

ESCRIPNAUTA: sistema de creación y generación de presentaciones multimedia

Ricardo Galli, Juan Pedrosa, Iker Tolosa, Valentín Ginard,
Carlos Guerrero, Antoni Bibiloni, Llorenç Valverde

Universitat de les Illes Balears

Resumen

El ESCRIPNAUTA es una aplicación desarrollada totalmente con Software Libre [Debian], orientada a la creación de redacciones multimedia en línea. El objetivo fundamental es la creación de una herramienta que permita la creación remota y cooperativa de presentaciones multimedia, integrando texto, imágenes y sonidos mediante el uso de sólo un navegador web estándar y una conexión a Internet. El sistema ha sido íntegramente desarrollado con el software disponible en la distribución Debian, principalmente Apache, PHP, MySQL, Ming y utilidades de ImageMagick

La aplicación se ha utilizado para la promoción de la sección joven de Banca March [March] mediante el concurso ESCRIPNAUTA [Escrip], donde los alumnos de los institutos de educación secundaria de las islas han podido crear sus “redacciones multimedia”.

El desarrollo específico del ESCRIPNAUTA comenzó en junio de 2002 y la versión final estuvo acabada a finales de enero de 2003. El periodo de pruebas, en el que participaron unos pocos institutos de la isla de Mallorca, abarcó desde diciembre de 2002 a enero de 2003.

La aplicación se estructura principalmente en tres módulos diferenciados: un sistema de gestión de contenidos, una herramienta de edición y un generador dinámico de redacciones multimedia. Todo ello está vinculado a una estructura de datos dinámica que permite enlazar los diferentes módulos.

1. Introducción

Las aplicaciones multimedia en Internet han experimentado un auge con la expansión de la banda ancha y la mejora en las tecnologías de compresión para aprovechar los recursos existentes. El ESCRIPNAUTA nació con la filosofía de llegar al máximo número de usuarios posible sin discriminación de sistema operativo y/o navegador, basándose en estándares y software libre.

Un grupo de becarios e investigadores del *Dept. de Matemàtiques i Informàtica* de la *Universitat de les Illes Balears* ya había participado en el Proyecto Europeo de Investigación y Desarrollo *Minority Newspapers to Multimedia* [MNM, Gel02, Mari02, Guer02, Mero03]. En dicho proyecto se desarrolló un prototipo basado en estándares y software libre para generar la salida de noticias periodísticas en varios formatos y dispositivos a partir de una base de datos única.

Durante la última fase del proyecto también se demostró la posibilidad de integrar las diversas fuentes (texto, imágenes, audio) para generar una salida multimedia basada en el estándar SMIL [Smil].

En contactos posteriores con directivos de la Banca March se vislumbraron las capacidades del prototipo como herramienta de creación audiovisual. Así surgió la idea de ampliarlo para permitir que usuarios “inexpertos” puedan crear sus propias

presentaciones multimedia, con el único requisito de tener disponible un navegador y conexión a Internet.

2. Objetivos

El objetivo fue desarrollar una herramienta que combine imágenes, texto y audio para crear presentaciones multimedia, usando un navegador web. De esta forma se ofrece a los usuarios un entorno de creación común y *universal*. Ésto evita las desigualdades o ventajas derivadas de que algunos centros o alumnos cuenten con más recursos de software que otros.

Además el sistema debe permitir modificar cualquier elemento de las animaciones, *en cualquier momento, en cualquier lugar*.

Otro de los objetivos era que los participantes trabajen en grupos y desde cualquier lugar con conexión a Internet. Por tanto el sistema debe soportar el trabajo colaborativo, es decir, que varias personas puedan trabajar simultáneamente sin que se produzcan inconsistencias en la información.

2.1. Requisitos de la aplicación

- Integración de elementos multimedia: texto, imágenes, sonido y animación de dichos elementos.
- Entorno de diseño de diapositivas simple e intuitivo.
- Permitir edición de redacciones multiusuario.
- Transferencia de contenidos multimedia personalizados por parte de los usuarios.
- Aplicación en línea multiplataforma y compatible con los estándares y que funcione en la mayoría de los navegadores (específicamente Mozilla, IE y Safari).
- Sistema de gestión de concursos, usuarios con varios niveles de privilegios (administradores, responsables de centros y autores) y centros.

3. Arquitectura del sistema

El sistema se basa en una arquitectura cliente-servidor centralizada. La interfaz de usuario reside completamente en el navegador web. En el servidor residen todos los demás módulos: base de datos y *middleware* de acceso a datos, lógica de edición, gestor de contenidos (galería) y el generador de las presentaciones.

El hecho de ser una aplicación cuya lógica esta totalmente centralizada confiere al sistema una gran flexibilidad, tanto de desarrollo como de acceso y utilización. De esta forma se da la oportunidad de desarrollar completamente el sistema sobre Linux y SL al ser independiente de las instalaciones y recursos de los usuarios.

El sistema funciona sobre un servidor Apache, con base de datos MySQL, PHP como lenguaje de programación y las librerías Ming para la generación dinámica de las presentaciones a partir de la información almacenada en la base de datos.

En los clientes se utiliza principalmente DHTML para la edición de las presentaciones y SWF [OpSWF] para la salida multimedia.

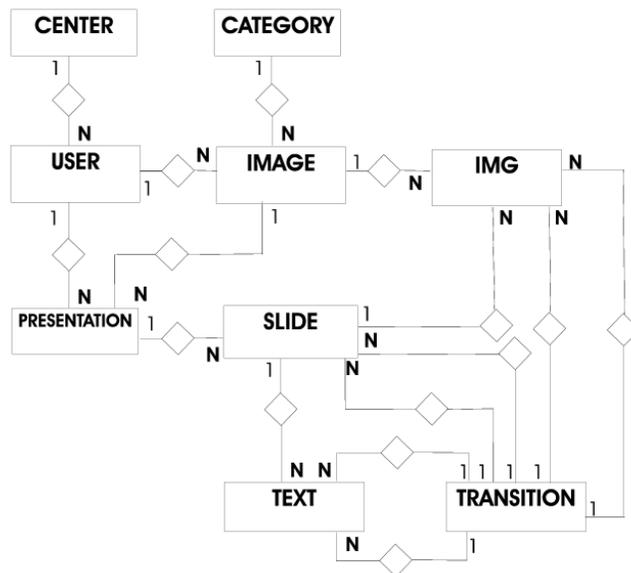


Figura 1: Diagrama entidad relación de la base de datos

3.1. Estructura de datos

La estructura de datos es el vínculo entre los distintos módulos de la aplicación y está definida por la base de datos (Figura 1) y una serie de módulos en PHP, el *middleware*, que facilita el tratamiento de los datos mediante la exposición de una estructura orientada a objetos al resto de los módulos del sistema.

Los métodos de cada una de las clases permiten separar las operaciones de consultas a la base de datos del tratamiento de estos datos por parte de la aplicación.

Una presentación o redacción multimedia está formada por diapositivas que a su vez contienen elementos multimedia configurables.

Los objetos básicos almacenados en la base de datos son:

- *Slide*: define cada “diapositiva” de una presentación. Una diapositiva contiene elementos de texto o imágenes descritos en las tablas *Text* e *Img*. Cada una de las diapositivas que compone una presentación siempre están relacionadas con una presentación.
- *Text*: define cada uno de los elementos de texto de cada diapositiva. Los textos están definidos por la caja que los contiene, la posición inicial y final en la diapositiva, color de fondo, color de texto, tipo de letra, transición de entrada, transición de salida, etc. (Figura 3). También pueden asignarse enlaces hipertexto (URL) a cada caja de texto.
- *Img*: al igual que el texto, los gráficos están definidos por cajas pero contienen imágenes, en cualquier formato gráfico, en vez de texto (Figura 4). En esta tabla se almacena la información de las imágenes que pertenecen a cada diapositiva, está relacionada con *Slide* y con *Image*. Sus atributos determinan el momento de inicio en la diapositiva, duración, posición, transparencia así como sus transiciones de entrada y salida. Las imágenes deben estar previamente almacenadas en la galería de imágenes¹. Se pueden incluir en una diapositiva siempre que se dispongan de los privilegios necesarios.

¹Tabla *Image*.

- *Transition*: esta tabla define los efectos de transiciones que pueden aplicarse tanto a cajas de texto e imágenes como a las transiciones entre diapositivas. A efectos de que se puedan generar salidas con distintos formatos (SMIL, SWF, DOM/Javascript), las equivalencias son almacenadas en esta tabla.
- *Image*: Esta tabla almacena el contenido (*blobs*) de las imágenes cargadas en la galería. La información almacenada incluye tipos MIME, propietario, tamaño, y los permisos de cada una de ellas:
 - privada: sólo para el usuario que la ha cargado, o
 - pública: puede ser usada por cualquier otro usuario.
- *Category*: Los elementos multimedia de la aplicación se dividen en categorías para facilitar la navegación y selección.
- *Presentation*: mantiene la secuencia de diapositivas que conforman una presentación completa. Los elementos de esta tabla son creados por los usuarios con los privilegios adecuados². A cada presentación se le puede asignar un audio como música de fondo. Los ficheros de audios deben estar almacenados previamente en galería de audios.
- *User*: esta tabla mantiene la información y privilegios de los usuarios registrados en el sistema. Cada presentación u elemento de la galería pertenece a un usuario.
- *Entity*: En esta tabla se describen distintos centros docentes que participan en el concurso, cada usuario del sistema pertenece a una de las entidades.

3.2. Gestión de la galería

La gestión de la galería hace referencia a la posibilidad del sistema de almacenar elementos multimedia (imágenes y audio) de carácter público o privado para, posteriormente, ser incluidos como elementos de una redacción multimedia. Estos elementos conformarán una galería multimedia accesible desde el entorno de edición (Figura 2).

En la galería se almacenan los ficheros gráficos en cualquier formato reconocido por la librería ImageMagick [IMagick]. Debido a los problemas de visualización de gráficos distintos a JPEG el sistema convierte el fichero original a dicho formato. Además, escala el gráfico al tamaño máximo en pixels admitido por el sistema y genera una imagen en miniatura para facilitar la edición y visualización (como se observa en el panel de la izquierda de la figura 2) y reducir los requerimientos de ancho de banda.

3.3. Editor

Las redacciones multimedia se basan en el concepto de línea de tiempo. Los distintos elementos multimedia se distribuyen en dicha estructura temporal, dependiendo de su momento de inicio y finalización.

El concepto de presentación multimedia hace necesario el control de distintos elementos: diapositivas, objetos texto, objetos imagen, sonido. Esta distinción ha permitido la estructuración del editor en *iframes* funcionales en un mismo documento web:

- *iframe diapositiva*: el marco de la izquierda de las figuras 3 y 4,

²“Alumno” en la implementación actual.

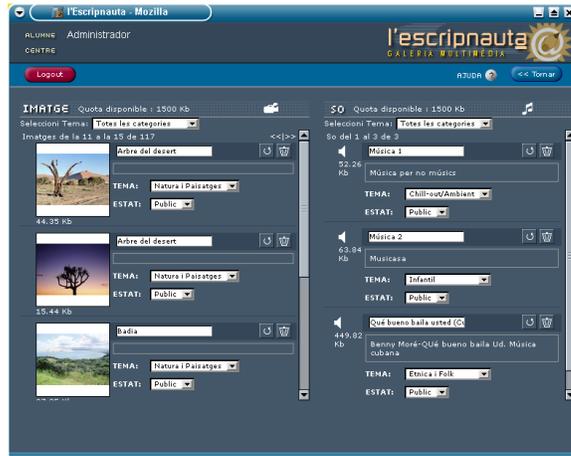


Figura 2: Galería de imágenes y audio

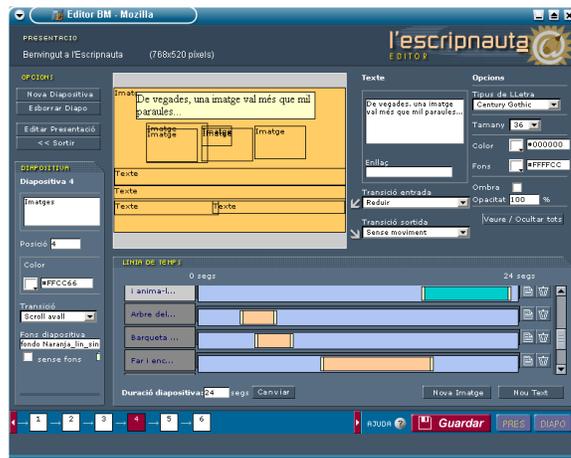


Figura 3: Edición de un objeto texto

- *iframe línea de tiempo*: el marco de la parte inferior derecha,
- *iframe texto/imagen*: el marco de la parte superior derecha.

Cada uno de ellos contiene un formulario y otros elementos DHTML que permiten configurar los distintos parámetros de cada objeto o elemento multimedia. Cada uno de estos elementos dispone de una serie de parámetros configurables (tamaño, color, tipo de letra, posición en la diapositiva, efectos de transición de entrada y salida, opacidad, etc.). El editor (Figura 4) permite añadir y modificar estos objetos y parámetros que conforman una redacción multimedia.

De esta forma, se han conseguido organizar los distintos aspectos de cada posible elemento de manera sencilla e intuitiva en un único documento web. En consecuencia, una sesión de trabajo típica sería la siguiente:

1. El usuario crea una presentación nueva o selecciona la presentación que desea modificar.
2. La aplicación realiza una consulta a la base de datos y carga los datos pertinentes.



Figura 4: Edición de un objeto imagen

3. El usuario trabaja en su entorno local sin generar conexiones al servidor: decide sobre qué elemento trabajar utilizando el contenido de los *iframes* correspondientes, realiza las modificaciones que considera oportunas y una vez finalizado puede guardar todo su trabajo.
4. La aplicación envía las sentencias necesarias para incluir las últimas modificaciones y así actualizar y así actualizar la base de datos.

Como se puede deducir de los puntos 3 y 4, las conexiones al servidor se reducen al máximo, sólo se producen cuando hay que acceder o guardar los cambios a cada objeto. Todas las demás operaciones se realizan localmente³, por lo que no sólo se reduce el uso de ancho de banda, sino que se mejoran considerablemente los tiempos de respuesta de la interfaz.

3.3.1. Programación de la interfaz DOM

El desarrollo del editor es compatible con el modelo DOM de nivel 3 a fin de garantizar la compatibilidad entre navegadores. La programación de la interfaz del editor fue una de las tareas más complejas por su sofisticación y porque debía funcionar en navegadores diferentes, algunos de ellos no totalmente compatibles con los estándares. El editor se ha realizado íntegramente programando en HTML, DHTML y Javascript.

Para facilitar el desarrollo y mantenimiento del numeroso contenido DHTML se han utilizado las librerías CBE, con licencia GNU LGPL, creadas por Mike Foster [CrBrow]. Las librerías CBE originales disponen de cinco módulos de los cuales hemos utilizado dos:

1. Módulo *CrossBrowserElement*:
 - a) *CBE Basics*: Creación de objetos DHTML, tamaño, recorte, posicionamiento en el documento, z-index, visibilidad, etc.
 - b) *CBE Sliding*: Transiciones y desplazamientos ya definidos.

³Se puede decir que en realidad hemos pasado una parte pequeña de la lógica de aplicación hacia el cliente, pero en este caso lo consideramos más que provechosa a esta “descentralización”.

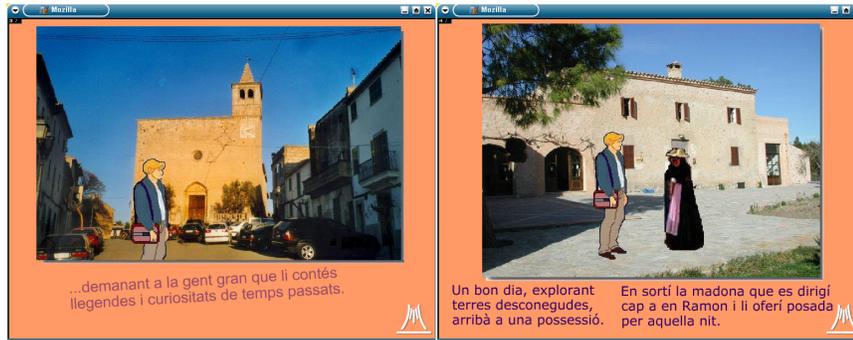


Figura 5: Presentación Flash

- c) *CBE Drag/Resize*: Ofrece la posibilidad de arrastrar elementos por el documento web. Creación de barras de desplazamiento. Redimensionado de objetos.
- 2. Módulo *CrossBrowserEvent*: En estas funciones se controlan los posibles eventos generados por el usuario en el editor, mediante teclado o ratón, sobre los objetos DHTML.

3.4. Motor de generación de las presentaciones

El motor de generación de redacciones multimedia se encarga de recuperar una estructura de datos previamente cumplimentada y almacenada en la base de datos durante el proceso de edición. El generador está enteramente programado en PHP y se ejecuta para cada petición que se realiza desde un navegador.

En cada ejecución se crean en memoria los elementos multimedia según sus diferentes parámetros y finalmente se pasa al cliente el *stream* ya en formato SWF. Desde el punto de vista del cliente es como si recuperase un fichero SWF estándar (Figura 5).

La ventaja de generar las presentaciones cada vez y bajo demanda es que eliminamos problemas de mantenimiento de ficheros temporales y posibles inconsistencias con la base de datos. En todas las presentaciones que se han realizado durante el concurso no hubo en ningún momento sobrecarga del servidor (un Pentium 3 Tuatlatin 1.4 GHz con 512 MB de RAM) y todas las presentaciones se generan en menos de un segundo.

Se ha tenido mucho cuidado de usar todas las posibilidades del SWF a efectos de disminuir el consumo de ancho de banda mediante la reutilización de objetos que aparecen varias veces en las presentaciones⁴ y de un pre-procesado, con ImageMagick, de las imágenes que se almacenan en la galería a efectos de maximizar la compresión teniendo en cuenta el tamaño máximo que tienen las presentaciones (800x600 pixels).

Debido a que algunas de las presentaciones generaban ficheros de más de 1 MB, el generador también agrega automáticamente un *preloader* para mostrar casi inmediatamente una pantalla de inicio e indicar al usuario cuantos bytes restan por bajar.

Todo el proceso se realiza en el mismo servidor Linux donde se ejecuta el Apache y el MySQL. El desarrollo está basado completamente en librerías Ming y el paquete php4-ming [Ming] distribuidos con Debian.

⁴En realidad fue muy sencillo de implementar gracias a las tablas con *hashing* del PHP.



Figura 6: Pàgina principal del ESCRIPNAUTA [Escrip]

3.5. Interfaz web

El acceso al sistema, tanto como autor, administrador o visitante anónimo, se realiza desde un documento web único (Figura 6, [Escrip]) que gestiona la detección de los requisitos mínimos del sistema cliente (navegador compatible y *plug-in* del reproductor de Flash). En dicho documento se permite la visualización de las redacciones multimedia publicadas y el acceso a la información sobre el concurso.

El módulo anterior enlaza con el de gestión de usuarios a través de un proceso de login. Una vez validado por el sistema, el usuario tendrá acceso a unos u otros contenidos dependiendo de su nivel de privilegios.

4. Conclusiones

Los objetivos fundamentales fueron dos:

1. Crear un sistema de generación de presentaciones multimedia *universal*, al alcance *de todos* los estudiantes y que no obligase a instalaciones de software o gastos de licencias, ni que fomentase la piratería.
2. Basar absolutamente todo el desarrollo en Linux y sólo con Software Libre.⁵

Consideramos que ambos objetivos se han cumplido completamente, dado que el concurso se ha desarrollado con total normalidad. El sistema ha funcionado veinticuatro horas al día, siete días a la semana, desde el día de su puesta en marcha. Tampoco se han descubierto errores importantes. Las únicas quejas que hemos recibido fueron problemas de saturación de red, probablemente relacionados con la saturación de las líneas de Internet o del ADSL instalado en los institutos.

Como resultado del primer concurso, la Banca March ya tiene planificado hacer una segunda versión en Baleares, y en setiembre hacer otro concurso en Canarias (allí se llamará ESCRITNAUTA).

⁵Para asegurarnos que así fuese, hemos usado Debian y sólo los paquetes oficiales.

El esfuerzo de respetar estándares

Otra cosa que queríamos demostrar, y consideramos que lo hemos logrado (aunque con bastante trabajo), es que se pueden hacer páginas web tan complejas y sofisticadas como el editor de presentaciones respetando estrictamente los estándares y haciendo que funcionen en todos los navegadores que hemos tenido acceso.⁶

Durante el desarrollo del proyecto nos hemos encontrado en determinados momentos donde no podíamos resolver un problema muy complejo en alguno de los navegadores y había la tendencia a pensar:

Y todo este trabajo sólo por si aparece una persona que se le ocurra usar el navegador XYZ, que nadie usa en estas islas, y aún menos en los institutos donde son todos Windows con IE.

Aunque fue muy duro, y hemos perdido quizás cientos de horas en el proceso, al final queda la satisfacción que nadie, durante más de seis meses, se ha quejado de que no haya podido usar el sistema⁷.

El abuso de las posibilidades

Como aspecto “negativo” nos vemos obligados a comentar que muchas de las presentaciones realizadas son bastantes “aburridas” debido principalmente a la duración exagerada de algunas de ellas (el límite impuesto era de diez minutos). Suponemos que es debido a la fiabilidad y facilidad de uso del sistema, que hacía que los chicos se divirtieran bastante creando sus redacciones.

Como no podemos hacer que el sistema sea más complicado de usar (por razones profesionales⁸), y menos aún disminuir la fiabilidad (¡que es una Debian!), creemos que la solución razonable para el siguiente concurso será limitar las presentaciones a no más de minuto y medio.

Agradecimientos: queremos agradecer especialmente a Jaume Bellido y Jaume Borrás, los responsables del proyecto por parte de la Banca March, por haber confiado en nosotros, su paciencia infinita con estos “universitarios” y por apostar por el Software Libre desde el inicio.

Referencias

- [Gel02] Gelabert, S. “Publicació de notícies sobre dispositius múltiples: mòduls d’estandardització de notícies i sortida multimèdia”. UIB. *Projecto Final de Carrera*. Abril 2002.
- [Guer02] Guerrero, C. “Publicación de noticias sobre dispositivos múltiples: acceso a datos, middleware XML y administración de contenidos”. UIB. *Projecto final de carrera de Ingeniería de informática*. Abril 2002. <http://mmm.uib.es/~carlos/PFC/html.htm>
- [Mari02] Marimón, L. “Publicación de noticias sobre dispositivos múltiples: salida WAP, SMS y PDF”. UIB. *Projecto Final de Carrera*. Abril 2002. <http://mmm.uib.es/trabajo/XML/lina/>

⁶En realidad no hemos podido solucionar un problema con los *iframes* animados del editor en el Konqueror y otros con el Javascript del Netscape 4.7x, pero no inutiliza totalmente al editor ni afecta a la parte pública.

⁷Aunque hay gente que ha tenido problemas con algunas versiones del IE 5.0.

⁸Es una pena que el uso de emoticones no sea demasiado ortodoxo en un artículo técnico.

- [Mero03] Mérou, J. “Publicación de noticias sobre dispositivos múltiples: interfaz XML-RPC”. UIB. *Proyecto final de carrera de ingeniería de telecomunicaciones técnica especialidad en telemática*. Julio 2003.
- [March] Banca March. <http://www.bancamarch.es/>
- [CrBrow] *Cross-Browser DHTML*. <http://www.cross-browser.com/>
- [Debian] *Debian Free Operating System*. <http://www.debian.org/>
- [Escrip] *Escripnauta*. Banca March, Universitat de les Illes Balears. <http://escripnauta.bancamarch.es/>
- [MNM] eContent 3316. *Minority News Paper to New Media*. <http://mnm.uib.es/>
- [Ming] *Ming - an SWF output library and PHP module*. <http://ming.sourceforge.net/>
- [IMagick] *ImageMagick - Convert, Edit, and Compose Images*. <http://www.imagemagick.org/>
- [Smil] *The Synchronized Multimedia Integration Language*. W3C. <http://www.w3.org/AudioVideo/>
- [OpSWF] OpenSWF.org. <http://www.openswf.org/>