas. La lista de cosas que cualquier persona puede hacer es interminable.

En sus inicios, la red no era ni tan versátil ni tan asequible, el acceso a la misma estuvo reservado solo a un número determinado de máquinas. En la actualidad prácticamente cualquier máquina con una mínima capacidad de procesamiento puede tener acceso a Internet. Los ordenadores personales desde hace ya tiempo no son los únicos dispositivos desde los cuales se puede acceder a la red.

Paralelamente al desarrollo de Internet se ha ido desarrollando la tecnología móvil. Los terminales móviles, a medida que evolucionaban se iban haciendo cada vez más potentes y la velocidad de transferencia de datos mayor, posibilitando en mayor o menor medida el acceso a redes de datos en busca de algún determinado servicio o contenido.

El precio de estos accesos siempre ha limitado su uso, ya que se trataban de servicios tremendamente caros. Por ejemplo, con la tecnología GSM se obtenía tan solo una velocidad de 9'6 kbps, con el coste de una llamada convencional. A precios de 1995 esto suponía que el descargar una página web o un correo de 225Kb suponía 3 minutos de espera y un coste aproximado de 200 ptas.

Actualmente, gracias al uso de redes móviles más rápidas y a la cada vez más popular tarifa plana de datos, el uso del acceso web se está extendiendo enormemente.

Uno de los problemas que supone el uso masivo de acceso web mediante terminales móviles es la heterogeneidad de los mismos. Al tratarse de un éxito comercial, los fabricantes inundan el mercado con diversos modelos, cada uno con unas determinadas características y capacidades como pueden ser; resolución de pantalla, número de colores de la misma, acceso a red GSM/UMTS, reproducción de MP3, MP4, GPS y un largo etcétera.



Imagen 1: Variedad de dispositivos móviles

Asimismo las características soportadas de los navegadores de cada uno de los dispositivos móviles pueden diferir enormemente de un modelo a otro, por esta razón la visualización del contenido de una página web puede llegar a ser muy distinta dependiendo de si accedemos desde un PC o desde un dispositivo móvil. Incluso puede ser bastante diferente entre 2 modelos de terminales supuestamente muy parecidos.

Esto puede ser un problema para las web que no tengan en cuenta esta consideración. Una visualización incorrecta o incómoda del contenido al acceder desde un terminal móvil puede tener como consecuencia una disminución tanto del número de visitas como de los ingresos debidos a las mismas.



Imagen 2: Visualización no adaptada desde un móvil (contenido cortado)

Para fomentar los accesos web desde dispositivos móviles, existen varios mecanismos para ofrecer contenidos de forma adaptada a los mismos. Es posible tener copias preparadas del contenido adaptadas para distintos modelos de dispositivos, pero tener que replicar cada uno de los contenidos puede resultar una alternativa costosa e ineficiente. No obstante el problema de la optimización sigue estando presente por la gran variedad de dispositivos en el mercado, y tener

una copia por terminal no es la solución.

Afortunadamente existen herramientas que pueden solucionar este problema, usando la adaptación dinámica del contenido en tiempo real, a medida que se vaya necesitando.

En el presente proyecto se ofrecerá una solución. Se creará una plataforma para el acceso web desde dispositivos móviles, en la que se cubrirán aspectos como la creación de un portal web, el uso de un servidor que se encargue de la adaptación del contenido y el desarrollo de módulos para el portal en los que, haciendo uso de todas estas herramientas, añadan funcionalidad al mismo.

2 – Evolución de la capacidad en las redes de comunicaciones móviles

En su concepción inicial, las redes de comunicaciones móviles fueron pensadas para ofrecer únicamente servicios de voz de baja calidad, ya que la tecnología existente no hacía posible otra cosa; la red era muy cara y de pobres características, los terminales muy simples, pero pesados y costosos y el uso de la red extremadamente caro.

A medida que las redes de comunicaciones fueron evolucionando, la capacidad de transferencia de voz y datos fue aumentando. Debido a los nuevos servicios que se iban ofreciendo y al abaratamiento progresivo tanto de terminales como de las tarifas fue popularizándose poco a poco.

2.1 – Primera generación (1G):

La primera generación de redes de comunicaciones móviles surge sobre los años 80.

Una de las principales características de esta generación es la ausencia de una normativa internacional que regule los protocolos de comunicación ni que defina las características que debían soportar los equipos y terminales móviles. Tampoco se establecía un sistema de comunicación entre distintas redes, por lo que no existía la posibilidad de roaming ni entre distintas operadoras ni entre países.



Imagen 3: Teléfono móvil de primera generación

Las bandas de frecuencias usadas tampoco estaban reguladas internacionalmente. En EEUU se usaban sistemas AMPs a 800Mhz, mientras que Europa se optó generalmente por sistemas TACS en la banda 900/1800Mhz.

Otra característica importante de esta generación es que se trataba de sistemas analógicos, por tanto para poder mandar datos era necesario modular la señal, obteniéndose tasas de datos de hasta 2400bps, lo que prácticamente restringía su uso a aplicaciones de información sobre alarmas.

2.2 – Segunda generación (2G):

Estos sistemas aparecen en 1992. Su objetivo era resolver las principales carencias de las redes de primera generación; mejorar la velocidad de transmisión/recepción de datos y crear un estándar único, para permitir la intercomunicación entre redes y posibilitar que un mismo terminal funcione correctamente en las distintas redes existentes.

A pesar de los esfuerzos para conseguir un estándar único mundial se desarrollaron diversas tecnologías:

- GSM (Global System Mobile): este sistema fue desarrollado en Europa y Asia. Usaba las bandas de frecuencias de los 900/1800Mhz.
- DAMPS y NADC/USDC: Son estándares americanos que trabajan a frecuencias entorno a los 800Mhz.
- CDMA ONE: Fue un estándar surgido también en EEUU, pero de carácter propietario. Este estándar fue el verdadero competidor de GSM y sentó las bases de la telefonía de tercera generación.
- PDC: Fue la alternativa desarrollada en Japón para esta generación, trabajando en la banda de frecuencias de los 900/1500Mhz.

Con el desarrollo de estas tecnologías, nos alejamos del objetivo de unificación de normas y estándares. No obstante, se consiguió que GSM fuera un estándar a nivel europeo y la posibilidad de roaming entre distintos operadores y países.



Imagen 4: Teléfonos móviles de segunda generación

Entre las mejoras generales de estas tecnologías se pueden destacar:

- Mejora de la calidad de las comunicaciones por voz. Los estándares de esta generación mejoraron solo un poco la calidad de las conversaciones de voz, ya que es una característica que el usuario final suele aceptar.
- Mejora de la seguridad. Al tratarse de comunicaciones digitales, no es tan fácil escanear conversaciones. Además es posible que las mismas estén encriptadas en función del PIN del usuario. La tarjeta SIM proporcionaba también más seguridad en el acceso.
- Mejora de la velocidad de transferencia de datos. Los nuevos terminales alcanzaban velocidades efectivas de hasta 9600bps.

Aunque las frecuencias en las que trabajaban las distintas tecnologías eran prácticamente las mismas, era necesario el uso de pasarelas para la interconexión entre las distintas redes. También existían terminales tribanda que incluso tenían implementadas varias tecnologías a la vez.

A pesar de que la mejor tecnología era CDMA ONE, la más popular fue GSM, con más de un 70% del mercado mundial. Los terminales GSM funcionaban bien, y además existía una fuerte competencia alrededor de ellos, que propició la fabricación de una gran variedad de terminales y un abaratamiento del precio de los mismos.

Por esta época nacieron numerosos servicios, como el SMS, el cambio dinámico de células, el roaming y empezó a popularizarse el uso de Internet y de las comunicaciones multimedia, en general.

El servicio de Internet requería una velocidad de conexión mayor a la disponible, por ello surgen diversas alternativas.

- Una de ellas es WAP, que permitía una especie de “Internet móvil” aunque muy rudimentario y bastante caro.
- Algo más tarde surge el I-mode, que usa un lenguaje muy parecido a HTML, pero modificado para teléfonos móviles cuyo resultado era más parecido a las páginas web como las conocemos hoy en día.

La falta de contenidos disponibles con estas tecnologías fueron las responsables de frenar su avance. Además, era necesaria una tasa de transferencia mayor para fomentar su uso.

2.3 – Generación 2.5:

Esta generación no implica un cambio drástico en la infraestructura de red desplegada y sirve de puente, como solución más o menos provisional, hasta la llegada de las redes de tercera generación.

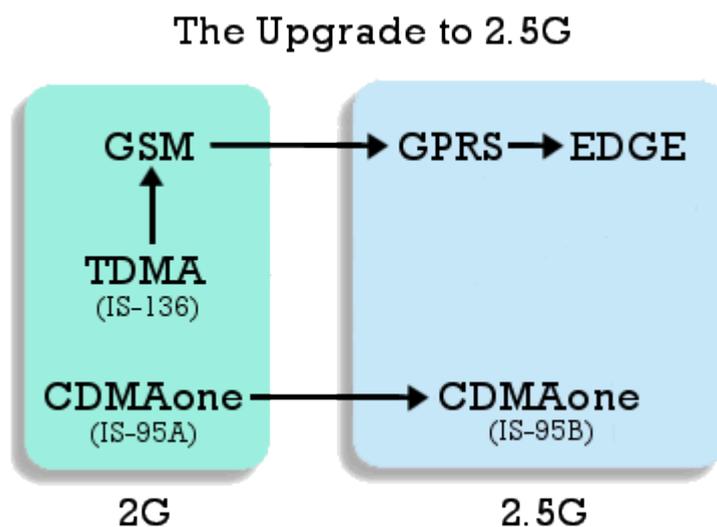


Imagen 5: Actualización de 2G a 2.5G

Básicamente las tecnologías desarrolladas tratan de aprovechar todos los recursos de la red para conseguir una mayor tasa de transferencia de datos.

Las principales tecnologías desarrolladas son:

- HSCSD (High-Speed Circuit-Switched Data):

Datos en conmutación de circuitos en alta velocidad. Esta tecnología era una de las fáciles de implementar. Se basa en el uso simultáneo de 3 interfaces GSM, por lo que permite el triple de capacidad. Al tratarse de conmutación de circuitos implica la reserva de 3 canales de voz, por lo que la tarificación es por tiempo de conexión, no por cantidad de datos.

- GPRS (General Packet Radio Service):

Servicio general de paquetes vía radio. En este caso, para transmitir los datos se siguen usando los canales de voz, pero sin reservarlos. De esta forma se puede aprovechar al máximo la red, puesto que al tener prioridad la voz sobre los datos, éste no interferirá sobre el anterior y podrá aprovechar los canales de voz ociosos para proporcionar una mayor tasa de transferencia. La tarificación en este caso es por cantidad de datos, no por tiempo, aunque las operadoras suele cobrar algo para evitar que se esté siempre conectado sin transmitir información. La tasa máxima de transferencia puede llegar hasta los 144 kbps. con esta tecnología.

- CDMA ONE (Code division multiple access ONE):

Acceso múltiple por división de código. Esta tecnología siguió desarrollándose paralelamente a las anteriores, por lo que cada vez iba obteniendo mayores velocidades de conexión.

2.4 – Generación 2.75

Posteriormente al sistema GPRS, aparece una mejora del mismo que se puede considerar como la generación 2.75, ya que hace de puente entre el 2.5G y el 3G.

EDGE (Enhanced data rates for GSM evolution) Datos avanzados para la evolución de GSM. Esta tecnología es una mejora del sistema GSM. Gracias a un cambio en la codificación permite velocidades de hasta 384 kbps en modo paquete, usando prácticamente la misma infraestructura de red. No obstante el cambio no es inmediato y se necesitan una serie de actualizaciones por parte del operador, por lo que es una tecnología que no ha llegado a implantarse.

2.5 – Tercera generación (3G):

Para continuar con la evolución de las capacidades de los teléfonos móviles iba siendo cada vez más necesario un salto tecnológico. Nuevamente se intentó aprovechar este hecho para establecer un verdadero estándar a nivel mundial.

La llegada del 3G se retrasó en gran medida por razones económicas. En muchos casos los operadores no habían amortizado aún sus redes de segunda generación. El despliegue de la tecnología era muy caro y los costes del nuevo espectro de frecuencia necesario, muy altos. A esto se le unió el alto coste de los terminales y el consumo de los mismos, que agotaban enseguida sus baterías.



Imagen 6: Móviles 3G

La ITU-T desarrolló la recomendación IMT-2000, donde se definen los requisitos que se deben cumplir para que se consideren tecnologías de tercera generación.

Entre estos requisitos estaba el de garantizar una alta tasa de datos. Se debían permitir velocidades de hasta 2Mbps, aunque con una alta dependencia de la velocidad física a la que se esté desplazando el terminal:

Categoría	Velocidad física del terminal	Tasa de transferencia
Movilidad limitada	Menos de 10 Km/h	2 Mbps
Movilidad completa	Entre 10 Km/h y 120 Km/h	384 Kbps
Movilidad alta	Más de 120 Km/h	144 Kbps

Tabla 1: Velocidades de transferencia en 3G

También se decidió que la tecnología empleada para la codificación fuera WCDMA, proponiendo la banda de frecuencias entorno a los 2Ghz, aunque más tarde se añadieron las bandas de 450, 800, 900, 1800 y 1900 Mhz.

Uno de los grandes cambios que supuso esta tecnología era la pérdida del concepto de reserva del canal de voz. En UMTS toda la información, ya sea voz o datos, es empaquetada y transportada por una red IP, lo que permite un mejor aprovechamiento de la capacidad.

Aunque el esfuerzo para crear una normativa fue importante, existían varios estándares. Cada uno apoyado por su propio organismo de normalización y estandarización.

- CDMA 2000
- UMTS
- UWC-136HS

El estándar dominante ha resultado ser la combinación GSM/UMTS, debido a que GSM era dominante y permitía el GPRS como un paso intermedio al UMTS, ya que permitía reutilizar parte de la infraestructura tecnológica, al tratarse de un sistema de conmutación de paquetes.

Por otro lado, los terminales con GSM/UMTS fueron los que dominaron el mercado, gracias al previo uso de GSM y a que los terminales eran los más económicos.

2.6 – Generación 3.5

A pesar del incremento de la capacidad de transferencia de información obtenida con la 3G, se han seguido desarrollando tecnologías para aumentarla aún más.

El HSDPA (High Speed Downlink Packet Access o Acceso descendente de paquetes a alta velocidad) cambia únicamente la interfaz radio, pero permite velocidades de hasta 7Mbps de descarga.

2.7 – Generación 3.75

La tecnología HSUPA (High-Speed Uplink Packet Access o Acceso ascendente de paquetes a alta velocidad) es una mejora del sistema anterior, ya que incrementa en gran medida la capacidad de subida del mismo, factor que generalmente ha estado en segundo plano. Con HSUPA es posible obtener hasta 2Mbps en el enlace ascendente.

2.8 – Cuarta generación (4G)

Actualmente no existen estándares que definan lo que será la cuarta generación de telefonía móvil, pero se espera que conlleve a una convergencia de las redes de cables con las inalámbricas, que permitan mantener una calidad de servicio entre los extremos y obtener velocidades de acceso entorno a 100Mbps.

3 – Servicios de Internet en dispositivos móviles

El desarrollo de las capacidades de los terminales y el progresivo auge de Internet fomentaron la creación y el desarrollo de servicios similares pero orientados a móviles.

Aunque el acceso directo a Internet desde los dispositivos móviles era posible, la limitación de la potencia de cálculo, memoria, capacidad de almacenamiento, tasa de transferencia de información y coste de la comunicación hicieron prácticamente inviable el acceso tal y como se hacía desde ordenadores personales.

Debido a lo anterior y al auge de todo lo relacionado con lo multimedia, surgen diversos “acercamientos” para emular estos servicios.

3.1 – WAP

El WAP (Wireless Application Protocol), o Protocolo de Aplicaciones Inalámbricas, nace como respuesta a la necesidad de un “Internet Móvil”. Se trata de un estándar abierto internacional para aplicaciones que hacen uso de las comunicaciones inalámbricas.

En dicho estándar se especifica cómo debe ser el entorno de aplicación y los protocolos a seguir para acceder a través de los dispositivos móviles a los servicios que se ofrecen.

Es un servicio que debe ser activado por el operador, y cuya tarificación suele realizarse por Kb descargado o por tiempo.

La oferta de contenidos e información disponible es la que se haya preparado exclusivamente para este sistema, ya sean versiones para WAP de páginas o portales web, ya existentes o creados únicamente con este fin.

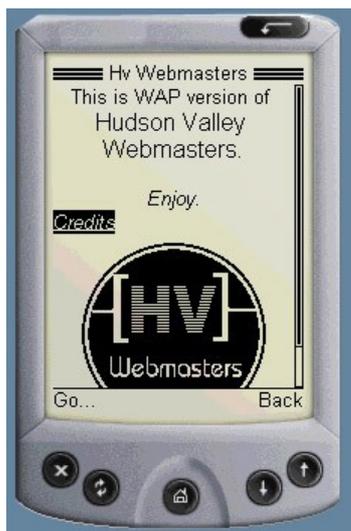


Imagen 7: Servicio WAP

La primera versión de WAP se definió en el año 2000 y la pila de protocolos es totalmente incompatible con la de Internet, lo que obligaba a usar pasarelas entre el terminal y el servidor de contenidos. Esta versión no tuvo demasiado éxito debido a que cuando apareció la tecnología disponible era lenta (9600bps), la conexión costosa, visualmente muy pobre, ya que únicamente ofrece contenido en monocromo y a los numerosos problemas de seguridad.

3.2 – WAP 2.0 y dominio .mobi

A partir de 2004 aparece WAP 2.0 para móviles de nueva generación, que permiten tasas de transferencia mayores. Esta nueva versión de WAP corrige casi todos los defectos de la anterior. Utiliza XHTML-MP y WCSS, un subconjunto de XHTML y CSS, con lo que gráficamente supone un gran avance al permitir colores y diferentes estilos. Otro punto importante es la compatibilidad con los protocolos de Internet, por lo que dejó de ser necesario el uso de pasarelas para acceder a la web.



Imagen 8: Captura de un portal .mobi

Poco después, con el objetivo de fomentar esta tecnología, la ICANN (Internet Corporation for Assigned Names and Numbers) aprobó el dominio .mobi. Este dominio de nivel superior está restringido a dispositivos móviles y es imprescindible que sea compatible con XHTML-MP. Esto garantiza que cualquier dispositivo con WAP 2.0 pueda acceder sin problemas a las páginas en ese dominio.

3.3 – I-mode

El i-mode es un conjunto de tecnologías y protocolos diseñados para la navegación web de dispositivos móviles.



Imagen 9: Navegación en i-mode

Este sistema fue ideado en 1999 por la compañía NTT DoCoMo para Japón, donde tuvo un gran éxito comercial. En España se adoptó por parte de Telefónica Móviles, pero no obtuvo tanta aceptación.

A diferencia de WAP 2.0 y el .mobi, se trata de una red paralela a Internet, en la que existen numerosos servicios; desde acceso a webs de todo tipo, servicios de e-mail, noticias, juegos e información hasta e-ticketing, generalmente controlados por parte de la operadora.

Los protocolos usados y los lenguajes en los que se basan son muy parecidos a los de Internet, no obstante son simplificaciones de los mismos, con el objetivo de hacerlos aptos para dispositivos con no mucha capacidad de procesamiento.

Al ser una red propietaria tiene las desventajas de limitación del operador para acceder a la misma, pero es más fácilmente controlable por su parte.

3.4 – Otras formas de acceso a Internet

Los sistemas citados anteriormente proveen de acceso a Internet, ya sea a una versión limitada del mismo o a una red distinta pero con servicios y contenidos similares. No obstante existen numerosas formas de acceder directamente a la red de redes.

El operador normalmente dispone de pasarelas propias que dan servicio a Internet de forma directa a través de las tecnologías que permita el terminal móvil, como GPRS, EDGE o UMTS.

Otra forma alternativa de acceso a Internet desde un dispositivo móvil es mediante el uso de tecnologías que incorporen los dispositivos, pero sobre las que el operador no tenga ningún tipo de control al no viajar los paquetes de datos a través de su red. Ejemplos de este tipo de tecnologías son WIFI y WIMAX

3.4.1 – WIFI

La tecnología WIFI (Wireless Fidelity) es un sistema de comunicaciones inalámbricas que permite la interconexión de equipos o dispositivos dentro de una red de ámbito local.

Este sistema está estandarizado con las normas de la serie 802.11 del IEEE. La velocidad de acceso dentro de la propia red (no confundir con velocidad de acceso a Internet) y las frecuencias usadas dependen de la norma concreta que se implemente. La más usada es la 802.11g, de 2003, que trabaja a una frecuencia de 2.4 Ghz y es capaz de obtener tasas de transferencia de hasta 54Mbps, dependiendo de la calidad de la señal que reciban los equipos.



Imagen 10: Conexión a Internet mediante WIFI

En principio esta tecnología no provee por sí misma acceso a Internet, ya que es el punto de acceso el que debe poseer dicha conexión. En dicho caso, para que un dispositivo móvil tenga acceso a Internet basta con que se conecte correctamente a dicho punto de acceso, que realizará funciones de enrutador de paquetes.

3.4.2 – WIMAX

WIMAX (Worldwide Interoperability for Microwave Access) es una tecnología de acceso radio

de las llamadas “de última milla”. Con la que se suprime la necesidad de instalar cableado desde un determinado domicilio hasta el proveedor de acceso telefonía fija y/o Internet, por lo que es perfecta para dar servicios de banda ancha en zonas donde el despliegue de cable o fibra presentan un coste por usuario muy elevado, debido a la baja densidad de población, como ocurre en las zonas rurales.

Esta tecnología está normalizada, formando parte de la familia de normas 802.16 del IEEE. Existen principalmente 2 variantes:

802.16d – Acceso fijo: Existe un enlace radio entre el lugar donde se encuentre el usuario y la estación base que le ofrece acceso. Permite obtener hasta 70Mbps dependiendo de la distancia, pudiendo llegar ésta hasta los 50km. Los espectros de frecuencias más usados son los 2.5 y 3.5Ghz. El principal problema del acceso fijo a Internet desde un dispositivo móvil es que no se permite el desplazamiento del terminal durante la conexión, no obstante es una posibilidad que algunos dispositivos móviles implementan.

802.16e – Movilidad completa: Con esta norma permite el desplazamiento del usuario de un modo similar al que se puede dar en GSM/UMTS. Las velocidades máximas teóricas son en este caso 144Mbps de bajada y 35Mbps, aunque actualmente apenas existen dispositivos en el mercado que implementen este estándar.

4 – Portal Web

Un portal de Internet es un sitio web, que sirve de partida a un determinado usuario, para obtener algún tipo de información o contenido, o para realizar alguna otra función. Dicha función puede consistir simplemente en la búsqueda de información de algún tipo en la red, ya sea en directorios, foros u otro tipo de medio. Podría decirse que el objetivo de un portal es ofrecer al usuario, de forma fácil e integrada, el acceso a una serie de recursos y de servicios. Principalmente están dirigidos a resolver necesidades específicas de un grupo de personas o de acceso a la información y servicios de una institución pública o privada.

Según el público objetivo al que esté dirigido, podemos clasificar los portales en 3 tipos:

Horizontales: Son portales masivos o de propósito general, se dirigen a una audiencia amplia, tratando de llegar a toda la gente ofreciendo búsqueda general de información, noticias, foros, etc.

Verticales: Son portales especializados que ofrecen un tipo de contenido concreto, como pueden ser portales de empleo, inmobiliarias y música.

Diagonales: Se trata de una combinación de los anteriores y suelen estar relacionados con la red 2.0. Suelen ser redes sociales generalistas, pero que ofrecen aplicaciones y contenidos para un público muy concreto.

4.1 – Portlets

En ciertos portales web desarrollados con Java es posible crear componentes para extender la funcionalidad de los mismos. Estos módulos son lo que se denominan Portlets.

4.1.1 – Qué es un portlet

Los portlets son componentes Web hechos en Java, que están manejados a través de un contenedor de portlets que procesa las peticiones de los clientes y produce contenido dinámico. Dicho contenido no es más que código HTML, XHTML, WML, etc. que sigue ciertas reglas.

A diferencia de los servlets, los portlets no tienen interacción directa con los clientes web ni tienen una URL asociada. En su lugar, los clientes web interactúan con el portal a través de un mecanismo de solicitud/entrega aplicado por un contenedor de portlets el cual también maneja el ciclo de vida de los mismos.

En un portal web que use esta tecnología, las páginas están compuestas por portlets. Cada uno de ellos puede aportar una determinada funcionalidad (portlet de registro, foros, búsquedas, etc.) o mostrar una determinada información (el tiempo, noticias, etc.).

Al cargarse la página en cuestión, estos módulos generan un determinado código. Dicho código formará parte de la página final mostrado al usuario.



Imagen 11: Portlet con la funcionalidad de calendario

4.1.2 – Especificación JSR-168

Antes de la existencia de esta especificación cada proveedor de portlets tenía que crear una versión distinta de cada portlet de acuerdo con la API que usara el portal en el que se quería usar, con las consecuentes desventajas en cuanto a coste y usabilidad.

La Java Portlet Request 1.0 fue desarrollada como la especificación JSR (Java Specification Request) 168. Fue creada por la JCP (Java Community Process), una comunidad internacional que coordina el desarrollo de numerosos estándares de Java e implementaciones de referencia.

La meta principal del JSR 168, es habilitar la interoperabilidad entre portlets y portales. Esta

especificación define el contrato entre el portlet y el contenedor de portlets, y coloca un conjunto de APIs de portlets que se encargan de la personalización, presentación y seguridad. La especificación también define el cómo empaquetar portlets en aplicaciones de portlets, incluyendo servlets y JSP.

También se define el ambiente de ejecución para los portlets y provee mecanismos para almacenar los datos transitorios y persistentes de los mismos.

La industria de TI ha aceptado ampliamente la JSR 168. Todas las grandes compañías en el campo de portales son parte del grupo de expertos del JSR 168: Apache, ATG, BEA, Boeing, Borland, Broadvision, Citrix, EDS, Fujitsu, Hitachi, IBM, Novell, Oracle, SAP, SAS Institute, Sun Microsystems, Sybase, TIBCO y Vignette. La lista de patrocinadores es mucho mayor.

4.1.3 – Especificación JSR-286

La JSR-286 es la versión 2.0 de Java Portlet Specification. Está desarrollada según el JCP y creada en alineación con la versión 2.0 de WSRP (Web Services for Remote Portlets, o Servicios Web para Portlets Remotos). Esta JSR fue desarrollada para mejorar las deficiencias de la versión 1.0 de la especificación JSR-168. Algunas de las características más importantes son:

- Comunicación entre portlets a través de eventos.
- Renderización de parámetros públicos.
- Los portlets sirven contenido generado dinámicamente de forma directa.
- Los portlets sirven datos de AJAX o JSON de forma directa.
- Se introducen filtros y “listeners” en los portlets.

4.1.4 – Ciclo de vida de un portlet

Como se mencionó antes, el ciclo de vida de un portlet está controlado por el contenedor de portlets, que es el elemento que define cómo se carga, instancia, inicializa, maneja las peticiones y se pone fuera de servicio.

En estas fases se hace uso de los siguientes métodos de la interfaz `Portlet`, que contienen toda la funcionalidad del mismo .

- `init()`
- `processAction()`
- `render()`
- `destroy()`

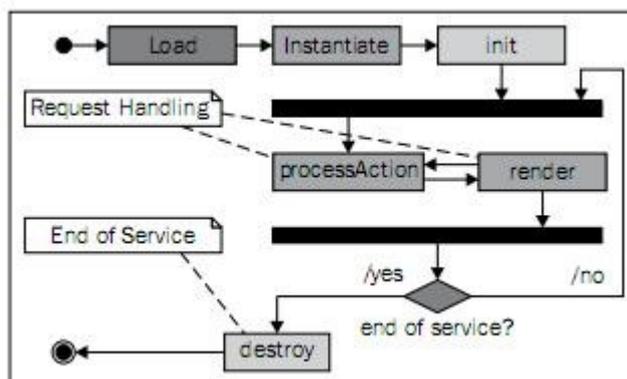


Diagrama 1: Ciclo de vida de un portlet

Fases:

Carga e instanciación: Esta fase puede tener lugar al arrancar el contenedor de portlets o se puede retrasar hasta que se reciba una petición relacionada con el portlet en cuestión.

Inicialización: Una vez que el portlet está instanciado, se ejecuta el método `init()`. Este método es útil, por ejemplo, para iniciar una conexión a una base de datos de la que haga uso la aplicación del portlet.

Manejo de peticiones: Una vez que el portlet está inicializado, el contenedor de portlets puede invocar al mismo para atender las distintas peticiones de los clientes que lo requiera.

Podemos considerar 2 tipos de peticiones distintas:

- Cuando un cliente hace una petición de una página en la que hay varios portlets. Se ejecuta el método `render()` en cada uno de ellos. Se puede considerar que una petición simplemente de “carga”
- Si la petición lleva asociada un formulario que va dirigido a un portlet específico, se ejecuta el método `processAction()` sólo de dicho portlet, además del `render()` de todos.

Puesta fuera de servicio: Un portlet, tras una petición puede mantenerse cargado en el servidor durante un periodo de tiempo. El contenedor de portlets decide si lo mantiene “activo” o no. En el caso en el que el contenedor de portlets decida que no es necesario, se ejecutará el método `destroy()`, en el que es recomendable cierra las conexiones abiertas, hilos en ejecución, entre otros.

5 – Gestor de contenidos

5.1 – Introducción

La construcción de un portal web puede llegar a ser muy laboriosa si no se usan las herramientas adecuadas. La forma “clásica” de trabajar era mediante editores que permitían crear las páginas individualmente. El esfuerzo estaba generalmente dedicado a la creación y no al mantenimiento

del portal. El problema surgía con las webs de noticias o contenidos similares, ya que cada vez que aparecía una nueva noticia era necesario crear una nueva página manualmente. En este caso el coste de mantenimiento sí que era bastante considerable, ya que requería establecer de nuevo la maquetación de la página. Un cambio de diseño, por ejemplo de los márgenes, suponía tener que modificar cada una de las páginas de noticias existentes, un proceso que podía llegar a ser tremendamente laborioso. La separación entre contenido y forma permite que se pueda modificar el aspecto visual de un sitio web sin afectar a los documentos ya creados y libera a los autores de preocuparse por el diseño final de sus páginas.

5.2 – ¿Qué es un gestor de contenido?

Un sistema gestor de contenido (CMS o Content Management System), es un conjunto de herramientas software que se utiliza principalmente para facilitar la gestión de webs, ya sea en Internet o en una intranet, aunque no se limita sólo a las webs. Permiten crear y mantener una web con facilidad, encargándose de los trabajos más tediosos que hasta ahora ocupaban el tiempo de los administradores.



Imagen 12: Diversos gestores de contenidos

Los gestores de contenidos proporcionan un entorno que posibilita la actualización, mantenimiento y ampliación de la web con la colaboración de múltiples usuarios, pudiendo tener cada uno funciones distintas a la mera creación de contenido.

5.3 – Funcionalidades dentro de un CMS

5.3.1 – Creación

Un CMS debe de aportar herramientas para la creación del contenido. Estas herramientas han de ser fáciles de usar y centrarse en el contenido en sí. Generalmente se provee de un editor de texto limitado del tipo WYSIWYG (What You See Is What You Get), en el que se muestra

directamente el resultado del formato del texto pero que al mismo tiempo no modifique demasiado el estilo propio del portal web. Otra posibilidad es el uso de XML y HTML para indicar la estructura y el formato del contenido.

5.3.2 – Gestión

Los documentos creados se depositan en una base de datos central donde también se guardan el resto de datos de la web, como son los datos relativos a los documentos (versión del mismo, autor, fecha de publicación y caducidad, etc.), datos y preferencias de los usuarios, etc.

La estructura de la web se puede configurar con una herramienta que, habitualmente, presenta una visión jerárquica del sitio y permite modificaciones. Mediante esta estructura se puede asignar un grupo a cada área, con responsables, editores, autores y usuarios con diferentes permisos. Eso es imprescindible para facilitar el ciclo de trabajo con un circuito de edición que va desde el autor hasta el responsable final de la publicación. El CMS permite la comunicación entre los miembros del grupo y hace un seguimiento del estado de cada paso del ciclo de trabajo.

5.3.3 – Publicación

Cada contenido creado no es accesible automáticamente. Dependiendo del sistema CMS y la configuración del mismo es posible definir, una vez aprobado el contenido, un periodo de publicación. Automáticamente una página o contenido pasa a ser accesible cuando llega la fecha de publicación y deja de serlo cuando caduca. No obstante por norma general no se elimina, sino que se archiva para futuras referencias.



Diagrama 2: Workflow de publicación

5.3.4 – Presentación

Un CMS puede gestionar automáticamente la accesibilidad de la web, con soporte de normas internacionales de accesibilidad, y adaptarse a las preferencias o necesidades de cada usuario. También puede proporcionar compatibilidad con los diferentes navegadores disponibles en todas

las plataformas (Windows, Linux, Mac, etc.).

El sistema se encarga de gestionar muchos otros aspectos como son los menús de navegación, la jerarquía de la página actual dentro de la web, la creación de enlaces de forma automática, etc. También gestiona todos los módulos, internos o externos, del sistema.

5.4 – Principales ventajas

El uso de CMS otorga al sistema flexibilidad y escalabilidad, por lo que es muy recomendable tanto si se trata de la gestión de una web personal como de una gran empresa. De esta manera podemos obtener como resultado un portal dinámico y lleno de funcionalidades.

Entre las ventajas de la utilización de CMS se encuentran las siguientes:

- **Facilidad para la incorporación de nuevas funcionalidades.** Incluir código para añadir funcionalidades suele implicar la revisión de multitud de páginas. Con un CMS eso puede ser tan simple como incluir un módulo, sin que eso suponga cambios en cada página de forma individual.
- **Mantenimiento de gran cantidad de páginas.** Posibilita la distribución de los trabajos de creación, edición y mantenimiento con permisos de acceso a las diferentes áreas. También se gestionan los metadatos de cada documento, las versiones, las fechas de publicación y caducidad de las páginas y los enlaces rotos, entre otros aspectos.
- **Reutilización de objetos o componentes ya almacenados.**
- **Páginas interactivas.** Las páginas dinámicas no existen en el servidor tal como se reciben en los navegadores, sino que se generan según las peticiones de los usuarios.
- **Facilidad para cambios del aspecto de la web.** Un cambio de diseño puede comportar la revisión de muchas páginas. Los CMS facilitan los cambios con la utilización, por ejemplo, de CSS con lo que se consigue la independencia de presentación y contenido.
- **Consistencia de la web.** La consistencia en una web no quiere decir que todas las páginas sean iguales, sino que existe un orden visual. Un usuario nota enseguida cuándo una página no es igual que el resto de las de la misma web por su aspecto, la ubicación de los objetos o por los cambios en la forma de navegar. Estas diferencias provocan sensación de desorden y dan a entender que la web no ha sido diseñada por profesionales. Los CMS pueden aplicar un mismo estilo en todas las páginas y aplicar una misma estructura mediante patrones de páginas.
- **Control de acceso.** Se pueden establecer grupos de usuarios y/o roles para gestionar cada área de la web, según la función del usuario, ya sea un simple visitante, un creador/aprobador de contenido, administrador, etc.
- **Funcionalidades generales.** Aunque no todas son necesarias, es deseable que un buen sistema CMS posea, entre otras, las siguientes funcionalidades:

- Editor de texto WYSIWYG a través del navegador.
- Herramienta de búsqueda.
- Comunicación entre los usuarios (foros, correo electrónico, chat).
- Posibilidad de creación de noticias y artículos con fechas de publicación y caducidad
- Posibilidad de creación de páginas personales y configuración personalizada por el usuario.
- Ciclo de trabajo con diferentes perfiles de usuarios y grupos de trabajo.
- Carga y descarga de documentos y material multimedia.
- Soporte multiidioma.
- Soporte de sindicación (RSS), múltiples formatos (HTML, Word, Excel, Acrobat, etc.) y navegadores web.
- Estadísticas de uso e informes.

6 – Web Services

Según la W3C, un "Servicio web" es un sistema de software diseñado para permitir interoperabilidad máquina a máquina en una red. En general, los servicios web son sólo APIs web que pueden ser accedidas en una red, como Internet, y son ejecutadas en un sistema remoto.

Esta amplia definición abarca múltiples y diferentes sistemas pero, en general, "Servicio web" suele referir a clientes y servidores que se comunican usando mensajes XML que siguen el estándar.

En definitiva, permite comunicación entre diferentes máquinas, con diferentes plataformas y entre programas distintos. Esta comunicación se logra a través de la adopción de diversos estándares abiertos.

- Web Services Protocol Stack: Así se denomina al conjunto de servicios y protocolos de los servicios web.
- XML (Extensible Markup Language): Es el formato estándar para los datos que se vayan a intercambiar.
- SOAP(Simple Object Access Protocol) o XML-RPC (XML Remote Procedure Call): Protocolos sobre los que se establece el intercambio.
- Otros protocolos: los datos en XML también pueden enviarse de una aplicación a otra mediante protocolos normales como HTTP o FTP
- WSDL (Web Services Description Language): Es el lenguaje de la interfaz pública para los servicios Web. Es una descripción basada en XML de los requisitos funcionales necesarios para establecer una comunicación con los servicios Web.
- UDDI (Universal Description, Discovery and Integration): Protocolo para publicar la información de los servicios Web. Permite comprobar qué servicios web están disponibles.
- WS-Security(Web Service Security): Protocolo de seguridad aceptado como estándar por OASIS (Organization for the Advancement of Structured Information Standards).

Garantiza la autenticación de los actores y la confidencialidad de los mensajes enviados.

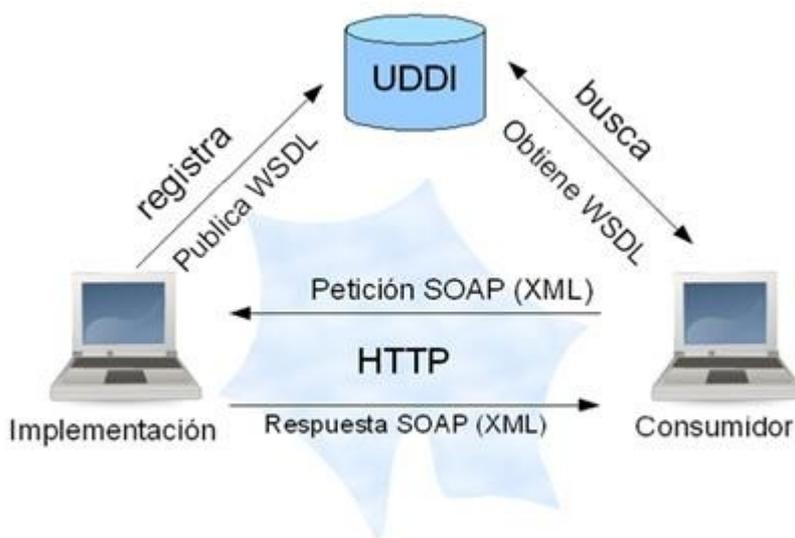


Diagrama 3: Funcionamiento de un Servicio Web

6.1 – Funcionamiento básico

Para que un cliente haga uso de un servicio alojado en otro equipo servidor, en primer lugar, debe de publicarse dicho servicio. Esta publicación consiste en informar de la accesibilidad mediante el protocolo UDDI. De esta forma se ofrecerá toda la información necesaria al cliente, en caso de que solicite saber los servicios ofrecidos y los parámetros de uso de los mismos.

En el lado del cliente, para obtener información del servicio publicado se usa el protocolo UDDI para obtener un documento en formato WSDL. Este documento contiene toda la información de los servicios disponibles, tanto los nombres de los mismos como el número y tipos de parámetros necesarios para llevar a cabo una petición del servicio remoto, pudiendo ofrecer información adicional sobre el servicio o los parámetros implicados.

Para realizar la invocación del servicio se realiza una petición SOAP sobre HTTP, pasando todos los parámetros necesarios. Si el lado servidor acepta la petición ésta procesa convenientemente y se devuelven los resultados correspondiente al cliente.

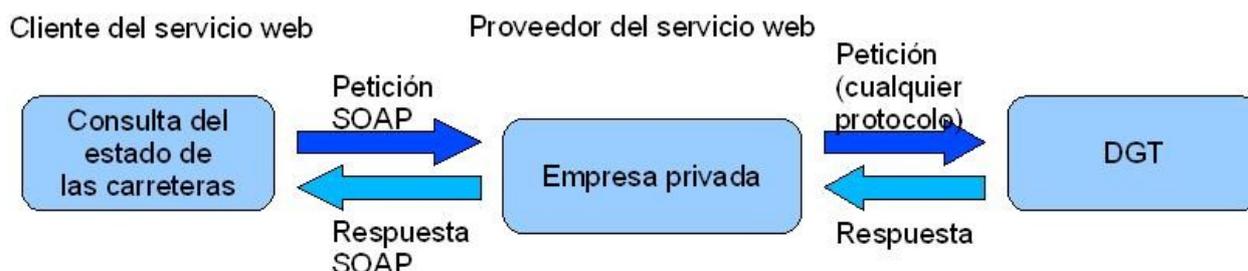


Diagrama 4: Ejemplo de uso de un servicio web

Es necesario indicar que la forma en la que se procesa la petición no está definida. Ésta se puede realizar en la misma máquina o en otra distinta, sobre cualquier lenguaje de programación. Incluso es posible que el servidor haga uso de otro servicio web para obtener parte o totalidad de la información necesaria para satisfacer la petición inicial. Todo de forma transparente al cliente.

6.2 – Ventajas e inconvenientes del uso de los servicios web

Aportan interoperabilidad entre aplicaciones de software independientemente de sus propiedades o de las plataformas sobre las que se instalen, facilitando la integración de servicios dispersos geográficamente.

Se basan en estándares y protocolos basados en texto, haciendo más fácil acceder y entender su contenido y funcionamiento, aunque a costa del rendimiento.

Al emplear HTTP, pueden atravesar un sistema firewall, sin cambiar las reglas de filtrado. La consecuencia de ello es un menor control por parte del mismo.

7 – Generación dinámica de contenido externo

7.1 – Introducción

En el mundo de Internet existen multitud de fuentes de información de las que obtener diversos tipos de contenidos. A menudo dichos contenidos están relativamente dispersos en varias webs, o simplemente se quiere una selección de los mismos para tenerlos localizados en un solo lugar.

Ante esta situación surge una oportunidad para los administradores de sitios públicos para atraer a un mayor público, o simplemente para ofrecer en su portal web una nueva funcionalidad. En estos casos se suele recurrir al uso de contenido dinámico.

El concepto de generación de contenido dinámico en este caso se refiere al hecho de poder usar contenido de fuentes distintas a la propia, de manera que se pueda ofrecer una cantidad mayor de información y/o servicios. Además, dicho contenido es accedido en tiempo real, lo que supone un valor añadido cuando se trata de noticias de actualidad o información del tiempo o estado de las carreteras, por ejemplo.

Existen actualmente multitud de métodos tanto para ofrecer este servicio.

7.2 – Sindicación de contenido

La sindicación de contenidos es un método de redifusión de contenido de un sitio web, que expone dicho contenido, a múltiples sitios webs o clientes que estén correctamente configurados

para acceder al mismo.

En un principio puede parecer que solo beneficia a quienes usan el contenido, puesto que manera rápida y sin coste pueden ofrecer información actualizada, haciendo a los sitios web que hacen uso de ellos más atractivos de cara a los usuarios. Por otro lado el contenido generado puede tener una gran difusión, hecho que se puede aprovechar para mostrar publicidad a una gran variedad de usuarios.



Cada entidad que ofrece información se denomina fuente web o feed. Dichas fuentes suelen codificarse en XML y transportarse mediante HTTP.

Los 2 principales formatos de sindicación de contenidos son RSS (Really Simple Syndication o Sindicación realmente simple) y Atom. Ambos ofrecen prácticamente las mismas funcionalidades aunque la proporción de usuarios está en torno 5 a 1.

Para leer una feed tan solo es necesario suscribirse mediante una aplicación (de escritorio o basada en web) que muestra los contenidos nuevos publicados por el proveedor del feed.

A pesar de que tuvo su origen en páginas de noticias y blogs, cada vez se utiliza más para redifundir cualquier tipo de información. La redifusión web también está ganando importancia en el comercio en línea, ya que los internautas son reacios a proporcionar información personal con fines comerciales (como apuntarse a un boletín de noticias) y en cambio realizan una suscripción a una fuente que permita acceder a sus contenidos.

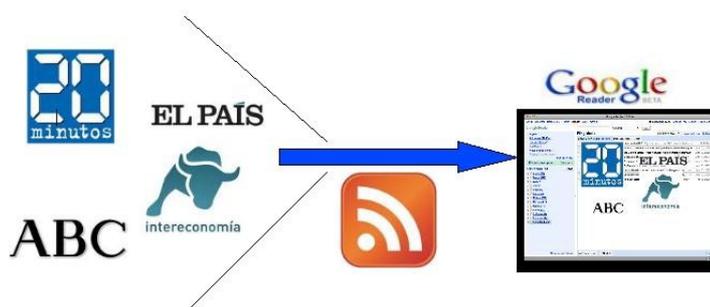


Imagen 13: Agregado de contenido con Google Reader

Para manejar la suscripción a múltiples canales de contenidos de este tipo se suele recurrir a lo que se conoce como agregadores de noticias o contenidos.

Un agregador no es más que un tipo de software que reúne los feeds a los que se está suscrito y muestra las novedades o modificaciones de los mismos desde la última lectura.

Los agregadores también suelen disponer herramientas para organizar, etiquetar y filtrar toda la información, que en muchos casos puede llegar a ser considerable y que sería inmanejable de otra manera.

Ejemplos de agregadores son: Google Reader, Netvibes o Bloglines.

7.3 – Dapper

Aunque el RSS es un estándar y es el mejor camino para seguir las actualizaciones de varios lugares sin tener que visitar cada uno de ellos, aún hay muchas webs que no ofrecen este servicio o esta característica.



Para suplir esta carencia aparece Dapper, un servicio que no solo es capaz de crear un RSS a partir de una web cualquiera, sino que ofrece múltiples formatos de salida de forma que incluso es posible montar una web a partir de contenido de otras, de forma dinámica, en tiempo real y eligiendo que elementos queremos que se muestren y cuáles no.

La idea de Dapper es crear un proceso automático y visual para extraer contenido HTML de diversas páginas web de forma dinámica para poder hacer uso del mismo de la manera que se crea conveniente, haciendo especial hincapié en determinar, de forma visual, qué contenido es el que interesa.

7.3.1 – Dapps

La base de Dapper son los Dapps, ya que todas las acciones de recuperación de información están relacionadas con, al menos, un Dapp.

Los Dapps son los elementos donde están definidos toda la información necesaria para recuperar y mostrar la el contenido deseado. Esta información no lleva los datos finales, sino que posee la información sobre la estructura de una o varias páginas y la forma en la que se va a mostrar la información que finalmente llega al usuario.

Con un Dapp se puede, por ejemplo, obtener el titular de la portada de varios periódicos distintos y mostrarlos en una única página, de forma que cada vez que se acceda a dicha página todas las noticias estén actualizadas.

Los Dapps, además, admiten el uso de variables que se pueden usar como parte de un formulario. Esto, unido a lo anterior puede permitir, por ejemplo, el hacer búsquedas en la Wikipedia, y que del artículo devuelto se muestre solamente una parte, como la introducción.

La creación de Dapps se basa principalmente en clicks y selecciones visuales sobre la web deseada, de forma que con un simple click de ratón elegiremos que parte de la web elegida formará parte de nuestro Dapp. Al estar las selecciones íntimamente ligadas a la estructura HTML de la página, Dapper es especialmente potente cuando el contenido se encuentra en páginas altamente estructuradas, aunque no es un requisito imprescindible.

Muy Interesante (jbb) (details)

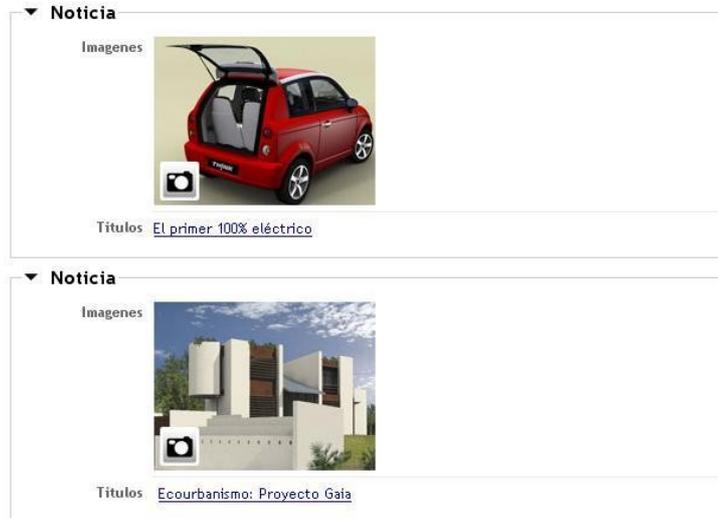


Imagen 14: Ejemplo de Dapp con 2 noticias de la portada de www.muyinteresante.es

7.3.2 – Funcionamiento

Dapper es capaz de convertir cualquier página web en una tipo especial de Web Service, un Data Service (o servicio de datos). La diferencia fundamental entre ambos es el carácter pasivo del último. Mientras en el Web Service se realiza una petición de un servicio concreto que suele llevar aparejado un procesamiento de información, en el Data Service lo que se obtiene es únicamente información. En el caso de Dapper, esta información será procesada para estructurar la información de la página web.

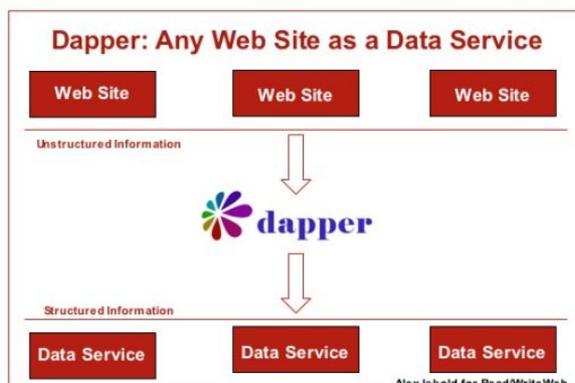


Diagrama 5: Webs como Servicio de Datos

Una vez que el tenemos definido el Dapp, para hacer uso de él, se realiza una llamada a una

determinada URL, pasando como parámetros de la llamada información relevante, como nombre del Dapp a usar, tipo de información que se desea como respuesta y otros parámetros.

Una vez recibida la petición, existe un proceso que se encarga de obtener las páginas web implicadas en el proceso y de aplicarles una serie de algoritmos. En primer lugar se determina la estructura de las páginas y se compara con la estructura que está definida en el Dapp. Posteriormente, de esta estructura se extrae la información que necesite el Dapp. Dicha información se deberá devolver al usuario en el formato que se pidió, como documento XML, feed RSS, como página web (HTML), etc.

```

- <elements xsi:noNamespaceSchemaLocation="http://www.dapper.net/websiteServices/generate-dapp-xsd.php?dappName=MuyInteresantejbb">
- <dapper>
  <status>OK </status>
  <dappName>MuyInteresantejbb</dappName>
  <dappTitle>Muy Interesante (jbb)</dappTitle>
- <urls>
  <url>http://www.muyinteresante.es/</url>
</urls>
<applyToUrl>http://www.muyinteresante.es/</applyToUrl>
<executionTime>0.188</executionTime>
<ranAt>2010-06-13 16:34:11</ranAt>
<encoding>UTF-8</encoding>
<ranEventChain>>false</ranEventChain>
<inputVars/>
</dapper>
- <Noticia groupName="Noticia" type="group">
  <Imagenes dataType="RawString" fieldName="Imagenes" href="http://www.muyinteresante.es/fotos-think-city-primer-coche-electrico-comercializado-espana"
  originalElement="img" src="http://www.muyinteresante.es/images/stories/portada/think-city-p3.jpg" type="field"/>
  <Titulos dataType="RawString" fieldName="Titulos" href="http://www.muyinteresante.es/fotos-think-city-primer-coche-electrico-comercializado-espana"
  originalElement="span" type="field">El primer 100% eléctrico</Titulos>
</Noticia>
- <Noticia groupName="Noticia" type="group">
  <Imagenes dataType="RawString" fieldName="Imagenes" href="http://www.muyinteresante.es/fotos-ecourbanismo-proyecto-gaia" originalElement="img"
  src="http://www.muyinteresante.es/images/stories/portada/proyecto-gaia-p3.jpg" type="field"/>
  <Titulos dataType="RawString" fieldName="Titulos" href="http://www.muyinteresante.es/fotos-ecourbanismo-proyecto-gaia" originalElement="span"
  type="field">¿Ecourbanismo: Proyecto Gaia</Titulos>
</Noticia>

```

Imagen 15: ejemplo de extracción en XML

7.3.3 – Herramientas de desarrollo

Pese a que la creación de un Dapp es bastante sencilla y el uso de la web es muy intuitivo, Dapper es una herramienta bastante avanzada. Existe una comunidad de apoyo detrás de Dapper que se encarga de desarrollar nuevos servicios y mejorar los ya existentes. Para esta comunidad se ofrecen una serie de herramientas y APIs con el objetivo de facilitar su tarea.

Biblioteca de “transformadores”

Los transformadores son un tipo de mini aplicaciones que se usan para transformar la información recibida en formato XML a otro tipo de formato de acuerdo con las necesidades.

Existen una gran variedad de transformadores, que permiten, por ejemplo el mostrar la información recogida por la herramienta en formato HTML, para su visualización directa en una página web. Otros transformadores pueden convertir la información a un formato adecuado como

RSS o usable por un Widget, por ejemplo.

Web Services

Una de las posibles alternativas para usar la aplicación es mediante el uso de web services para la búsqueda de Dapps creados con anterioridad por cualquier usuario, siempre que el creador haya permitido esta posibilidad durante el proceso de creación.

Este servicio web sirve de interfaz para buscar Dapps por nombre, por categoría, mediante el uso de etiquetas o por el nombre de los campos de los que se hagan uso.

Como resultado de esta búsqueda podemos obtener una lista con formato XML o JSON, que podremos usar aplicándole el proceso de parseo correspondiente.

SDK

Existe un SDK (software Developer Kit o Kit de desarrollo de software) desarrollado por la propia compañía que provee un mecanismo para acceder a los distintos Dapps y obtener la información. De esta forma se ahorra tiempo a la hora de programar los métodos para extraer el contenido XML de las webs. Actualmente existen versiones de SDK para PHP 4, PHP 5 y Java 5.

8 – Software libre

Una característica deseable es el uso de software libre y gratuito en todos los componentes de la plataforma, no solo por el ahorro de costes, sino también por una serie de ventajas. Para poner de manifiesto la bondad de esta decisión se van a señalar ciertos puntos importantes.

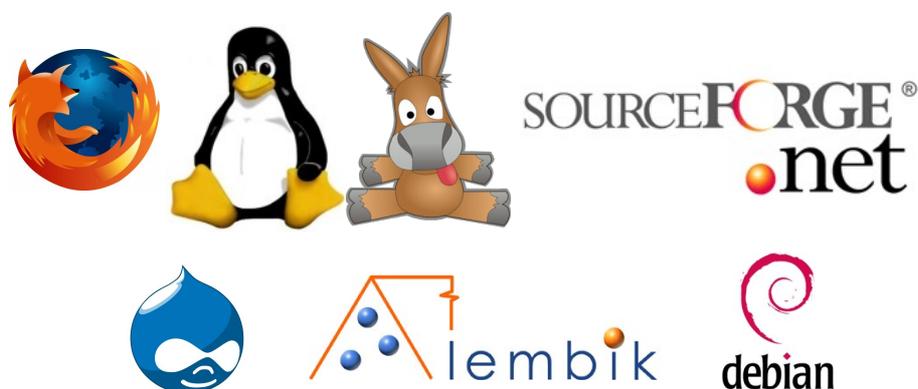


Imagen 16: Ejemplos de software libre

8.1 – Código fuente abierto

El código fuente abierto es aquel que está desarrollado por individuos, grupos o empresas que permiten el acceso libre y la modificación del código fuente.

Por contraposición está el software propietario o código fuente cerrado, comercializado por empresas, que consideran el código fuente un activo más que tienen que mantener en propiedad, por lo que no permiten que terceros tengan acceso al mismo.

La disponibilidad del código fuente posibilita que se hagan personalizaciones del producto, correcciones de errores y desarrollo de nuevas funciones. Este hecho es una garantía de que el producto podrá evolucionar incluso después de la desaparición del grupo o empresa creadora y de paso se evita el caer en un mercado cautivo, con el riesgo que esto conlleva.

Algunas empresas también dan acceso al código, pero sólo con la adquisición de una licencia especial o después de su desaparición.

8.2 – Flexibilidad

Los programas código abierto son mucho más flexibles que los propietarios, ya que cualquier persona con suficientes conocimientos podría modificar el código, corrigiendo los errores que aparezcan y mejorarlo por ello el producto.

En el caso de los sistemas propietarios, generalmente las modificaciones sólo pueden hacerlas los mismos desarrolladores, y siempre según sus prioridades.

8.3 – Estabilidad

Se podría considerar que la herramienta comercial será más estable y coherente al estar desarrollada por un mismo grupo. En la práctica esta ventaja no es tan grande, ya que los programas de código abierto de mayor relevancia también están coordinados por un único grupo o por empresas, de forma similar a los comerciales.

8.4 – Coste

Utilizar un software de código abierto tiene otra ventaja que hace decidirse a la mayoría de usuarios: su coste. Habitualmente todo el software de código abierto es de acceso libre, es decir, sin ningún coste en licencias. Sólo en casos aislados se hacen distinciones entre empresas y entidades sin ánimo de lucro o particulares. Los productos comerciales pueden llegar a tener un coste que sólo una gran empresa puede asumir.

8.5 – Soporte

En cuanto al soporte, el software comercial acostumbra a dar soporte profesional, con un coste elevado en muchos casos, mientras que los de código abierto se basan más en las comunidades de usuarios que comparten información y se ayudan entre sí para dar solución a los problemas. Las formas de soporte se pueden mezclar, y así encontramos software de código abierto con empresas que ofrecen servicios de valor añadido y con activas comunidades de usuarios. En el caso comercial también sucede, pero el coste de las licencias hace que el gran público se decante por otras opciones y por lo tanto las comunidades de soporte de este tipo son más pequeñas.

8.6 – Documentación

Pese a todo lo anterior, existe un problema que acostumbra a tener el software de código abierto: la documentación generalmente es escasa y está dirigida a usuarios técnicos o está mal redactada. Este problema se agrava en el caso de los módulos desarrollados por terceros, que no siempre incorporan las instrucciones de su funcionamiento de forma completa y entendible.

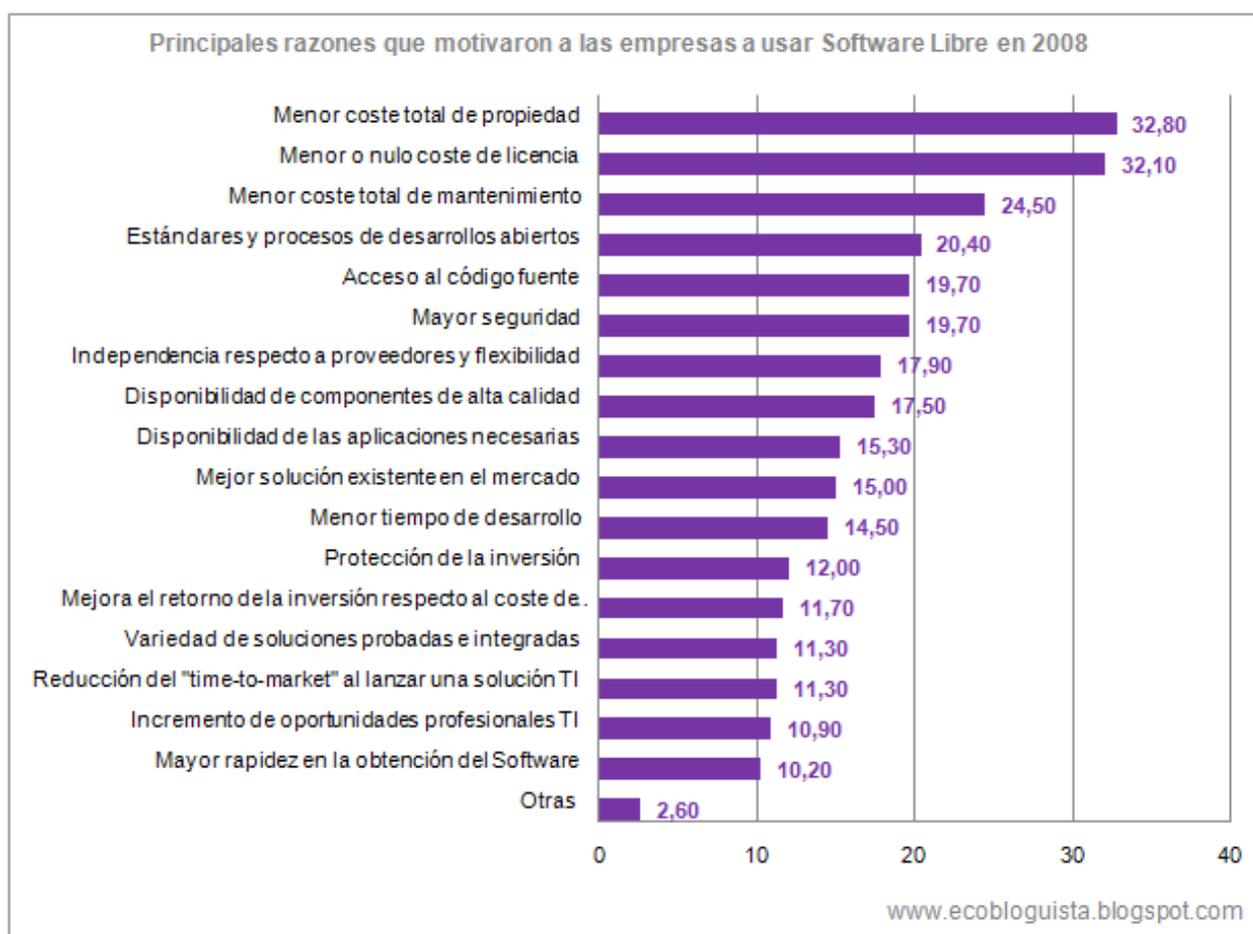


Gráfico 1: Razones por las que empresas adoptan software libre